Linköping Studies in Science and Technology
Thesis No. 1122

An Automotive Lane Guidance System

Andreas Eidehall

Division of Control & Communication
Department of Electrical Engineering
Linköpings universitet, SE–581 83 Linköping, Sweden
www.control.isy.liu.se
eidehall@isy.liu.se

Linköping 2004

An Automotive Lane Guidance System

Department of Electrical Engineering
Linköpings universitet
SE–581 83 Linköping
Sweden.

ISBN 91-85295-63-9
ISSN 0280-7971
LiU-TEK-LIC-2004:51

Printed by UniTryck, Linköping, Sweden 2004

Abstract

Automotive lane guidance systems, usually referred to as Lane Keeping Aid or Lane Keeping System, are designed to prevent or warn the driver of lane departure. They typically use a buzzer to alert the driver or a steering wheel torque to actually steer the vehicle back into the center of the lane. Emergency Lane Assist (ELA) combines conventional lane guidance systems with a threat assessment module that tries to activate the lane guidance interventions according to the actual risk level of lane departure. The goal is to only prevent dangerous lane departure manoeuvres.

Such a threat assessment algorithm is dependent on detailed information about the vehicle surroundings, i.e., positions and motion of other vehicles, but also information about road and lane geometry parameters such as lane width and road curvature. The thesis demonstrates that the lane estimate can be improved by using an integrated filter that combines information from object and lane tracking. This is done by introducing a road aligned, curved coordinate system which also brings other advantages when it comes to modelling and prediction.

Evaluation of the integrated tracking system has been carried out on real data and the ELA decision algorithm has been tested in a demonstrator. ELA successfully distinguishes between dangerous and safe lane changes on a small set of test scenarios and is, if activated, able to take control of the vehicle and put it in a safe position in the original lane.

Keywords: *active safety, collision avoidance, lane guidance, state estimation, target tracking, Kalman filter, centralized filtering, threat assessment*

Acknowledgements

First of all, I want to thank Volvo Car Corporation, the Volvo Ph.D. program and my manager Robert Hansson for giving me the opportunity to work with this project and with the inspiring people at the Vehicle Dynamic & Active Safety group. I also would like to thank Professor Lennart Ljung for allowing me to join Automatic Control in Linköping, the most ambitious and encouraging group of people imaginable.

I want to thank my academic supervisor, Professor Fredrik Gustafsson, who has supported and inspired me throughout the project. I could not imagine a better supervisor. Thanks also to my industrial supervisor, Doctor Jochen Pohl, for great support and many rewarding discussions.

I also would like to thank Robert Hansson, and Jonas Ekmark at the Vehicle Dynamics & Active Safety group at Volvo Cars who have both provided excellent guidance and inspiration.

Next, I would like to thank Lena Westervall, also at Volvo Cars, who has worked with me during the last phase of the project. Your help with developing the demonstrator has been invaluable and has no doubt improved its performance substantially.

Eleonor Åkesson at the Volvo Safety Center has helped me navigate through the enormous road traffic accident databases at your department. Thanks a lot!

Several people have read and commented this thesis. Special thanks to Thomas Schön and Jonas Jansson. Your reviews were extremely helpful.

Slutligen ett stort tack till min familj som alltid stöttat mig vad jag än tagit mig för och inte minst Anna som stått ut med mina många resor till Linköping. Du är bäst!

Andreas Eidehall
Linköping, October 2004

Contents

Chapter 1

Introduction

1.1 Background

In the European Union, nearly 40,000 people are killed in traffic each year [3]. Naturally, the automotive industry is more focused on safety technology than ever. With the rise of well established car safety test institutions like EuroNCAP (European New Car Assessment Program), automotive safety systems are becoming important means of competition.

Safety systems are often put in the categories *passive* and *active*, although in which category certain systems belong are not always clear. Passive safety systems are designed to mitigate the effects of accidents when they happen, examples are air bags and seat belts. Active safety systems, on the other hand, try to help the driver prevent accidents *before* they happen. Examples are ABS (Anti-lock Braking System) and anti-spin systems. Other examples of active safety are the see-through A-pillars of the Volvo Safety Concept Car, or even the air conditioning system, keeping the driver alert. See [17] for a thorough discussion on this topic.

With the rise of affordable computer and sensor technology, more advanced active safety systems have become a major area of research in the automotive industry. The next generation of active safety technology are collision warning and avoidance systems, which are now starting to emerge.

Collision warning systems aims at preventing accidents by warning the driver of an approaching threat where the warning has to come early enough to allow a braking or a steering manoeuvre. Collision avoidance refers to systems that autonomously intervene in the driving situation, of course raising tough liability questions.

Trends in automotive active safety

This section will give a brief overview and some of the milestones of the development of active safety systems during the last few decades. Here we have focused on mechatronic chassis functions.

Anti-lock Braking System (1978) ABS prevents the wheels from locking and will maintain the steering ability of the vehicle during hard braking. ABS will also, during bad road conditions, reduce the stopping distance. The system measures the velocity on all four wheels, and if one of the sensors reports an abnormal acceleration (higher than a physically reasonable value) it concludes that the wheel is about to lock, and the pressure in the braking system is reduced. The German automotive supplier Bosch actually has a patent from 1936 for a "mechanism to prevent locking of the wheels of a motor vehicle". The first ABS prototype was tested in 1970, but reliability of the electronics was too low and it was not before 1978 that the first system was put in production and was manufactured by Bosch. Since 1978, ABS technology has been developed further, Figure 1.1 shows that the physical size of the system has been reduced significantly.

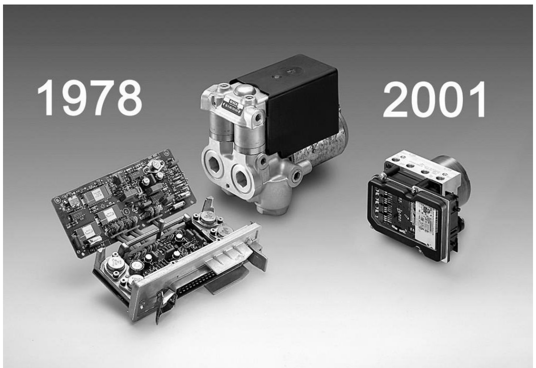


Figure 1.1: 1978 and 2001 ABS unit. The 2001 system is much more compact. Photo: Bosch

Traction control (1985) The functioning of the traction control system is very similar to that of the ABS. The system prevents the wheels from slipping during accelerations by using the same velocity sensors as the ABS. If a vehicle starts to slip, the engine power is reduced in order to maintain lateral control of the vehicle. The first traction control system was launched in 1985 and was also a Bosch system.

Stability control (1995) Again, Bosch was first with their stability control system ESP (Electronic Stability Program) in 1995. While slightly different configurations exist, a stability control system basically measures the yaw rate of the vehicle, i.e., the rotation in the ground plane, and compares with the desired trajectory. If the deviation is greater than some threshold, the system will activate the brake on one side of the vehicle to correct this.

Adaptive cruise control (1998) While sources differ on this, [18] claims that in May 1998, Toyota became the first to introduce an Adaptive Cruise Control (ACC). ACC uses a forward looking sensor, usually radar or laser, to monitor the distance to leading vehicles. If the cruise control is active and time gap to the leading vehicle falls below some threshold, the ACC vehicle will automatically brake in order to maintain the distance.

While ACC is often not considered a safety system in itself, it usually comes bundled with a forward collision warning. In Europe, government restrictions typically limit the allowed braking to 3.0 or 3.5 m/s^2. If the vehicle detects that a higher deceleration is required to avoid colliding with the leading vehicle, an audio warning is given to the driver.

Forward collision mitigation (2003) Forward collision mitigation refers to systems that will try to reduce the impact speed by applying the brakes when a collision with the leading vehicle appears to be unavoidable. While many car manufacturers have announced near term availability of such systems, there are only a few Japanese manufacturers that are currently selling them. Honda has sold a Collision Mitigation System (CMS) since 2003.

Most systems have a similar functionality when it comes to the intervention strategy. They use increasing warning levels as the threat approaches. Hondas system, for example, uses the following technique:

Primary warning When there is a risk of collision with the vehicle ahead or if the distance between the vehicles has become too short, an alarm sounds, and the message "BRAKE" appears on the multi-information display in the instrument panel, prompting the driver to take preventative action.

Secondary warning If the distance between the two vehicles continues to diminish, CMS applies light braking, and the seat belts are retracted gently two or three times, providing the driver with a tactile warning. At this point, if the driver applies the brakes, the system interprets this action as emergency braking, and activates the brake assist function to reduce impact speed.

Collision damage reduction If the system determines that a collision is unavoidable, the seat belt pretensioners are activated with enough force to compensate for seat belt slack or baggy clothing. The CMS also activates the brakes forcefully, approximately 6 m/s^2, to further reduce the speed of impact.

The system is presented in [23]. It has not been revealed how many systems are actually sold, but it was mentioned that customer acceptance of the system has been quite low. For example, it seems that the false alarm rate, especially for aggressive drivers, has been high.

Lane guidance system (2003) Lane guidance system refers to systems that try to help the driver stay in the lane. Systems typically use an acoustic warning or a steering wheel torque to alert the driver if the vehicle is approaching the lane markings. If a steering wheel torque is used, some of the proposed systems will automatically steer the vehicle back into the center of the lane and work almost like an autopilot.

In Japan, Honda has been selling their Honda Intelligent Driver Support (HIDS), which includes the Lane Keeping Assist System (LKAS), since 2003. The system is a combination of an audio warning and a steering wheel torque. However, Honda's idea is that the driver should be kept in the loop at all times. Therefore, the system only supplies 80% of the required torque, the remaining 20% has to be provided by the driver.

Their system has been approved by the Japanese Ministry of Land, Infrastructure and Transport and is allowed on expressways in Japan. See [16] for further details.

The market introduction years for these active safety systems are also illustrated in Figure 1.2.

1.2 Ph.D. project description

The department of Vehicle Dynamics & Active Safety at the Volvo Car Corporation is conducting research in, among other areas, the area of collision

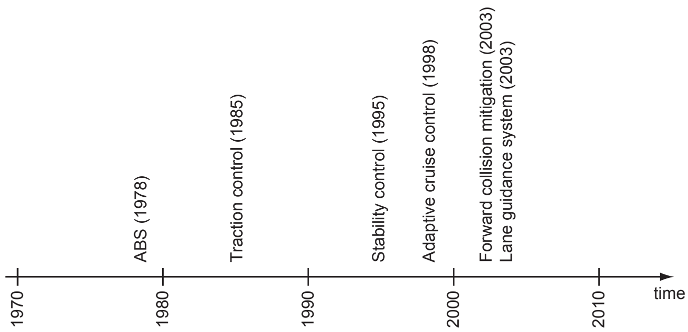


Figure 1.2: Milestones of the development of active safety systems during the last few decades.

warning and avoidance. An important step in the future development of active safety is, of course, answering the question: What is the next big active safety function?

The Ph.D. project CAbS (Collision Avoidance by Steering) was launched in 2002 and will hopefully help in finding the answer to this question. In particular, Volvo want to know if a controllable steering system can be used in collision mitigation or avoidance. The research is conducted in collaboration with the Automatic Control group at the University of Linköping.

1.3 Document outline

First, trying to find the next "big" active safety function, Chapter 2 presents an evaluation method for safety functions which is then applied to a list of candidates. The chosen candidate is then presented in Chapter 3. The chosen active safety function will involve a signal processing unit used for tracking and a decision unit used to determine in which situations the function should intervene. The tracking algorithm, of course, requires information from different sensors, and uses some sort of actuator in order to deploy countermeasures in potentially dangerous situations.

The sensors, actuators and other hardware are presented in Chapter 4. Then, the signal processing unit and the decision unit are presented and evaluated in Chapters 5 and 6 respectively.

After that, in Chapter 7, the simulation environment that was used in the research is presented. Finally, the report ends with conclusions in Chapter 8.

1.4 Contributions

The main contributions of this work are the following:

- The evaluation method for active safety functions in Chapter 2.
- The derivation of the general coordinate transformation in Section 5.2 and the comparison between the different approximations in Section 5.4. Also published in: A. Eidehall and F. Gustafsson. Combined road prediction and target tracking in collision avoidance. In *Proceeding of IEEE Intelligent Vehicles Symposium*, pages 619–624, Parma, Italy, June 2004.
- The chosen active safety function in itself. It includes the function description is presented in Chapter 3, the decision algorithm in Section 6.1 and the combining of the threat assessment and the lateral control algorithm in Section 6.2.
- Implementation and demonstration of the algorithms in a vehicle.

Chapter 2

Evaluation of active safety systems

2.1 Introduction

In this research project, Collision Avoidance by Steering (CAbS) was initially thought of as a system that would make the host vehicle automatically steer away from a potential hazard in cases where a pure braking manoeuvre would be insufficient. However, the problem is currently addressed with a wider perspective. The formulation has changed to: “If we were to design *any* active safety system, which would have the greatest impact on road traffic statistics with regards to system complexity (cost/unit)”.

2.2 Method

The first step in our analysis is to construct a list of potential active safety systems, we are then going to evaluate these potential systems by their utility and their complexity. Their utility is estimated from statistics and their complexity is based on their cost-per-unit. As an additional basis for our decision, a brief competitor analysis and an overview of available technology is presented.

2.2.1 Statistics

Since accident statistics is the base for the system utility calculation, statistical analysis is required. The main disadvantage of most of the currently available statistics is its focus on *passive* safety, i.e., vehicles with similar damages are more likely to be statistically associated than accidents caused by similar circumstances. To obtain relevant information on active safety one has to "read between the lines" and the results always has to be treated with a certain level of suspiciousness.

Rather than projecting accident statistics directly on our potential active safety systems, the statistics is first grouped into relevant types and subtypes. As an example, all lane change accidents are identified as one main type, and then grouped into different subtypes depending on the relative speed and relative position of the vehicles. Other main types are "unintentional lane departure", "intersection accidents" etc. Detailed definitions can be found in Appendix 2.B.

Since the statistics is going to be used for evaluation of active safety systems, we are always interested in the first event in an accident sequence, i.e., a car unintentionally leaving its lane and causing an accident is put in the group of "unintentional lane departure" regardless of what it hit and the type of damage on the vehicle.

We can then evaluate any potential active safety system by adding frequencies of the accident types it would affect. Compared to evaluating specific active safety systems directly from statistics, this method is more flexible since new systems easily can be added and evaluated.

2.2.2 Estimating system complexity

The complexity/cost of a system is based on the cost of its hardware components and a template development cost. The development cost is divided into three different areas:

- Engineering
- Test objects
- Tooling and production equipment

The sum of these costs are then divided by an estimated volume times the expected penetration of the system throughout the product line. These figures are all presented in Section 2.3.2 and are estimated based on the vast experience of the engineers at the department of *Vehicle Dynamics & Active Safety* at Volvo Cars.

2.3 Data analysis

The list of active safety systems that we will evaluate is presented in Appendix 2.A. Some of these concepts have been presented before, but most of them are new and have been developed in creative discussions.

2.3.1 Estimating system utility

In the first step we will use a European traffic accidents database called EACS (European Accident Causation Survey). It contains just under 2,000 accidents and will be used partly to evaluate our own decision strategy, which, if proven to be feasible, could be applied to a much larger German database GIDAS (German In-Depth Accident Study).

First, assume we have defined n different accident types and that these have the statistical frequencies $\mathbf{a} = (a_1, a_2, \ldots, a_n)$. Then the utility of a system j can be estimated by first forming a vector $\mathbf{x}_j = (x_{1j}, x_{2j}, \ldots, x_{nj})$ where $0 \leq x_{ij} \leq 1$ describes the assumed or estimated effect the active safety system j has on accident type i. (If system j has 75% impact on accident type i we would have $x_{ij} = 0.75$.) Then the total system utility can then be defined as

$$u_j = \sum_{i=1}^{n} a_i x_{ij}$$

The accident frequencies from EACS are presented in Table 2.1. Definitions of the accident types can be found in Appendix 2.B. We have created these accident types to reflect important issues in potential active safety systems, but not with a particular active safety function in mind. Note that only the data in type 8x has been divided into its subtypes.

Table 2.2 shows the full utility matrix with the x_{ij}:s and corresponding active safety systems. Empty positions represent the value zero.

Type	Light/no injuries	Severe/fatal accidents
1x	321	94
2x	80	14
3x	60	15
4x	97	24
5x	419	126
6x	25	15
7x	91	30
81	7	3
82	116	114
83	79	39

Table 2.1: Data from the EACS (European Accident Causation Survey) database.

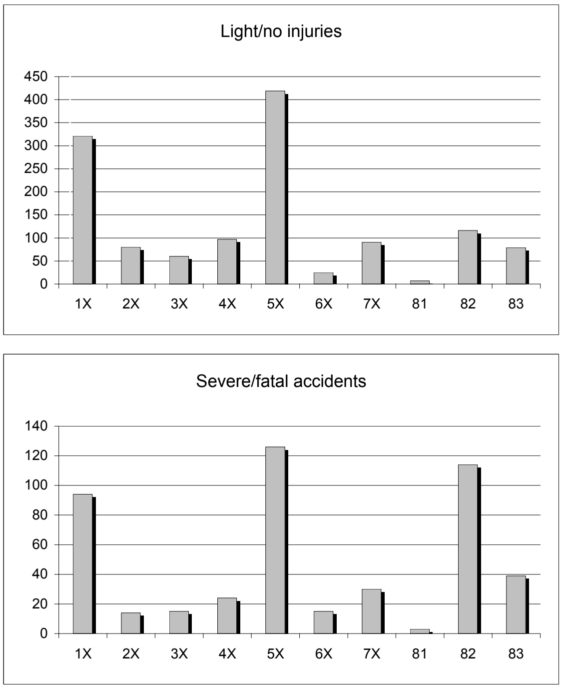


Figure 2.1: Diagram of the data from Table 2.1.

	Accident type 1x	Accident type 2x	Accident type 3x	Accident type 4x	Accident type 5x	Accident type 6x	Accident type 7x	Accident type 81	Accident type 82	Accident type 83	Light/no personal injuries	Fatal/severe accidents
Frequency: Fatal/severe accidents	94	14	15	24	126	15	30	3	114	39		
Frequency: Light/no personal injuries	321	80	60	97	419	25	91	7	116	79		
LKA - Lane Keeping Aid	20%										64,2	18,8
LCA - Lane Change Aid			10%								6	1,5
CMbB	5%	30%	5%			5%					44,3	10,4
CMbB2 (CMbB + RFD)	5%	33%	5%			5%					46,7	10,82
CMbB3 (CMbB2 + RFD + Driver monitoring)	5%	40%	5%			5%					52,3	11,8
CW - Curvature Warning				15%							14,55	3,6
LAS1 - Lane Assist System (LKA + LCA)	20%		10%								70,2	20,3
LAS2 (LAS1 + traffic outside blind spot)	20%		25%								79,2	22,55
ELA (LAS2 + traffic in opposite direction)	30%		25%								111,3	31,95
GWW - Give Way Warning					5%						20,95	6,3
GS - Game Scanner								10%			0,7	0,3
PBS - Pedestrian & Bicycle scanner									25%	25%	48,75	38,25
OGS - Overtaking Guidance System						10%					2,5	1,5
True CAbS - Collision Avoidance by Steering	30%	40%	25%			10%					145,8	39,05
True CAbS2 (True CAbS + RFD + Driver mon.)	30%	50%	25%			10%					153,8	40,45

Table 2.2: Utility matrix with the assumed impact on the different accident types and the resulting utility value of each system.

2.3.2 Estimating system complexity

System complexity can be estimated similarly to the way the system utility was estimated, with the difference that system development cost also needs to be considered.

This time we start with a set of m hardware components with a cost vector $\mathbf{b} = (b_1, b_2, \ldots, b_m)$. Next, for system j, we construct the vector $\mathbf{y}_j = (y_{1j}, y_{2j}, \ldots, y_{2m})$, where $y_{kj} = 1$ if system j involves hardware component k, $y_{kj} = 0$ if not. The component cost c_j^C of system j can then be calculated as

$$c_j^C = \sum_{i=1}^{n} b_i y_{ij}$$

The hardware components and their costs are presented in Table 2.3.

Component	Assumed cost per unit (USD)
Laser	300
Front view camera	200
Side mirror camera	300
Infrared camera	700
RTI (GPS navigation system)	250
Steering wheel sensor	0
Gyro	0
Road friction detector	2
Driver monitoring system	150
Brake actuator	0
Steering actuator	0

Table 2.3: Cost of the hardware components.

As was mentioned in Section 2.2.2, the development cost can be estimated as a combination of the assumed *engineering*, *test objects* and *tooling and production equipment* costs. For system j, these are labelled c_j^E, c_j^T and c_j^P respectively. Now, if the total volume is V and p_j denotes the amount of these vehicles system j is installed in (p_j is usually referred to as *penetration* or *take rate*), then the development cost c_j^D per unit can be calculated as

$$c_j^D = \frac{c_j^E + c_j^T + c_j^T}{p_j V}$$

Then we get the total cost per unit as

$$c_j = c_j^D + c_j^C$$

Component and development costs are put together in Table 2.4. Again, empty positions represent the value zero.

Cost per unit (USD)	300	450	200	300	700	250	2	150	0	0	Development cost (MUSD)						
	Hardware components																
	Laser	Wide Laser	Front view camera	Side mirror camera	Infrared camera	RTI - GPS	Road friction detector	Driver monitoring	Braking actuator	Steering actuator	Engineering	Test objects	Tooling and production equipment	Penetration	Component cost/unit (USD)	Development cost/unit (USD)	Total cust/unit (USD)
LKA - Lane Keeping Aid			1					1		1	2	0,3	1	10%	350	66	416
LCA - Lane Change Aid				1							1,5	0,3	1,5	10%	300	66	366
CMbB	1		1						1		4	0,5	1,5	100%	500	12	512
CMbB2 (CMbB + RFD)	1		1				1		1		4,5	0,5	1,5	100%	502	13	515
CMbB3 (CMbB2 + RFD + Driver monitoring)	1		1				1	1	1		6	0,7	2	20%	652	87	739
CW - Curvature Warning						1	1				1	0,3	0,5	10%	252	36	288
LAS1 - Lane Assist System (LKA + LCA)			1	1							4	0,6	2,5	10%	500	142	642
LAS2 (LAS1 + traffic outside blind spot)		1	1	1							5,5	1,5	3,5	10%	950	210	1160
ELA (LAS2 + traffic in opposite direction)		1	1	1						1	6,5	1,5	3,5	10%	950	230	1180
GWW - Give Way Warning			1			1	1		1		1,5	0,6	1	5%	452	124	576
GS - Game Scanner			1		1		1		1		2,5	1,5	2	10%	902	120	1022
PBS - Pedestrian & Bicycle scanner		1	1						1		3	0,4	2	100%	650	10,8	660,8
OGS - Overtaking Guidance System	1		1	1		1					3	0,8	3	10%	1050	136	1186
True CAbS - Collision Avoidance by Steering		1	1	1					1	1	11,5	2	4	10%	950	350	1300
True CAbS2 (True CAbS + RFD + Driver mon.)		1	1	1			1	1	1	1	13,5	2,2	4,5	10%	1102	404	1506

Volumes	
Year 1	100000
Year 2	200000
Year 3	200000
	500000

Table 2.4: Cost matrix

2.3.3 Results

In Table 2.5 we compute, for each system, the ratio u_j/c_j, utility/cost. This is a way to compare systems of different complexity and fields of application. The target function u_j/c_j seems plausible, although other functions, such as u_j^2/c_j could also be studied.

In Table 2.5 the target function is calculated in two columns, light/no injuries in the first and severe/fatal accidents in the second. Also, to indicate potential candidates, the values higher than half of the highest value of that column are marked with arrows.

	Total cost/unit	Utility (light/no injuries)	**Ratio (100*utility/cost)**	Utility (severe/fatal accidents)	**Ratio (100*utility/cost)**
LKA - Lane Keeping Aid	416	64,2	**15,43** ←	18,8	**4,52** ←
LCA - Lane Change Aid	366	6	**1,64**	1,5	**0,41**
CMbB	512	44,3	**8,65** ←	10,4	**2,03**
CMbB2 (CMbB + RFD)	515	46,7	**9,07** ←	10,82	**2,10**
CMbB3 (CMbB2 + RFD + Driver monitoring)	739	52,3	**7,08**	11,8	**1,60**
CW - Curvature Warning	288	14,55	**5,05**	3,6	**1,25**
LAS1 - Lane Assist System (LKA + LCA)	642	70,2	**10,93** ←	20,3	**3,16** ←
LAS2 (LAS1 + traffic outside blind spot)	1160	79,2	**6,83**	22,55	**1,94**
ELA (LAS2 + traffic in opposite direction)	1180	111,3	**9,43** ←	31,95	**2,71**
GWW - Give Way Warning	576	20,95	**3,64**	6,3	**1,09**
GS - Game Scanner	1022	0,7	**0,07**	0,3	**0,03**
PBS - Pedestrian & Bicycle scanner	660,8	48,75	**7,38**	38,25	**5,79** ←
OGS - Overtaking Guidance System	1186	2,5	**0,21**	1,5	**0,13**
True CAbS - Collision Avoidance by Steering	1300	145,8	**11,22** ←	39,05	**3,00** ←
True CAbS2 (True CAbS + RFD + Driver mon.)	1506	153,8	**10,21** ←	40,45	**2,69**

Table 2.5: Decision matrix. Values higher than half of the maximum value have been marked with arrows.

2.4 Conclusions

Inspecting the decision matrix (Table 2.5) shows that the following choices have a high target function value.

- Collision Mitigation by Braking (CMbB)
- Emergency Lane Assist (ELA)
- Pedestrian & Bicycle Scanner
- True Collision Avoidance by Steering (True CAbS)

The system True CAbS is based on ELA and CMbB. Given the fact that CMbB is already being developed, this suggests that going via ELA towards True CAbS seems to be reasonable direction for our future work.

2.4.1 Future trends

This section will try to give an idea about what other research groups are doing by presenting some of the latest applications in the area of automotive safety. Many of these ideas have been presented at various conferences but none of them has yet reached the market.

ACC with GPS In addition to the Adaptive Cruise Control system explained in Section 1.1, which is able to adjust the speed to a leading vehicle, future ACC systems are believed to also being able to slow down in sharp curves. The most common approach for achieving this is by using a GPS satellite positioning system together with a digital road map which provides information about the curvature approaching curves.

Pedestrian detection Pedestrian detection is a major research area. Image processing is used to analyze data from a camera, usually standard video, but sometimes also infra-red, and if a pedestrian is detected in a dangerous location, the system typically brakes the vehicle automatically. Examples of pedestrian detection systems are [11] from DaimlerChrysler and [13] from Volvo Technology Corporation, both presented at the 2004 Intelligent Vehicles Symposium.

Interior sensing It is a well known fact that many accidents occur due to driver drowsiness or driver inattention. By mounting cameras inside the vehicle looking at the driver, different parameters related to drowsiness or distraction can be monitored. For drowsiness, the most well established and promising measure is PERCLOS [28] which averages eyelid closure over time. The cameras can also monitor gaze direction and thereby detect when the driver is not looking at the road.

The information can be used, either to recommend the driver to rest or to adjust different thresholds in other active safety functions. For example, we could allow a collision mitigation system to brake earlier if the driver is not focusing on the road, since it is less likely that he or she will steer away in the last moment.

Pre-crash safety Much effort is put into the area of preparing the vehicle for an imminent collision. For example, DaimlerChrysler has announced that they are experimenting with extending bumpers that would increase the deformation zone and also with active interior components, for example extending door panels, that pushes the driver away from the side door just before the collision [5].

Night vision Many car manufacturers, for example Volvo Car Corporation, have been working on night vision systems, an example is shown in Figure 2.2. The systems are based on an infra-red camera mounted in the front of the vehicle from which the image is then projected onto a screen in front of the driver. An infra-red night vision system has been reported to allow up to five times the viewing distance in darkness.

Figure 2.2: The night vision system of the Volvo Safety Concept Car. Photo: Volvo Car Corporation.

2.A Potential active safety functions

This is a very brief description of the active safety functions evaluated in Chapter 2. Some of these systems have been presented elsewhere, but most of them are new.

Lane Keeping Aid Includes a vision system for lane detection and uses a steering wheel actuator to keep the vehicle in the lane at all times [24].

Lane Change Aid Assists during lane changes by activating a warning light if another vehicle is driving in the blind spot. This system is currently sold by Volvo Cars, see the web page www.volvocars.com for details.

Collision Mitigation by Braking Automatic braking when forward collision is unavoidable [17].

Collision Mitigation by Braking 2 Same as above but also includes road friction detector to be able to brake earlier during bad road conditions.

Collision Mitigation by Braking 3 Collision Mitigation by Braking with friction detection and a system for detecting drowsy or distracted drivers to be able to brake even earlier.

Curvature Warning A warning is activated if the host vehicle is approaching a sharp curve too fast.

Lane Assist Systems Combination of Lane Keeping Aid and Lane Change Aid. Does not use steering wheel actuator, only a sound warning.

Lane Assist System 2 Same as above, but also monitoring traffic which is not necessarily in the blind spot but still poses a threat. Audio warning on a dangerous lane change.

Emergency Lane Assist Same as Lane Assist System 2 but also monitoring oncoming traffic. This system uses a steering wheel actuator to prevent dangerous lane departure.

Give Way Warning Warning if host vehicle is approaching an intersection with for example a red light or stop sign too fast.

Game scanner Game warning system. Warning if a big animal is entering the road ahead of the host vehicle.

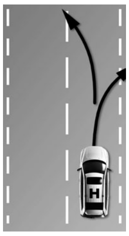
Pedestrian & Bicycle scanner Searching for pedestrians/bicycles in front of the vehicle [13, 11]. Automatic braking.

Overtaking Guidance System Warning the driver for dangerous overtaking situations and informing about coming overtaking opportunities.

True Collision Avoidance by Steering System which is actually steering away when a pure braking manoeuver is insufficient. Includes Emergency Lane Assist + Collision Mitigation by Braking 3.

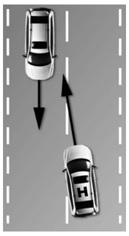
2.B Accident types

These are the accident types and subtypes. As was mentioned in Section 2.3.1, these accident types were created to reflect important issues in potential active safety systems, but not with a particular active safety function in mind. Note that the vehicle marked with a "H" in each scenario refers to the host vehicle which is equipped with the active safety function. Each row consist of the main type and its subtypes, e.g., 11, 12, 13 are subtypes of 1x.



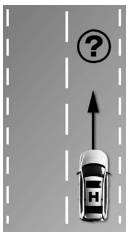
Type 1x
Unintentional lane departure

Type 11
Running off road

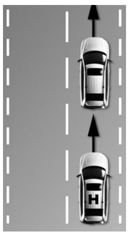


Type 12
Collision with vehicle in opposite direction

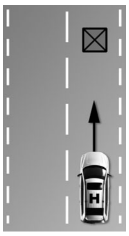
Type 13
Collision with infrastructure



Type 2x
Collision with object in same lane



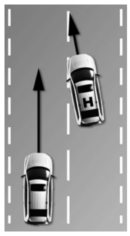
Type 21
Vehicle in same lane



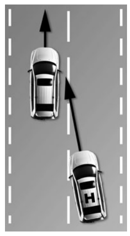
Type 22
Other object in same lane

Type 3x
Lane change
accident

Type 31
Blind spot
related accident

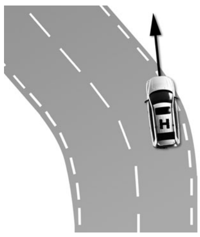


Type 32
Collision with
faster moving
vehicle



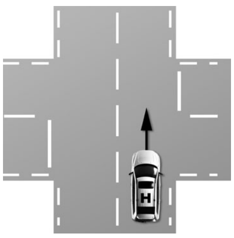
Type 33
Collision with
slower moving
vehicle

Type 4x
Loss of grip

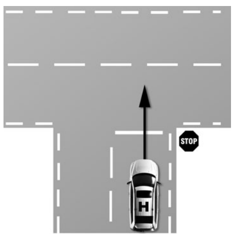


Type 41
Entering curve
too fast

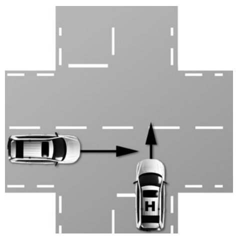
Type 42
Loss of grip due
to ice, snow etc.



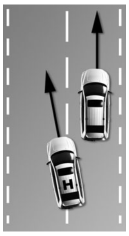
Type 5x
Accident in
intersection



Type 51
Approaching
give-way situation
too fast

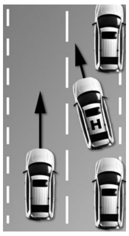


Type 52
Other accident in
intersection



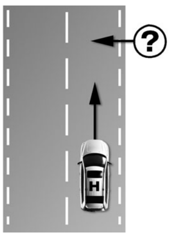
Type 6x
Overtaking in two-way traffic

Type 7x
Miscellaneous, parking, backing etc.

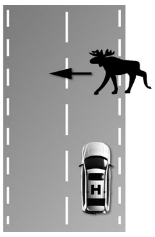


Type 71
Leaving car park

Type 72
Other accidents related to parking, backing etc.

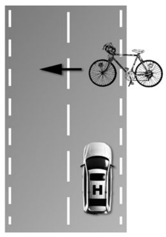


Type 8x
Obstacle enters lane



Type 81
Game

Type 82
Pedestrian



Type 82
Bicyclist

Chapter 3

Function description

3.1 Description

Many lane guidance systems have been proposed in the recent years. Some of them use a buzzer and some of them use a steering wheel torque to indicate or prevent lane departure. There are two major problems with that kind of systems. The first is false alarms when changing lane intentionally. It is often claimed that this can be solved by disabling the interventions when the indicator is used, but studies have shown that people generally do not use the indicators at every lane change. Also, a very common behavior is to cross the lane marking slightly on the inside of curves, usually referred to as "curve cutting". The second problem is misuse. A system that applies a steering wheel torque in order to keep the vehicle in the lane can almost be used as an autopilot. Typically, the driver could rely on the system totally for short periods of time while carrying out distractive tasks like changing CDs or writing text messages, which would clearly be a very precarious situation.

Honda has a proposed solution in [16], which was also discussed in Section 1.1, were they only apply 80% of the required torque to keep the vehicle in the lane. This is to keep the driver in the loop at all times. The problem is that if the driver is actually not in the loop, i.e., is distracted or misjudging the situation, the system will not prevent the lane departure. Their studies certainly showed that people found the vehicle more stable and easy to steer, this makes the system more of a convenience system than a safety system.

Another possible solution is to combine the lane guidance system with some sort of driver monitoring device. Clearly, if the system could be activated only when the driver is distracted or drowsy, this would reduce the number of false alarms. As driver monitoring systems improve, this could certainly become an interesting combination.

Emergency Lane Assist (ELA) provides another alternative of eliminating false alarms and misuse. The function will only try to prevent *dangerous* lane departure. The system monitors adjacent lanes and as long as there are no other vehicles approaching, the lane markings can be crossed without ELA intervention, but as soon as a commenced lane change manoeuvre is considered dangerous with respect to, for example an oncoming vehicles, a torque is applied to the steering wheel in order to prevent lane departure. The risk level of a lane change manoeuvre is judged based on the position and motion of vehicles in the adjacent lanes, but also road edges and barriers or even solid lane markings could be used to activate the intervention.

This approach makes ELA a pure safety system rather than a comfort/convenience system. Figure 3.1 shows critical ELA situations.

Prerequisites

The function must never prevent evasive action, i.e., if it is assumed that the driver is departing the lane due to some obstacle in front of the vehicle. First of all, evasive action can be detected by searching for threats in front of the vehicle. If there is a risk of collision with a leading vehicle, a lane departure manoeuver should never be prevented since it would lead the host vehicle towards this threat. Evasive action could also be detected by measuring the strength and speed of the steering manoeuvre. A powerful and fast steering manoeuvre could be interpreted as if the driver is taking evasive action.

In cases where an undesired intervention has taken place, it must always be possible to override the system through a resolute steering manoeuvre.

3.2 Technical function overview

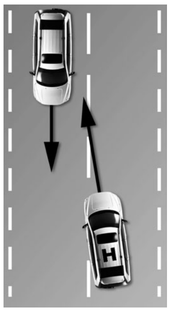
It is clear ELA requires information about the road geometry, lane markings and other vehicles, both in front of and behind the host vehicle. It will also require some sort of controllable steering system.

In this research project, we did have access to a camera and a radar, these were chosen as forward looking sensors. The radar detects leading vehicles by processing radar reflexes and the vision detects both vehicles and lane geometry by using image processing. These two sensors also came with a basic sensor fusion unit for objects.

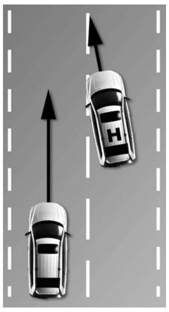
At this point, no rearward looking sensors were available, and thus a somewhat restricted ELA which only regards objects in front of the host vehicle had to be studied.

At an early stage it was noticed that the lane geometry estimate, in particular the curvature, needed to be improved in some way. This is done in a Kalman Filter based tracking system. Another big part of the ELA function is, of course, a decision and intervention unit. A schematic overview of how these units are connected are shown in Figure 3.2. It also shows in which chapters of the thesis these different parts are explored.

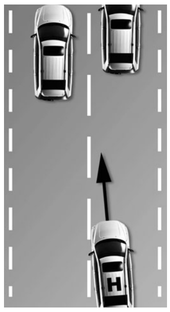
No ELA intervention since there is no threat in the adjacent lane.



ELA intervention.



ELA intervention.



No ELA intervention since there is a threat also in the own lane.

Figure 3.1: Critical ELA situations. The letter "H" indicate the ELA host vehicle.

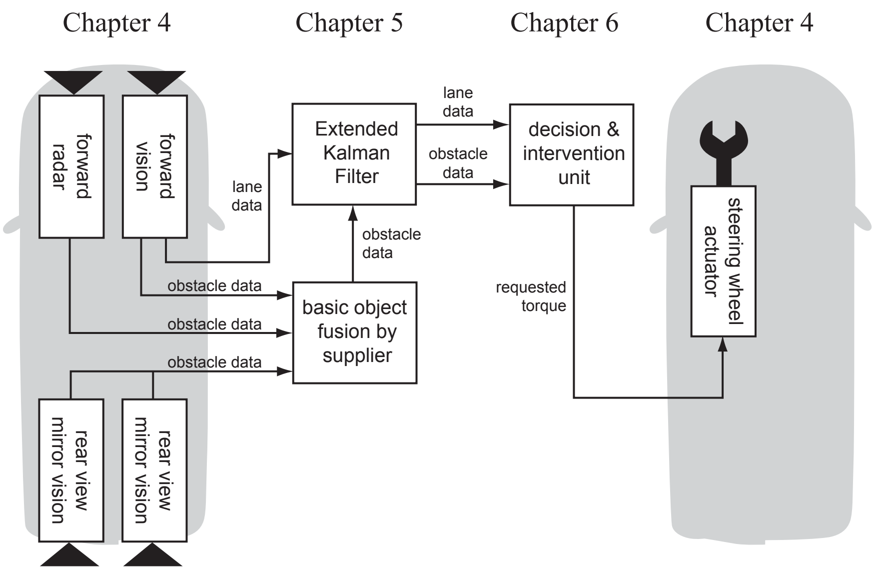


Figure 3.2: ELA system overview and connections to the chapters in the thesis. Chapter 4 discusses the sensors and actuators of the vehicle, details about the Extended Kalman Filter based tracking system are given in Chapter 5 and the decision and intervention strategies are explained in Chapter 6. Note that the rear view mirror cameras were not used in our demonstrator.

Chapter 4

Hardware

This chapter will give some details about the electronic hardware that has been used in the ELA development, such as sensors, actuators, onboard computers and computer networks, see Figure 3.2 on page 27.

4.1 Sensors

4.1.1 Radar

The radar that was used is a 77GHz unit which is placed in the grill of the vehicle, as shown in Figure 4.1. The approximate range is 150 meters and about 95% of the measurements end up within 20 to 30 centimeters from the correct value. The radar also uses doppler shift analysis to determine range rate, i.e., relative velocity, with a very high accuracy.

The field of view of the radar is 15° and it has an update frequency of 10Hz. The radar unit is equipped with its own signal processing unit with a tracking system and for each object, the radar outputs range, range rate and azimuth angle, but also information about how long the object has been visible.

One problem with the radar is that other objects than vehicles, such as road humps or road barriers, also give radar reflexes. In particular, there is a sampling phenomenon when driving next to a road barrier and the host vehicle move from one pole to the next with about the same sampling time as the radar is using. This will give the impression of a vehicle driving just in front of the host vehicle.

Figure 4.1: The radar sensor.

4.1.2 Vision

The vision system is based on a single, black and white CCD camera mounted next to the center rear view mirror, as shown in Figure 4.2. The camera has a resolution of 640 times 480 pixels and is connected to an image processing unit which uses advanced pattern recognition to find other vehicles, bicycles, or even pedestrians, in the images.

The typical range at which vehicles are detected is about 60 - 70 meters, but the accuracy is lower than the radar. The azimuth resolution, on the other hand, is much better than the radar, which is why a vision system is a very good complement to a radar sensor.

The image processing unit is also able to classify objects into cars, trucks, motorcycles and pedestrians.

Another important use of the vision system is lane tracking. Distances to the lane markings, plus some additional road geometry parameters like heading angle and curvature, are determined by the vision algorithms. The lane marking distances and the heading angle measurements are very robust during bad weather conditions or worn lane markings. The curvature measurement, on the other hand, can be quite unreliable, especially during rain or fog. The system does, however, provide a quality signal which indicates the confidence of the road geometry measurements.

The sampling time of the vision system is about 0.1 seconds, but it can raise to around 0.14 seconds when the scene is very complex, for example

Figure 4.2: The vision sensor.

in city traffic.

4.1.3 Sensor fusion

Since the radar has range measurements with high accuracy and the vision system has azimuth angle measurement with high accuracy, the two sensors are a perfect combination when it comes to fusing the data.

Another important benefit from combining the two sensors is the pattern recognition in the image processing system. It can be used to rule out radar reflexes that do not originate from vehicles or pedestrians, for example reflexes from road humps or road barriers as it was described above.

A sensor fusion unit for basic fusion of objects from the vision with tracks from the radar comes with the sensors.

4.2 Steering wheel actuator

It would of course have been convenient to use a vehicle equipped with Electric Power Assist Steering (EPAS). Then the desired torque could just have been added to the normal steering assist torque. The Volvo V70 we used, however, is equipped with a standard hydraulic steering system which is why an additional electric motor had to been mounted. This electric motor, often referred to as a Haptic Steering Column or HapCol, sits on the steering column, just behind the steering wheel, and is able to deliver up to 17 Nm of torque. However, for safety reasons it is restricted in software to 7 Nm.

The steering wheel actuator is connected to the CAN bus, see Section 4.3.1 and also comes with a steering wheel angle sensor.

4.3 Electronics

4.3.1 CAN bus

The Controller Area Network (CAN) bus has come to dominate the automotive industry in Europe, and U.S. manufacturers are starting to adopt it. Using vehicles with a CAN bus is a great advantage when it comes to building test cars since an on board computer connected to the bus immediately can communicate with all nodes in the vehicle without any extra wiring.

A CAN network communicates with messages of up to 8 bytes plus a Cyclic Redundancy Check (CRC) and an identifier. All nodes on the network receive each message and then decide whether that identifier value is of interest.

Choosing a CAN controller defines the physical and data-link portions of the Open Systems Interconnect (OSI) protocol stack, see Figure 4.3. This means that it involves basic error checking, acknowledgements and retransmitting.

There are two versions of the physical protocol layer, low speed CAN of 5 kbit/s and high speed CAN of 1 Mbit/s. Low speed CAN is typically used for communicating with lights, mirrors and seat control whereas high speed CAN is used, for example, for engine control and ABS.

We have been communicating with the sensors on a dedicated high speed CAN bus and with the steering wheel actuator over the standard vehicle high speed CAN bus.

4.3.2 dSpace AutoBox

For real time code execution, a dSPACE AutoBox equipped with a PowerPC 750 FX board running at 480MHz and a CAN controller was used. The real time code was generated using Real Time Workshop which is a Matlab compiler that generates machine code based on a Simulink model.

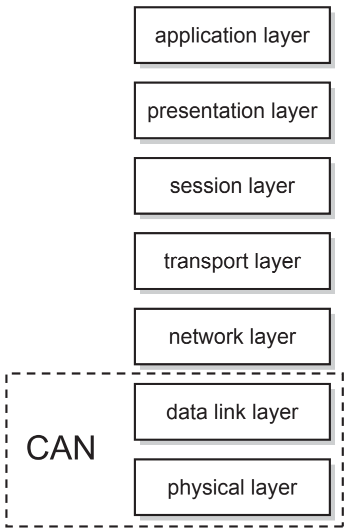


Figure 4.3: For those familiar with the Open Systems Interconnect 7-layer ISO standard, CAN specifies the bottom two layers. This means that it involves basic error checking, acknowledgements and retransmitting.

Chapter 5

Tracking system

This chapter will explain the tracking system, see Figure 3.2 on page 27.

5.1 Introduction

Active safety technology, such as the Emergency Lane Assist system described in Chapter 3, will require detailed knowledge about the vehicle surroundings. In this chapter, vehicle surroundings will refer to lane geometry and other vehicles. Typically, lane information is obtained from a vision system and other vehicles are detected with vision and radar.

The importance of integrating data from object tracking and road geometry tracking has quite recently been recognized [8, 2, 29, 25]. The main idea is to try to improve the road geometry estimate by studying the motion of other vehicles and vice versa. For example, if a couple of tracked vehicles suddenly all start moving right, one of two things can have happened. The first is that they all started a lane change manoeuvre and the road remains straight. The other is that we are entering a curve and the vehicles are still following the center of their lanes. These possibilities can be treated in a Bayesian framework, together with the information from the lane tracker, to build a new estimator. In order to do this we need to construct a new object measurement equation based on the road geometry. The derivation of this, presented in Section 5.2 and the evaluation in Section 5.4 have previously been published by the author in [10].

Typically, this geometric model is derived by introducing a road aligned, curved coordinate system, which also brings a couple of other interesting features. First of all, the motion model of other vehicles can be simplified. The assumption that they will stay in their lane can simply be expressed as $\dot{y} = 0$ where y denotes lateral position in the road coordinates. In a Cartesian or polar coordinate system, a higher order system would have to be used and would still only describe a more primitive shape, see Figure 5.1.

This also relates to predicting future positions of other vehicles. With the simple motion model, vehicles are predicted to follow straight lines, which in the curved coordinate system means that they will follow the road. Prediction with a higher order model can be difficult since many people tend to wobble slightly when driving. This means that a prediction, say 50 or 100 meters ahead will often be outside the road. This is illustrated to the right in Figure 5.1.

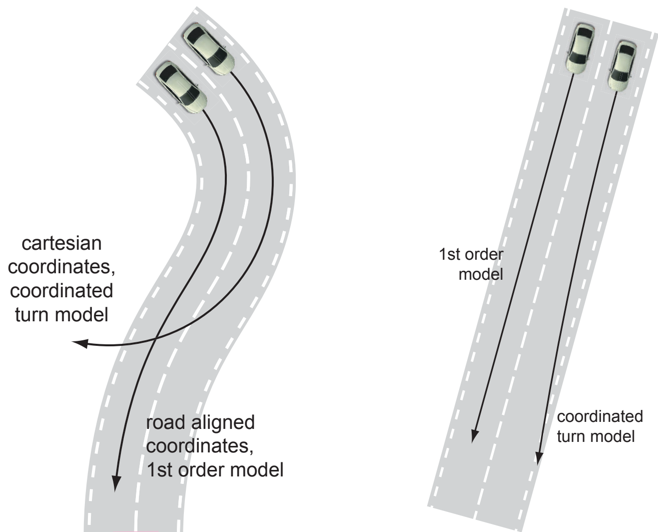


Figure 5.1: Left: Comparison between a road aligned coordinate system with a low order model and a Cartesian coordinate system with a higher order model. In order to illustrate this difference, the curvature and clothoid parameters of this road has been exaggerated. Right: Prediction with a higher order model can be difficult since people tend to wobble slightly when driving.

Also, since all positions are already given in the road coordinates, it is easier to build different automotive applications. For example, if an Adaptive Cruise Control would want to know the distance to the leading vehicle, it could simply look at vehicles with y-coordinate close to zero. They will be in the same lane as the host vehicle, even if the road is turning.

5.1.1 Overview

The position on the road for each vehicle is denoted (x, y). This means that x is the driven distance along the road and y the lateral position in the lane. To relate this to what the sensors on the vehicle will see, we need to relate this to a Cartesian coordinate system attached to the vehicle, which will be denoted $(\tilde{x}, \tilde{y})$. The purpose of Section 5.2.1 is to find such a transformation $T : (x, y) \longrightarrow (\tilde{x}, \tilde{y})$. To derive this, we need a model of the road, and we start with a general model describing the road curvature as $c(x) = c_0 + c_1 x$. This describes a clothoid curve and is a commonly used parametrization in collision avoidance applications [22, 29, 25]. The trigonometric formulas that arise do not give an explicit expression for $T(x, y)$. If such an expression is needed, either the trigonometric functions can be Taylor expanded, or a simpler model with $c_1 = 0$ can be used (constant curve radius). These approximations are treated in Section 5.2.3. To be able to apply a Kalman filter, we first define a state vector that contains road geometry, the host vehicle's and the tracked vehicles' positions. Section 5.2.4 gives the measurement equations, where the host vehicle's sensors are expressed as functions of the state vector. A suitable motion model for the host vehicle and the tracked vehicles is suggested in Section 5.2.4.

5.1.2 Preliminary definitions

In order to derive the measurement equation, a certain amount of mathematics and geometry will be required. First of all, if $\mathbf{v}$ is a vector with components $v_1, v_2, \ldots, v_n$, then the derivative of this vector is defined as the vector consisting of the derivatives of these components, i.e.,

$$\frac{d\mathbf{v}}{dx} = \begin{pmatrix} dv_1/dx \\ dv_2/dx \\ \vdots \\ dv_n/dx \end{pmatrix}$$

Similarly, the integral of a vector is defined as the vector consisting of the integrals of the vector components. We will also use the exponential function for matrices, which is defined as

$$\exp(A) = \sum_{k=0}^{\infty} \frac{A^k}{k!}$$

The two dimensional rotational matrix will frequently be used. It is used to rotate vectors α radians in the coordinate plane:

$$R(\alpha) \triangleq \begin{pmatrix} \cos\alpha & -\sin\alpha \\ \sin\alpha & \cos\alpha \end{pmatrix}$$

5.2 Model derivation

5.2.1 Coordinate system derivation

We will start by deriving a two dimensional coordinate transformation which is a mapping T from a curved coordinate system (x, y) which follows the road to a cartesian coordinate system $(\tilde{x}, \tilde{y})$ which is attached to the host vehicle, see Figure 5.2. The first part of the derivation is similar to what was presented in [9].

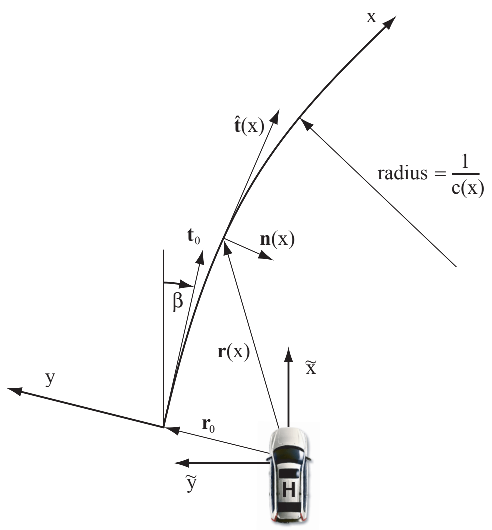


Figure 5.2: Vectors used in the derivation of T. $\mathbf{r}(x)$ describes the $y = 0$ curve in the (x, y)-coordinates. $\hat{\mathbf{t}}(x)$ is the tangent to $\mathbf{r}(x)$.

We start with a planar curve $\mathbf{r}(x)$, where x is distance along a curve. Assume the curvature along the curve is given by $c = c(x)$. Now, if $\hat{\mathbf{t}}(x)$ is the tangent vector we define the normal vector $\mathbf{n}(x)$ as

$$\mathbf{n}(x) = \frac{d\hat{\mathbf{t}}}{dx}(x)$$

From vector analysis we get an alternative expression for $\mathbf{n}(x)$ where we use the fact that it is perpendicular to $\hat{\mathbf{t}}(x)$ and that its length is precisely $c(x)$. This can be expressed using the rotational matrix as $\mathbf{n}(x) = c(x)R(-\frac{\pi}{2})\hat{\mathbf{t}}(x)$. Thus, we end up with the differential equation

$$\frac{d\hat{\mathbf{t}}}{dx}(x) = c(x)R(-\frac{\pi}{2})\hat{\mathbf{t}}(x)$$

which has the solution

$$\hat{\mathbf{t}}(x) = \exp(R(-\frac{\pi}{2})\int_0^x c(\tau)d\tau)\hat{\mathbf{t}}_0 = R(-\int_0^x c(\tau)d\tau)\hat{\mathbf{t}}_0 \tag{5.1}$$

The last equality is proved in Appendix 5.A. This can then be integrated to obtain an expression for the position:

$$\begin{aligned}\mathbf{r}(x) &= \int_0^x \hat{\mathbf{t}}(\tau)d\tau + \mathbf{r}_0 = \\ &= \int_0^x R(-\int_0^{\tau_1} c(\tau_2)d\tau_2)d\tau_1\hat{\mathbf{t}}_0 + \mathbf{r}_0 \end{aligned} \tag{5.2}$$

The vectors $\mathbf{r}(x)$ and $\hat{\mathbf{t}}(x)$ are illustrated in Figure 5.2.

To construct the coordinate transformation T we first define one of our coordinates to be the distance along the curve: $T(x, 0) = \mathbf{r}(x)$. To define the other coordinate we require T to be orthogonal, i.e.,

$$\frac{dT(x,y)}{dy} \perp \hat{\mathbf{t}}(x) \quad \text{for all} \quad x$$

A natural choice is to simply extend a straight line at $T(x, 0)$ along $-R(\frac{\pi}{2})\hat{\mathbf{t}}(x)$, a vector that is orthogonal to $\hat{\mathbf{t}}(x)$ for all x. This choice will also give us a positively oriented transformation. T will look like

$$\begin{pmatrix}\tilde{x}\\ \tilde{y}\end{pmatrix} = T(x,y) = \mathbf{r}(x) - R(\frac{\pi}{2})\hat{\mathbf{t}}(x)y \tag{5.3}$$

5.2.2 Choosing curvature function

The road is often modelled as segments of straight lines, arcs and clothoids, see for example [9]. Clothoids are segments were the curvature changes linearly with the distance along the curve. According to [1], this agrees well with how roads are constructed. The function

$$c(x) = c_0 + c_1 x \tag{5.4}$$

will suffice for all these cases. Of course, when approaching a curve we might, for example, have situations were the section 0 - 50 meters of our field of view is a straight line and the section 50 - 100 meters is a clothoid, a case which can not be modelled with a linear curvature law.

Plugging (5.4) into (5.1) and using

$$\hat{\mathbf{t}}_0 = \begin{pmatrix} \cos\beta \\ \sin\beta \end{pmatrix}$$

we get

$$\begin{aligned}
\hat{\mathbf{t}}(x) &= R(-c_0x - c_1x^2/2)\begin{pmatrix} \cos\beta \\ \sin\beta \end{pmatrix} = \\
&= \begin{pmatrix} \cos(c_0x + c_1x^2/2) & \sin(c_0x + c_1x^2/2) \\ -\sin(c_0x + c_1x^2/2) & \cos(c_0x + c_1x^2/2) \end{pmatrix}\begin{pmatrix} \cos\beta \\ \sin\beta \end{pmatrix} \\
&= \begin{pmatrix} \cos\beta\cos(c_0x + c_1x^2/2) + \sin\beta\sin(c_0x + c_1x^2/2) \\ -\cos\beta\sin(c_0x + c_1x^2/2) + \sin\beta\cos(c_0x + c_1x^2/2) \end{pmatrix} \\
&= \begin{pmatrix} \cos\beta & -\sin\beta \\ \sin\beta & \cos\beta \end{pmatrix}\begin{pmatrix} \cos(c_0x + c_1x^2) \\ -\sin(c_0x + c_1x^2) \end{pmatrix} \\
&= R(\beta)\begin{pmatrix} \cos(c_0x + c_1x^2) \\ -\sin(c_0x + c_1x^2) \end{pmatrix}
\end{aligned} \tag{5.5}$$

(5.5) gives the second term of (5.3). In order to get an expression for the first term we need to integrate (5.5), which can not be done analytically. Instead we need to do some sort of approximation.

Before we continue we shall use the free parameter $\mathbf{r}_0$ to describe an offset perpendicular to $\hat{\mathbf{t}}_0$ by simply setting

$$\mathbf{r}_0 = y_{off}R(-\frac{\pi}{2})\mathbf{t}_0 = y_{off}\begin{pmatrix} 0 & 1 \\ -1 & 0 \end{pmatrix}\begin{pmatrix} \cos\beta \\ \sin\beta \end{pmatrix} = R(\beta)\begin{pmatrix} 0 \\ -y_{off} \end{pmatrix}$$

The vector $\mathbf{r}_0$ is shown in Figure 5.2.

5.2.3 Approximations

If we try to use (5.5) in the coordinate transforation, it can not be expressed on closed form. A closed form expression is desirable since in the Kalman filter, we will need to evaluate it and its partial derivatives very frequently. Using the exact expression would involve numeric integration of (5.5) which might be an option, but this have not been investigated so far.

In this section, three different approximations of (5.2) will be derived. While the the first two, Approximation A and B, does not appear in the literature for lane tracking, the last one, Approximation C, is a commonly used expression [22, 29, 25].

Approximation A: Omitting the clothoid parameter

Using $c_1 = 0$ in (5.5) we get

$$\hat{\mathbf{t}}(x) = R(\beta)\begin{pmatrix} \cos(c_0 x) \\ -\sin(c_0 x) \end{pmatrix}$$

which, used in (5.2) gives

$$\mathbf{r}(x) = R(\beta)\begin{pmatrix} \sin(c_0 x) \\ \cos(c_0 x) - 1 \end{pmatrix}\frac{1}{c_0} - R(\beta)\begin{pmatrix} 0 \\ y_{off} \end{pmatrix}$$

and, since rotations commute, the coordinate transformation (5.3) becomes

$$\begin{aligned} T_a(x, y) &\triangleq \mathbf{r}(x) - R(\frac{\pi}{2})\mathbf{t}(x)y = \\ &= R(\beta)\begin{pmatrix} (1 + c_0 y)\sin(c_0 x) \\ (1 + c_0 y)\cos(c_0 x) - 1 - c_0 y_{off} \end{pmatrix}\frac{1}{c_0} \end{aligned}$$

Approximation B: Linearizing the trigonometric functions

If we use $\sin\tau = \tau$ and $\cos\tau = 1$ then (5.5) becomes

$$\hat{\mathbf{t}}(x) = R(\beta)\begin{pmatrix} 1 \\ -c_0 x - c_1 x^2/2 \end{pmatrix}$$

Plugging this into (5.2) we get

$$\mathbf{r}(x) = R(\beta)\begin{pmatrix} x \\ -c_0 x^2/2 - c_1 x^3/6 \end{pmatrix} - R(\beta)\begin{pmatrix} 0 \\ y_{off} \end{pmatrix}$$

and from (5.3) we get the coordinate transformation

$$T_b(x, y) \triangleq R(\beta)\begin{pmatrix} x + y(c_0 x + c_1 x^2/2) \\ y - y_{off} - c_0 x^2/2 - c_1 x^3/6 \end{pmatrix}$$

Approximation C: As B, plus further approximations

In the last few approximation steps, we ignore all curve-effects in the $\tilde{x}$-coordinate, i.e., $\tilde{x} = x$

$$\begin{aligned}
T_b(x, y) &\approx R(\beta)\begin{pmatrix} x \\ y - y_{\mathit{off}} - c_0x^2/2 - c_1x^3/6 \end{pmatrix} \approx \\
&\approx \begin{pmatrix} 1 & -\beta \\ \beta & 1 \end{pmatrix}\begin{pmatrix} x \\ y - y_{\mathit{off}} - c_0x^2/2 - c_1x^3/6 \end{pmatrix} = \\
&= \begin{pmatrix} x - \beta(y - y_{\mathit{off}} - c_0x^2/2 - c_1x^3/6) \\ \beta x + y - y_{\mathit{off}} - c_0x^2/2 - c_1x^3/6 \end{pmatrix} \approx \\
&\approx \begin{pmatrix} x \\ y - y_{\mathit{off}} + \beta x - c_0x^2/2 - c_1x^3/6 \end{pmatrix} \triangleq T_c(x, y)
\end{aligned}$$

5.2.4 State space model

To be able to use a Kalman filter, we will build three state space models, based on the approximations from the previous section. The states for the host vehicle are shown in Figure 5.3. Note that we have $\beta = -\Psi_{rel}$. We need c_1 as state in all three filters for dynamic reasons. Observed vehicles will have the states x_i, $\dot{x}_i$ and y_i, where i runs through all detected objects.

Measurement equations

The measurements for the host vehicle are Ψ^m_{rel}, c^m_0, L^m and R^m where the last two are the distances to the left and right lane marking. Superscript m denotes measured quantities. For other vehicles we measure the position, $\tilde{x}^m$ and $\tilde{y}^m$. These relate to the states as

$$\begin{aligned}
L^m_t &= -W_t/2 - y_{\mathit{off},t} + e_{1,t} \\
R^m_t &= W_t/2 - y_{\mathit{off},t} + e_{2,t} \\
\Psi^m_{rel,t} &= \Psi_{rel,t} + e_{3,t} \\
c^m_{0,t} &= c_{0,t} + e_{4,t}
\end{aligned} \tag{5.6a}$$

$$\begin{pmatrix} \tilde{x}^{i,m}_t \\ \tilde{y}^{i,m}_t \end{pmatrix} = T(x^i_t, y^i_t) + \begin{pmatrix} e_{5,t} \\ e_{6,t} \end{pmatrix}^i \tag{5.6b}$$

where T can be replaced with any of the approximations. Of course, T depends on all the host vehicle states as well. The variables $(e_1, \ldots, e_6)$ are some stochastic measurement noise.

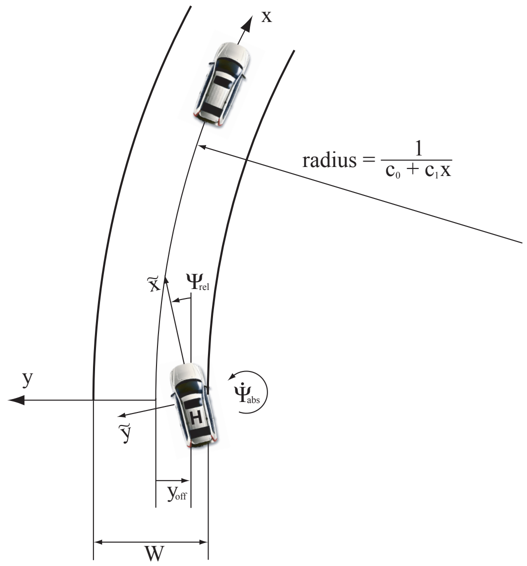


Figure 5.3: W, y_{off}, Ψ_{rel}, c_0 and c_1 are the host vehicle states. The mapping T transforms from the coordinate system (x, y) to the coordinate system $(\tilde{x}, \tilde{y})$.

Motion models

Since x and y are the curved road coordinates, the motion model of other vehicles can be greatly simplified. For example, it allows us to use the equation $\dot{y}^i = 0$ which simply means that we assume that other vehicles will follow their own lanes. In the longitudinal direction we will use $\ddot{x}^i = 0 - a_{host} \cos \Psi_{rel}$, a_{host} being the measured acceleration of the host vehicle so that we get the motion equations:

$$\begin{aligned} \dot{x}^i &= v^i \\ \dot{v}^i &= a_{host,t} \cos \Psi_{rel,t} \\ \dot{y}^i &= 0 \end{aligned} \tag{5.7a}$$

where v^i is the longitudinal velocity of object i, i.e., the time derivative of x^i. For the road geometry parameters we first clarify that Ψ_{rel} is the angle offset to the lane and Ψ_{abs} is the angle to some fix reference. We can obtain a relationship between the two by taking the time derivative of Ψ_{rel}

$$\begin{aligned}
&\Psi_{rel} = \Psi_{abs} + \gamma \quad \Rightarrow \\
&\dot{\Psi}_{rel} = \dot{\Psi}_{abs} + \dot{\gamma} = \dot{\Psi}_{abs} + \frac{v}{r} = \dot{\Psi}_{abs} + c_0 v
\end{aligned}$$

where r is the current road radius, v the velocity and γ denotes the angle between the lane and some fix reference. $\dot{\Psi}_{abs}$ can typically be measured with a yaw rate sensor. We also have

$$\dot{y}_{\mathit{off}} = \sin(\Psi_{rel})v \approx \Psi_{rel} v$$

Using $\dot{W} = 0$ and $\dot{c}_1 = 0$ we can write the time continuous motion equations for the host vehicle states:

$$\begin{aligned}
\dot{W} &= 0 \\
\dot{y}_{\mathit{off}} &= v\Psi_{rel} \\
\dot{\Psi}_{rel} &= vc_0 + \dot{\Psi}_{abs} \\
\dot{c}_0 &= vc_1 \\
\dot{c}_1 &= 0
\end{aligned}$$

The discrete time motion dynamics can then be computed using the formulas

$$\begin{aligned}
A &= \exp(A^c T_s) \\
B &= \int_0^T \exp(A^c s) B^c ds
\end{aligned}$$

from [14] where T_s is the sample time and (A, B) and (A^c, B^c) refers to the discrete time and continuous time system matrices, respectively. Also adding stochastic process noise, the discrete time system dynamics becomes

$$\begin{aligned}
x^i_{t+1} &= x^i_t + T_s \dot{x}^i_t + a_{host,t} \cos \Psi_{rel,t} T_s^2/2 + w_{1,t} \\
v^i_{t+1} &= \dot{x}^i_t + a_{host,t} \cos \Psi_{rel,t} T_s + w_{2,t} \\
y^i_{t+1} &= y^i_t + w_{3,t}
\end{aligned} \tag{5.7b}$$

and for the host vehicle states

$$\begin{aligned}
W_{t+1} &= W_t + w_{4,t} \\
y_{\mathit{off},t+1} &= y_{\mathit{off},t} + vT_s\Psi_{rel,t} + v^2T_s^2 c_{0,t}/2 + v^3 T_s^3 c_{1,t}/6 + vT^2\dot{\Psi}_{abs,t}/2 + w_{5,t} \\
\Psi_{rel,t+1} &= \Psi_{rel,t} + vT_s c_{0,t} + v^2 T_s^2 c_{1,t}/2 + T_s \dot{\Psi}_{abs,t} + w_{6,t} \\
c_{0,t+1} &= c_{0,t} + vT_s c_{1,t} + w_{7,t} \\
c_{1,t+1} &= c_{1,t} + w_{8,t}
\end{aligned} \tag{5.7c}$$

The variables $(w_1, \ldots, w_8)$ are stochastic process noise.

5.3 Applying the Extended Kalman Filter

We can now construct an observer based on this model. We define the matrices

$$A_{host} = \begin{pmatrix} 1 & 0 & 0 & 0 & 0 \\ 0 & 1 & vT_s & v^2T_s^2/2 & v^3T_s^3/6 \\ 0 & 0 & 1 & vT_s & v^2T_s^2/2 \\ 0 & 0 & 0 & 1 & vT_s \\ 0 & 0 & 0 & 0 & 1 \end{pmatrix} \quad A_{obj} = \begin{pmatrix} 1 & T_s & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{pmatrix}$$

$$A = \begin{pmatrix} A_{host} & 0 \\ 0 & I_N \oplus A_{obj} \end{pmatrix}$$

$$B_{host} = \begin{pmatrix} 0 & 0 \\ vT^2/2 & 0 \\ T_s & 0 \\ 0 & 0 \\ 0 & 0 \end{pmatrix} \quad B_{obj} = \begin{pmatrix} 0 & T_s^2/2 \\ 0 & T_s \\ 0 & 0 \end{pmatrix} \quad B = \begin{pmatrix} B_{host} \\ B_{obj} \\ \vdots \\ B_{obj} \end{pmatrix}$$

$$C_{host} = \begin{pmatrix} -1/2 & -1 & 0 & 0 & 0 \\ 1/2 & -1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 \end{pmatrix}$$

where N is the number of objects, and the vectors

$$x_{host,t} = \begin{pmatrix} W \\ y_{\mathit{off}} \\ \Psi_{rel} \\ c_0 \\ c_1 \end{pmatrix}_t \quad x^i_{obj,t} = \begin{pmatrix} x^i \\ v^i \\ y^i \end{pmatrix}_t \quad x_t = \begin{pmatrix} x_{host} \\ x^1_{obj} \\ \vdots \\ x^N_{obj} \end{pmatrix}_t$$

$$y_{host,t} = \begin{pmatrix} L^m \\ R^m \\ \Psi^m_{rel} \\ c^m_0 \end{pmatrix}_t \quad y^i_{obj,t} = \begin{pmatrix} \tilde{x}^m_i \\ \tilde{y}^m_i \end{pmatrix}_t \quad y_t = \begin{pmatrix} y_{host} \\ y^1_{obj} \\ \vdots \\ y^N_{obj} \end{pmatrix}_t$$

$$u_t = \begin{pmatrix} \dot{\Psi}_{abs,t} \\ a_{host,t} \cos \Psi_{rel,t} \end{pmatrix}$$

We also introduce

$$h(x_t) = \begin{pmatrix} C_{host}x_{host,t} \\ T(x^1_{obj,t}) \\ \vdots \\ T(x^N_{obj,t}) \end{pmatrix}$$

The Kalman filter will also require the process and measurement noise covariance matrices which we define as

$$Q = \begin{pmatrix} Q_{host} & 0 \\ 0 & I_N \oplus Q_{obj} \end{pmatrix} \quad R = \begin{pmatrix} R_{host} & 0 \\ 0 & I_N \oplus R_{obj} \end{pmatrix}$$

where Q_{host} and Q_{obj} are the process noise covariance matrices for the host and object states and R_{host} and R_{obj} are the measurement noise covariance matrices for the host and object measurements. The measurement equations (5.6) and motion equations (5.7) can now be rewritten as

$$\begin{aligned} x_{t+1} &= Ax_t + Bu_t + w_t \\ y_t &= h(x_t) + e_t \end{aligned}$$

In the remainder of this chapter we shall often use the term "track". A track simply refers to a set of states and a covariance matrix corresponding to one object.

5.3.1 Data association

For data association, a standard method that can be found in for example [6], is used. It should be mentioned that only a suboptimal solution to the assignment problem is used. The statically optimal solution is defined as the solution that minimizes the sum of the distances between the assigned tracks and measurements. There do exist algorithms that find the optimal solution, they are however rather complicated. Here, a faster and less complex algorithm has been used, but no problems related to the data association have been noticed.

The algorithm starts with computing the distance between all possible track-to-measurement pairs. The distance can be the Euclidian distance but also some other distance measure based on probability. The distances are put in a matrix called the assignment matrix and the algorithm then uses the following two steps:

1. Search the assignment matrix for the closest (minimum distance) track-to-measurement pair and make the indicated assignment.

	M1	M2	M3	M4
T1	9	3	17	5
T2	7	9	4	8
T3	2	11	13	14

	M1	M2	M3	M4
T1	9	3	17	5
T2	7	9	4	8
T3	[2]	11	13	14

	M1	M2	M3	M4
T1	9	[3]	17	5
T2	7	9	4	8
T3	[2]	11	13	14

	M1	M2	M3	M4
T1	9	[3]	17	5
T2	7	9	[4]	8
T3	[2]	11	13	14

Table 5.1: Example of the data association algorithm. T1 - T3 are the tracks and M1 - M4 are the measurements. The distance matrix is shown in the upper left table. The result of the first iteration is shown in the upper right table and the remaining two iterations in the lower two tables. The shaded numbers indicate that the numbers are removed according to step 2 of the algorithm.

2. Remove the observation-to-track pair identified above from the assignment matrix and repeat step 1 for the reduced matrix.

An example is shown in Table 5.1.

Based on the results from the data association at time t, two permutation matrices, $\mathbf{X}_t$ and $\mathbf{Y}_t$, will be formed. The matrix $\mathbf{X}_t$ will remove targets that were not associated. The matrix $\mathbf{Y}_t$ will rearrange the measurements to match the targets and remove measurements that were not associated. The matrices will also feed through the host states and measurements.

If the measurement equation of a model is

$$y_t = h(x_t) + e_t$$

then a "new" h can be formed as $\mathbf{X}_t h$ and the new innovation can be written

$$\varepsilon_t = \mathbf{Y}_t y_t - \mathbf{X}_t h(\hat{x}_t)$$

Also, if the covariance matrix of e_t is R, then the covariance of $\mathbf{Y}_t e_t$ is $\mathbf{Y}_t R \mathbf{Y}_t^T$.

5.3.2 Filter equations

In this section we build a recursive one step predictor with the structure

$$\hat{x}_{t+1|t} = A(\hat{x}_{t|t-1} + \tilde{K}_t[y_t - h(\hat{x}_{t|t-1})]) + Bu_t$$

as an observer to the combined target/geometry system

$$x_{t+1} = Ax_t + Bu_t + w_t$$
$$y_t = h(x_t) + e_t$$

The Extended Kalman Filter (EKF) will provide us with a feedback K. See [14, 19, 20, 21] for details on the Kalman Filter and the Extended Kalman Filter. These are the EKF equations for a non-linear measurement equation:

$$C_t = D_x h(\hat{x}_{t|t-1}) \tag{5.8a}$$

$$K_t = P_{t-1} C_t^T (C_t P_{t-1} C_t^T + R)^{-1} \tag{5.8b}$$

$$P_t = A P_{t-1} A^T + Q - A K_t C_t P_{t-1} A^T \tag{5.8c}$$

where

$$[D_x h]_{ij} = \frac{\partial h_i}{\partial x_j}$$

Next we modify (5.8) by applying the data association matrices derived in Section 5.3.1. The EKF then looks like:

$$C_t = \mathbf{X}_t D_x h(\hat{x}_{t|t-1}) \tag{5.9a}$$

$$K_t = P_{t-1} C_t^T (C_t P_{t-1} C_t^T + \mathbf{Y}_t R \mathbf{Y}_t^T)^{-1} \tag{5.9b}$$

$$P_t = A P_{t-1} A^T + Q - A K_t C_t P_{t-1} A^T \tag{5.9c}$$

$$\hat{x}_{t+1|t} = A(\hat{x}_{t|t-1} + K_t[\mathbf{Y}_t y_t - \mathbf{X}_t h(\hat{x}_{t|t-1})]) + Bu_t \tag{5.9d}$$

Note that it is important to include tracks that were not associated in the measurement update. All tracks are interacting via the road geometry model, and thus all tracks will be affected by the measurement update, even though some of them might not have been associated with any measurements.

The derivatives of the measurement equation that are needed in the Extended Kalman Filter are stated in Appendix 5.B.

5.3.3 Creation/Destruction of tracks

In this section the method for creating and destroying tracks that have been used is explained. After the data association algorithm has been run we are left with a set of measurements that were not associated to any tracks. These measurements will be used to generate new tracks. The states corresponding to one object is added to the state vector. In order to initiate the states, the inverse of the measurement equation has to be used. Inverses of the different geometric models are derived in Appendix 5.C.

A counter is associated with each track, and for each time a measurement is associated with the track the counter is increased one step, and for each time the track is not associated to any measurement, the counter is decreased one step. Once a counter reaches zero, the track which it corresponds to will be removed.

This is to make sure single false measurements do not result in persisting tracks, and also to make sure tracks are not deleted due to a single missed measurement. A saturation of the counter should be used and the saturation level is a tuning variable that has to be adjusted to reach the best performance.

5.3.4 Change detection

Lane change: other vehicles

One of the main features of the tracking system we have derived in this chapter is that it allows us to improve the road geometry estimate by studying the motion of leading vehicles. This is achieved using the assumption that other vehicles stay in their lanes most of the time. So what happens when leading vehicles actually do not follow their lanes, for example during a lane change?

If this issue is not treated explicitly, there is a good chance we will be misled if relying too much on leading vehicles following their lanes. There is, of course, always room for lateral motion of tracked vehicles. The third line of the motion equations (5.7b) certainly has an additive noise term. The Kalman filter, however, expects this noise to be white, which is why it is not ideally suited for typical lane change manoeuvres.

We could simply ignore this problem, trying to tune the lateral process noise of vehicle tracks so that it fits approximately in both cases. If we do get an overall improvement in the lane geometry estimate, we are still doing something good.

Another way of dealing with the problem could be some sort of change detection algorithm. Such an algorithm would try to detect a lane change, by for example doing a whiteness test on the corresponding innovations. If a commenced lane change was detected, the lateral process noise could temporarily be increased in order to rely less on the assumption that this particular vehicle will follow its lane. Many such tests are proposed in [14].

A slightly different approach is suggested in [27]. Here, a concept called Interacting Multiple Models (IMM) is used to incorporate lane changes.

Several filters based on models with different dynamics are run in parallel and the output from these are weighed together with a statistical method based on the innovations from the different filters.

Lateral change detection for leading vehicles has not yet been implemented in the demonstrator, but it would certainly be interesting to investigate the potential increase in performance of the different methods, in particular when using them together with the road-aligned coordinate system.

Lane change: host vehicle

Host vehicle lane changes also need to be detected. The vision system will report an abrupt change in the distances to the two lane markings. It is important not to regard this as an innovation, instead it should be used to detect the lane change. Then simply add or subtract, depending on the direction of the lane change, the distance of one lane width to the lateral position estimate of the host vehicle.

5.4 Evaluation

5.4.1 Geometric comparison

From [1] we get the following guidelines for road construction. For a 50 km/h road, the minimum radius is 140 meters and for a 90 km/h road it is 550 meters. The recommended maximum clothoid parameters for these curves are given by the formula

$$c_1 = \frac{k}{v^3}$$

where $k = 0.45$ m/s^3 which is the maximum "jerk" and v the velocity, giving the clothoid $1.7 \cdot 10^{-4}$ and $2.9 \cdot 10^{-5}$ 1/m^2 for the 50 km/h and the 90 km/h curve respectively. In Figure 5.4 we have compared the approximations with the exact transformation for a clothoid curve. Note that if we would have analyzed a curve with $c_1 = 0$, i.e., a straight line or a pure circle segment, T_a would have coincided with the exact transformation.

It can be seen that all three approximations clearly deviates from the exact transformation. Also note the artifact $\tilde{x} = x$ in Approximation C, seen as a horizontal edge.

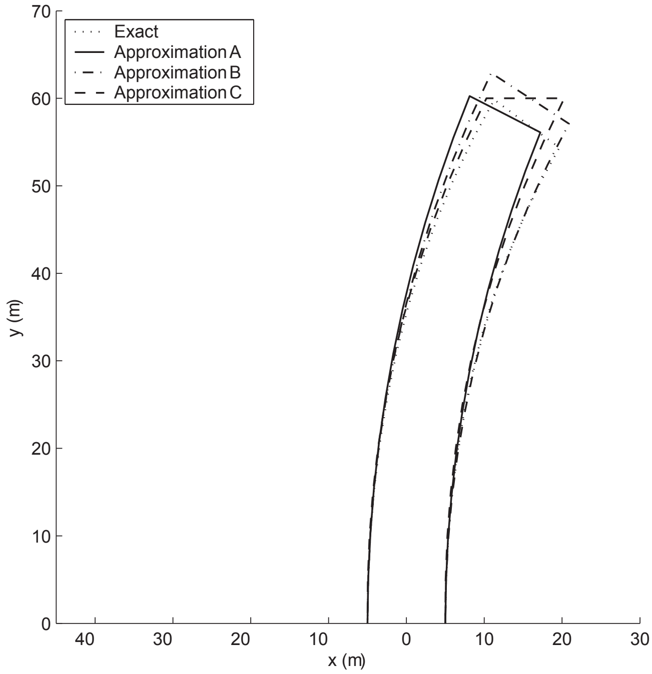


Figure 5.4: Illustration of the different approximations. A road with edges at $y = -5$ and $y = 5$ has been transformed with the three approximations and with the true transformation, based on typical a 50 km/h curve. The true curve was obtained from numeric integration.

5.4.2 Accuracy of the curvature estimate

To analyze the different models further, the extended Kalman filter from the previous section was used. The filter was implemented with the different geometric approximations. The accuracy of the lane geometry was then evaluated during a test drive. A vehicle equipped with a camera and a radar was used to record data. Lane geometry measurements were given by the camera and measurements of other vehicles were given both from the camera and the radar. Some additional data from the vehicle, such as velocity and yaw rate, were also collected. The true values of the lane geometry were obtained from a detailed map.

Four filters were then run on the same data, the three approximations that was derived in Section 5.2.3, plus a completely decoupled model, i.e.,

	Host	Obstacles
Process noise	5	3
Measurement noise	4	2

Table 5.2: Number of tuning parameters

tracking of obstacles and lane geometry done separately.

Filter tuning

Filter tuning is the process of adjusting the entries in the "Q" and "R" matrices, which are often interpreted as process and measurement noise covariance. If we constrain these to be diagonal, we have 14 parameters to tune, as shown in Table 5.2.

The tuning was started by first using "physical" intuition trying to judge errors in measurements and changes in different states. This was then used as a starting point for hours of manual tuning in order to get acceptable performance from the three filters.

After that, a more systematic approach was used. A scaling parameter was applied to selected filter parameters. For example, if the host states process noise covariance matrix is called Q_{host}, the filters was run with this matrix replaced with λQ_{host}, where λ for example ranges from 10^{-2} to 10^2. Then the mean error in some parameter, usually the curvature c_0, was computed. This procedure was then applied to different combinations of parameters or single parameters. This is of course only a sub-optimization in the 14 dimensional parameter space.

Results

Figure 5.5 and Figure 5.6 show some preliminary results of the performance of the different filters. The three approximations have been compared to a decoupled linear filter, where lane geometry and obstacles were tracked separately. Figure 5.5 shows a data sequence recorded during bad visibility. It shows that the performance can be improved by using a combined filter, which has also been demonstrated in for example [29] and [12]. Figure 5.6 shows a data set recorded during good visibility. In this case, there is only a small improvement by using integrated filtering.

To make the experiment more interesting, we have only allowed the process noise of the road measurements to vary between the two cases, all other parameters are kept constant. It can be seen in Figures 5.5 and 5.6 that the optimal performance is reached at higher measurement noise for

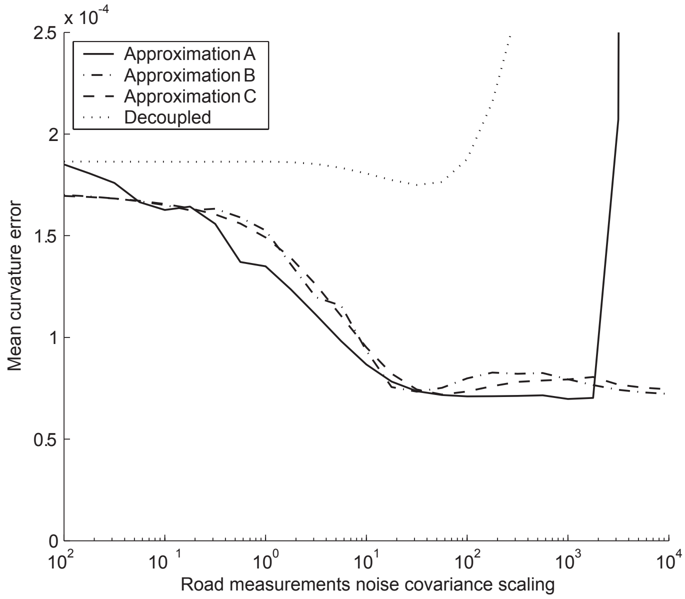


Figure 5.5: **Curvature error during bad visibility.** The measurement noise of the road measurements has been scaled from 10^{-2} to 10^{4}. The plot shows the error in the curvature estimate for the different filters. This also includes a decoupled filter where the road geometry and the obstacles are treated separately. Note that Approximation A becomes unstable for high measurement noise values.

the bad visibility case than the good visibility case. This is intuitive, if bad visibility was detected by for example the vision system, you would typically increase the process noise of road measurements in the Kalman filter in order to rely more on other measurements and on the motion model.

It should be noted that the data in Figure 5.5 and Figure 5.6 are from different roads and during different traffic conditions. Therefore, care should be taken before comparing the performance in the two experiments.

Also, even though the curvature is important in many applications, it is just one of many parameters. A more thorough evaluation of the performance could be done if the particular application of the filter was known.

In the next section we examine a property that is important in many collision avoidance applications.

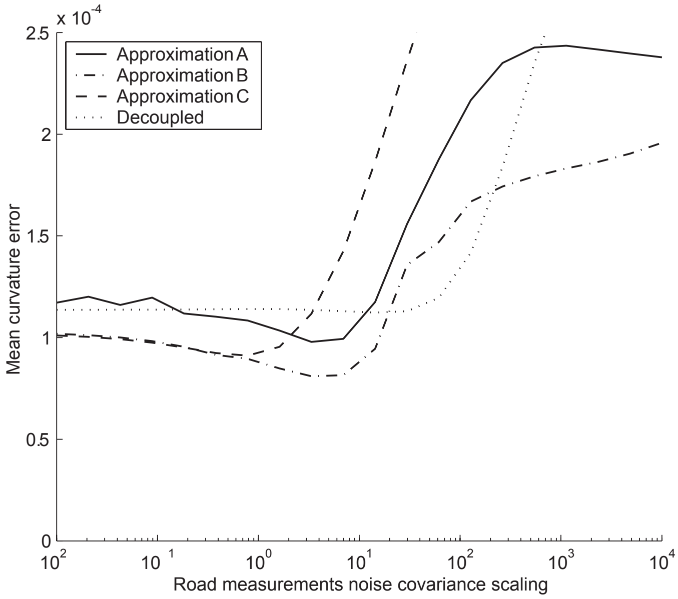


Figure 5.6: **Curvature error during good visibility.** The measurement noise of the road measurements has been scaled from 10^{-2} to 10^4. The plot shows the error in the curvature estimate for the different filters.

5.4.3 Lane assignment

Lane assignment is the problem of deciding in which lanes the tracked vehicles are currently driving. This is where the quality of the lane geometry estimate becomes utterly important, even the slightest error in heading angle or curvature will result in a significant lateral error for vehicles at a long distance, say 70 - 100 meters.

All three model approximations from Section 5.2.3 were run on the same data, again together with a decoupled filter. Doing lane assignment when using these filters becomes trivial due to the curved coordinate system. We only need to consider the lateral position of each estimate, i.e., the y_t^i states, and the current lane width, W_t. We will refer to this quantity as the estimated lane index, $\hat{\chi}_t^i$. It is defined as

$$\hat{\chi}_t^i \triangleq \begin{cases} -1 & y_t^i < -W_t/2 \\ 0 & -W_t/2 < y_t^i < W_t/2 \\ +1 & W_t/2 < y_t^i \end{cases}$$

Filter	Bad visibility	Good visibility
Approximation A	0.84	0.94
Approximation B	0.84	0.88
Approximation C	0.78	0.74
Decoupled	0.12	0.81

Table 5.3: Lane assignment accuracy

This value then needs to be compared to the true value, χ_t^i. This was obtained by placing the measured position of other vehicles on the true map. This method is based on the assumption that the error in the position of the obstacles is small compared to the error in the lane geometry.

We then compute the accuracy of the lane assignment as the ratio of the correct number of assignments and the sum of the total number of obstacles summed over the entire data set, i.e., if $n_{\text{tot},t}$ is the total number of obstacles for time step t and $n_{\text{correct},t}$ is the number of correct assignments, we define

$$\text{Lane assignment accuracy} \triangleq \frac{\sum_t n_{\text{correct},t}}{\sum_t n_{\text{tot},t}}$$

Results

The lane assignment accuracy of the different approximations from Section 5.2.3 are shown in Table 5.3. It shows that the improvement achieved by integrated filtering during the bad visibility case is significant. In the case of good visibility, there is some improvement for Approximation A and B.

5.5 Conclusions

It is clear that combined lane prediction and target tracking can give better estimates and improve the accuracy of state estimates and improve the performance of applications such as lane assignment. The integrated filter bring equations which need to be approximated, and it has been shown that there might be better alternatives than linearizing the trigonometric functions, which is often done.

Tuning the different filters is a very difficult task. The performance of all three filters can probably be improved if a more systematic tuning procedure is used. There are also stability issues, all three filters tend to diverge for certain, badly chosen, sets of tuning parameters. Stability of the extended Kalman filter can never be guaranteed.

Despite this, the improvement in performance shown in these experiments cannot be overlooked, no matter which approximation is used. In the development of ELA, we have chosen Approximation A due to its high accuracy in lane assignment. The instable behavior shown in Figure 5.5 is not a problem since we can simply choose a measurement noise matrix in the stable region.

5.A Proof of (5.1)

In this section we show that the equation

$$\exp(R(-\frac{\pi}{2})\int_0^x c(\tau)d\tau)\hat{\mathbf{t}}_0 = R(-\int_0^x c(\tau)d\tau)\hat{\mathbf{t}}_0 \tag{5.10}$$

from page 39 holds. The expression holds if, for any scalar function $\phi(x)$

$$\exp(R(-\frac{\pi}{2})\phi(x)) = R(-\phi(x)) \tag{5.11}$$

In order to evaluate the left hand side we use diagonalization of the argument:

$$R(-\frac{\pi}{2})\phi(x) = \begin{pmatrix} 0 & 1 \\ -1 & 0 \end{pmatrix} \phi(x) = VD\phi(x)V^*$$

where

$$V = \frac{1}{\sqrt{2}}\begin{pmatrix} 1 & 1 \\ i & -i \end{pmatrix} \quad \text{and} \quad D = \begin{pmatrix} i & 0 \\ 0 & -i \end{pmatrix}$$

Then, since V is orthonormal we have that $V^{-1} = V^*$ and thus

$$\exp(VD\phi(x)V^*) = V\exp(D\phi(x))V^* = V\begin{pmatrix} \exp(i\phi(x)) & 0 \\ 0 & \exp(-i\phi(x)) \end{pmatrix} V^*$$

Now, we multiply these three matrices and get

$$\begin{aligned}
&\exp(R(-\frac{\pi}{2})\phi(x)) = \exp(VD\phi(x)V^*) = \\
&= \frac{1}{2}\begin{pmatrix} \exp(i\phi(x)) + \exp(-i\phi(x)) & -\exp(i\phi(x)) + \exp(-i\phi(x)) \\ \exp(i\phi(x)) - \exp(-i\phi(x)) & \exp(i\phi(x)) + \exp(-i\phi(x)) \end{pmatrix} = \\
&= \begin{pmatrix} \cos\phi(x) & \sin\phi(x) \\ -\sin\phi(x) & \cos\phi(x) \end{pmatrix} = R(-\phi(x))
\end{aligned}$$

This is the same expression as (5.11) and thus (5.10) holds.

5.B Measurement equation derivatives

In this chapter the derivatives of the different measurement equations that are needed in the Extended Kalman Filter are stated. We need

$$D_x h = \begin{pmatrix} \partial y_{host}/\partial x_{host} & \partial y_{host}/\partial x^1_{obj} & \cdots & \partial y_{host}/\partial x^N_{obj} \\ \partial y^1_{obj}/\partial x_{host} & \partial y^1_{obj}/\partial x^1_{obj} & \cdots & \partial y_{obj^1}/\partial x^N_{obj} \\ \vdots & \vdots & \ddots & \vdots \\ \partial y^N_{obj}/\partial x_{host} & \partial y^N_{obj}/\partial x^1_{obj} & \cdots & \partial y_{obj^N}/\partial x^N_{obj} \end{pmatrix}$$

where

$$\frac{\partial y_{host}}{\partial x_{host}} = C_{host}$$

$$\frac{\partial y_{host}}{\partial x^i_{obj}} = 0 \quad \forall \quad i$$

$$\frac{\partial y^i_{obj}}{\partial x^j_{obj}} = \begin{cases} \partial T(x^i_{obj})/\partial x^i_{obj} & i = j \\ 0 & i \neq j \end{cases}$$

$$\frac{\partial y^i_{obj}}{\partial x_{host}} = \partial T(x^i_{obj})/\partial x_{host}$$

Approximation A

The derivatives are:

$$\frac{\partial T}{\partial W} = 0$$

$$\frac{\partial T}{\partial y_{off}} = R(-\Psi_{rel}) \begin{pmatrix} 0 \\ -1 \end{pmatrix}$$

$$\frac{\partial T}{\partial \Psi_{rel}} = -R'(-\Psi_{rel}) \left[\begin{pmatrix} (1 + c_0 y^i) \sin(c_0 x^i) \\ (1 + c_0 y^i) \cos(c_0 x^i) - 1 \end{pmatrix} \frac{1}{c_0} - \begin{pmatrix} 0 \\ y_{off} \end{pmatrix} \right]$$

$$\frac{\partial T}{\partial c_0} = R(-\Psi_{rel}) \begin{pmatrix} (c_0 x^i + c_0^2 x^i y^i) \cos(c_0 x^i) - \sin(c_0 x^i) \\ 1 - (c_0 x^i + c_0^2 x^i y^i) \sin(c_0 x^i) - \cos(c_0 x^i) \end{pmatrix} \frac{1}{c_0^2}$$

$$\frac{\partial T}{\partial x^i} = R(-\Psi_{rel}) \begin{pmatrix} (1 + c_0 y^i)\cos(c_0 x^i) \\ (-1 - c_0 y^i)\sin(c_0 x^i) \end{pmatrix}$$

$$\frac{\partial T}{\partial y^i} = R(-\Psi_{rel}) \begin{pmatrix} \sin(c_0 x^i) \\ \cos(c_0 x^i) \end{pmatrix}$$

Approximation B

The derivatives are:

$$\frac{\partial T_b}{\partial x} = R(-\Psi_{rel}) \begin{pmatrix} 1 + y(c_0 + c_1 x) \\ -c_0 x - c1^2/2 \end{pmatrix}$$

$$\frac{\partial T_b}{\partial y} = R(-\Psi_{rel}) \begin{pmatrix} c_0 x + c_1 x^2/2 \\ 1 \end{pmatrix}$$

$$\frac{\partial T_b}{\partial W} = R(-\Psi_{rel}) \begin{pmatrix} 0 \\ 0 \end{pmatrix}$$

$$\frac{\partial T_b}{\partial y_{off}} = R(-\Psi_{rel}) \begin{pmatrix} 0 \\ -1 \end{pmatrix}$$

$$\frac{\partial T_b}{\partial \Psi_{rel}} = -R'(-\Psi_{rel}) \begin{pmatrix} x + y(c_0 x + c_1 x^2/2) \\ y - y_{off} - c_0 x^2/2 - c_1 x^3/6 \end{pmatrix}$$

$$\frac{\partial T_b}{\partial c_0} = R(-\Psi_{rel}) \begin{pmatrix} yx \\ -x^2/2 \end{pmatrix}$$

$$\frac{\partial T_b}{\partial c_1} = R(-\Psi_{rel}) \begin{pmatrix} yx^2/2 \\ -x^3/6 \end{pmatrix}$$

Approximation C

The derivatives are:

$$\frac{\partial \varphi_i}{\partial W} = 0 \quad \frac{\partial \varphi_i}{\partial y_{off}} = -\frac{1}{x^i} \quad \frac{\partial \varphi_i}{\partial \Psi_{rel}} = -1$$

$$\frac{\partial \varphi_i}{\partial c_0} = -\frac{1}{2}x^i \quad \frac{\partial \varphi_i}{\partial c_1} = -\frac{1}{6}(x^i)^2$$

$$\frac{\partial \varphi_i}{\partial x^i} = -\frac{y^i - y_{off}}{(x^i)^2} - \frac{1}{2}c_0 - \frac{1}{3}c_1 x^i \quad \frac{\partial \varphi_i}{\partial y^i} = \frac{1}{x^i}$$

Polar coordinates

If we instead were to measure angle and range, we use the transformation

$$z_{obj} = \begin{pmatrix} \varphi \\ r \end{pmatrix} = \begin{pmatrix} \arctan(\tilde{y}/\tilde{x}) \\ \sqrt{\tilde{x}^2 + \tilde{y}^2} \end{pmatrix}$$

with

$$\frac{dz_{obj}}{dy_{obj}} = \begin{pmatrix} -\tilde{y}/r^2 & \tilde{x}/r^2 \\ \tilde{x}/r & \tilde{y}/r \end{pmatrix}$$

5.C Measurement equation inverse

In order to initiate new tracks we need to compute the inverse of the measurement equation. i.e., for a given set of $(W, y_{off}, \Psi_{rel}, c_0)$ we need to find the mapping $T^{-1} : (\tilde{x}, \tilde{y}) \to (x, y)$.

Approximation A

Defining

$$\begin{pmatrix} \xi \\ \eta \end{pmatrix} = R^{-1}(\Psi_{rel}) \begin{pmatrix} \tilde{x} \\ \tilde{y} \end{pmatrix} + \begin{pmatrix} 0 \\ y_{off} \end{pmatrix} = \begin{pmatrix} (1 + c_0 y)\sin(c_0 x) \\ (1 + c_0 y)\cos(c_0 x) - 1 \end{pmatrix} \frac{1}{c_0} \tag{5.15}$$

we first note that

$$\lim_{c_0 \to 0} \begin{pmatrix} \xi \\ \eta \end{pmatrix} = \begin{pmatrix} x \\ y \end{pmatrix}$$

On the other hand, if $c_0 \neq 0$ we get from row one of equation 5.15, assuming also that $x > 0$

$$(1 + c_0 y) = \frac{c_0 \xi}{\sin(c_0 x)}$$

which, used in the second row of equation 5.15, yields

$$\eta = (c_0 \xi \frac{\cos(c_0 x)}{\sin(c_0 x)} - 1)\frac{1}{c_0} = \frac{\xi}{\tan(c_0 x)} - \frac{1}{c_0}$$

i.e.,

$$x = \frac{1}{c_0} \arctan \frac{\xi}{\eta + 1/c_0}$$

We can then use this in row one of equation 5.15 and get

$$y = \frac{1}{c_0}(\frac{c_0 \xi}{\sin(c_0 x)} - 1) = \frac{\xi}{\sin(c_0 x)} - \frac{1}{c_0}$$

We have thus constructed the mapping $(\tilde{x}, \tilde{y}) \to (\xi, \eta) \to (x, y)$.

Approximation B

No explicit inverse of approximation B exists, since it involves finding the roots of a polynomial of order five [4]. Instead, it can be approximated with one of the other two.

Approximation C

Since we have $\tilde{x} = x$ we can plug this directly into row two of equation 5.2.3 and get

$$\tilde{y} = y - y_{off} + \beta\tilde{x} - c_0\tilde{x}^2/2 - c_1\tilde{x}^3/6$$

and thus

$$y = \tilde{y} + y_{off} - \beta\tilde{x} + c_0\tilde{x}^2/2 + c_1\tilde{x}^3/6$$

Chapter 6

Decision and intervention

This chapter will explain the decision and intervention unit, see Figure 3.2 on page 27.

6.1 Decision strategy

The decision algorithm will be based on the observer that was derived in Chapter 5. As a reminder, the states for the host vehicle and lane geometry are

W	-	Lane width
y_{off}	-	Host vehicle lateral position in lane
Ψ_{rel}	-	Heading angle: Angle between host vehicle and lane
c_0	-	Lane curvature parameter
c_1	-	Lane clothoid parameter

and for each observed object they are

x^i	-	Longitudinal coordinate of object i
$\dot{x}^i$	-	Time derivative of x^i
y^i	-	Lateral coordinate of object i

Two external signals are also available:

v	-	Host vehicle velocity
$\dot{\Psi}_{abs}$	-	Host vehicle yaw rate

6.1.1 Time to lane crossing

An essential part of the decision algorithm will be a measure called Time to Lane Crossing (TLC) and will here refer to the predicted time until one of the front tires intersects the lane boundaries. The article [26] suggests different ways to compute this estimate. The simplest one, which we have used here, is simply based on the lateral position and the lateral velocity. Two values are computed, $TLC1$, the time to reach the first lane marking, and $TLC2$, the time to reach the second lane marking, i.e., the time to exit the adjacent lane again on the other side. The following expressions assume $\dot{y}_{off} > 0$, the other case is treated similarly.

$$
\begin{aligned}
TLC1 &= \frac{W/2 - W_{veh}/2 - y_{off}}{\dot{y}_{off}} \approx \frac{W/2 - W_{veh}/2 - y_{off}}{\Psi_{rel} v} \\
TLC2 &= \frac{3W/2 + W_{veh}/2 - y_{off}}{\dot{y}_{off}} \approx \frac{3W/2 + W_{veh}/2 - y_{off}}{\Psi_{rel} v}
\end{aligned}
$$

Here, W_{veh} refers to the host vehicle width. So far, the performance of these measures has been satisfactory, but if problems are detected in the future, more advanced versions will be investigated.

6.1.2 Decision algorithm

The goal of the decision strategy, according to Chapter 3, is to detect when a commenced lane change manoeuvre will result in a dangerous situation. This is done in the following steps:

1. $TLC1$ and $TLC2$ are calculated using the method described in the previous section.

2. A region is defined in the adjacent lane (region C in Figure 6.1), where the length is the sum of the host vehicle length, the threat vehicle length and an extra safety buffer zone.

3. The position of the threat vehicle at the time between $TLC1$ and $TLC2$ is predicted. In Figure 6.1, x_{TLC1} and x_{TLC2} are the positions of the tracked vehicle at times $TLC1$ and $TLC2$. If the line between these two points intersects region C, the lane change manoeuvre would result in a collision and is considered dangerous with respect to this particular vehicle, otherwise not. If so, a flag is raised and the time to collision for this particular object is calculated. Note that no distinction needs to be made between vehicles coming from different directions.

4. Step 3 is then repeated for all tracked objects.

5. An important final step is to then check for objects in front of the host vehicle. If it is detected that there is a risk of collision with a leading vehicle, ELA will interpret any lane departure manoeuver as evasive action and therefore not intervene.

Furthermore, if the sensors have the capability of detecting solid lane markings, road barriers or even road edges this too could be incorporated into the algorithm, i.e., if a lane change manoeuvre is commenced in the direction of a solid lane marking ELA could also be activated and give a steering wheel torque, trying to prevent lane departure.

Next, if a flag was raised for any of the tested objects, the minimum time to collision for those objects together with an ELA warning flag is sent to the intervention module.

One appealing property of the road aligned coordinate system is that this kind of decision algorithms can be specified without having to regard the curvature of the road. If the road coordinate system was not used, we would have to, for each observed obstacle, judge its lane position based on its (ϕ, r)-coordinates. It also makes the accuracy of predicted positions x_{TLC1} and x_{TLC2} higher, since the assumption is that they will follow their lane, not their current tangent.

6.2 Intervention strategy

When a dangerous commenced lane change manoeuvre has been detected, we would like to apply some sort of steering wheel torque in order to interrupt/prevent lane departure. There are many alternative ways to do this. First of all, the duration of the intervention has to be chosen. There are basically two choices. The first is to apply a torque only a short time and in one direction, i.e., to simply steer away, in order to take the vehicle out of the imminent dangerous situation. A shorter intervention will minimize the interference with the driver, which is always desirable. The problem is that if the driver is not in control of the vehicle, it could end up in a new dangerous situation on the other side of the road.

The second choice is a longer intervention in order to first steer back into the safe lane and then place the vehicle straight ahead in the center of the lane. While this approach certainly involves a greater deal of interference with the driver, it will not leave the vehicle in a new dangerous situation. It will also contribute to a general feeling of safety and control of the system.

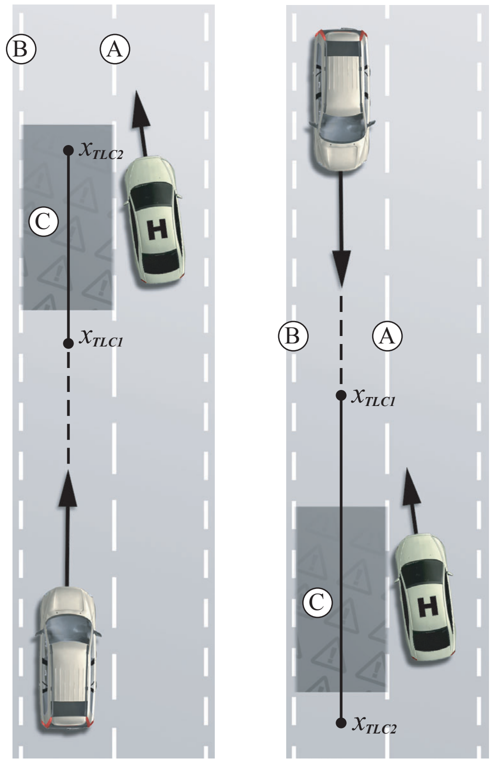


Figure 6.1: TLC1 and TLC2 are the times to cross lane A and lane B respectively, and x_{TLC1} and x_{TLC2} are the positions of the tracked vehicle at these times. A lane change manoeuvre is considered dangerous if another vehicle is predicted to enter region C during this time interval. The same strategy can be applied to vehicles in both directions.

6.2.1 Activation

The main activation signal is of course the ELA warning from the algorithm in Section 6.1. But we will also require some additional conditions to be satisfied. Since we are going to interfere with the steering of the vehicle, we have to be sure that the lane markings we see are not false ones. We will therefore require the road model confidence level to be high.

Next we put a requirement on the lateral position. If we have a positive lateral velocity we also require the lateral position to be higher than some threshold. This is to make sure we are actually on our way to leave the lane. Finally we also want the time to collision to be lower than some threshold. This is to make sure that the system will not intervene to early when there is still plenty of time for the driver to avoid the accident in a safe manner. Summarizing these conditions:

- ELA warning flag active
- Road geometry estimate confidence high
- Lateral offset threshold: If $\Psi_{rel} > 0$ we also require $y_{off} > y_{off}^{act}$. Similarly, if $\Psi_{rel} < 0$ we also require $y_{off} < -y_{off}^{act}$
- Time to collision: $TC < TC^{act}$

6.2.2 Deactivation

Note that we do not deactivate when the threat disappears, instead, to make sure the vehicle is in a safe state when we deactivate the controller, we will put some requirements on the position of the vehicle in the lane. First of all, the magnitude of the lateral position is required to be small, this is to ensure the vehicle is in the center of the lane. Secondly, we require the heading angle to be small, which will guarantee a small lateral velocity. Summarizing these two requirements:

- $|y_{off}| < y_{off}^{deact}$
- $|\Psi_{rel}| < \Psi_{rel}^{deact}$

There will also be a time limit on the intervention; the intervention is always deactivated after a certain time, even if the requirements above are not satisfied.

6.2.3 Driver interpretation

It was mentioned earlier that we try to detect if the driver is taking evasive action by searching for objects in the own lane in front of the host vehicle. What we really would like to know, is if the driver is still in control of the vehicle or not. For example, if the driver is steering resolutely in any direction, the intervention should be deactivated immediately. Also, if for example the indicators are activated or if the vehicle is accelerating hard could also be cues on that the driver is in charge of the situation.

In such cases the system should never intervene, since it is likely to be more distracting than supporting.

A typical case where we always want intervention is when the driver has released the steering wheel with both hands. A method for detecting this was proposed in [24].

6.2.4 Lateral control system

The system we are trying to control is approximately a quadruple integrator. We apply a steering column torque which is integrated to a steering wheel angular velocity which is integrated to a steering wheel angle. The steering wheel angle is then integrated to a vehicle heading angle which is again integrated to become a lateral position. Of course, this is a simplification, there is additional dynamics in all these steps, but the system is highly unstable.

Building a lateral controller for a vehicle is a well studied problem and lane following vehicles have been presented in the past [7, 24, 16, 15].

Volvo has experience, both with model based H_∞ control design and with simpler PID lateral control systems, and since the performance of an existing PD controller was satisfactory, we chose this concept for our control design.

To be able to use the low complexity controller, we have used the fact that the vehicle is equipped with a steering wheel angle sensor. This way we can split the controller into two nested loops. We first build a steering wheel angle controller, which can be tuned and verified separately. Then we build an outer control loop for the lateral position which is connected to the steering wheel loop. This method is based on the work presented in [24]. Two simple PD controllers can now be used to stabilize the system. Figure 6.2 shows the layout of the control system.

The control gains from [24] was used as a starting point, and then the controller was tuned to suit the ELA avoidance manoeuvres. All tuning of the lateral controller has been done manually so far. It would certainly be interesting to use a more systematic approach.

In addition to this, the outer loop is modified in order to take the time-to-collision value from the ELA algorithm in Section 6.1 into account. The time-to-collision enters as a ramp for the controller, i.e., the control gains starts at zero and are then increased linearly with time until they reach their final value. A short time to collision will make the gains increase fast while a slower increase rate will be used for a higher time-to-collision value.

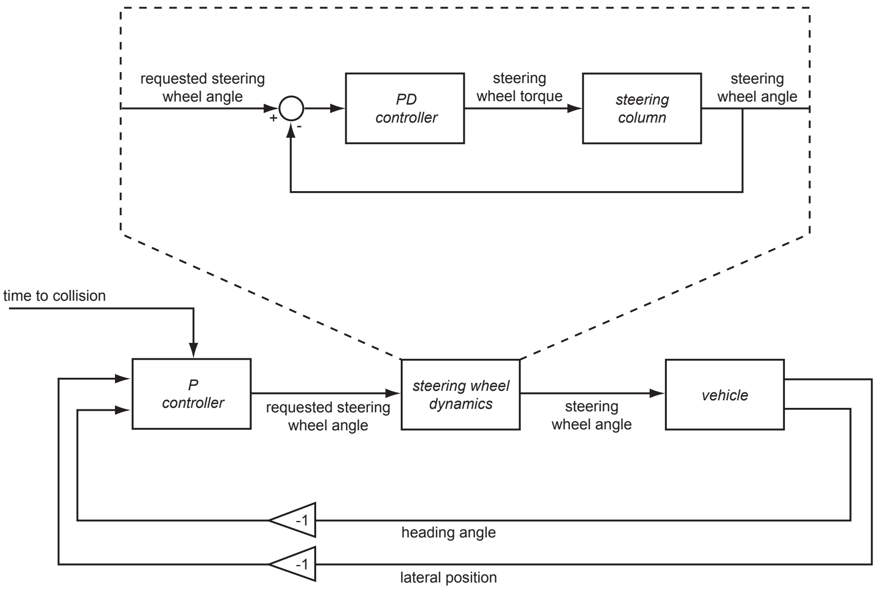


Figure 6.2: Layout of the lateral control system. There are no reference signals on the outer loop, we always use zero as control goal.

6.3 Host vehicle lane change hysteresis

In this section we will address a problem related to lane changes of the host vehicle. When changing lane, the lateral position reported by the signal processing unit will jump just as the center of the vehicle crosses the lane marking. For example, on a road of width 3 m, the lateral position would jump from 1.5 m to -1.5 m when changing lanes to the left. Now imagine the controller is active when this occurs. The controller will then suddenly start to steer towards the center of the "new" lane. This will not only generate an uncomfortable jerk, but the result is also that the vehicle will now steer towards the threat it was just trying to avoid, a behavior that could end in a disaster.

An additional problem with this is that the decision algorithm will also record a lane change, and will start looking for threats in the adjacent lanes with respect to the "new" lane, i.e., not in the lane the vehicle is currently moving into.

A third aspect relates to the definition of ELA. According to the function description in Chapter 3, ELA should not intervene if the vehicle has changed lane completely, instead we expect some sort of forward collision warning, mitigation or avoidance systems such as [17, 23] to deal with such situations. So the question is, when do we consider the vehicle to have changed lane. Typically, if we only require the center of the vehicle to have passed the lane marking, ELA will be disabled in many cases where the collision could have been avoided with a very undramatic steering intervention.

These issues can be resolved by introducing a sort of hysteresis on the lateral position. When crossing the lane marking, we keep measuring the lateral position from the "old" lane a little while longer, see Figure 6.3. We could for example require that all four wheels cross the lane marking before we start measuring from the new lane.

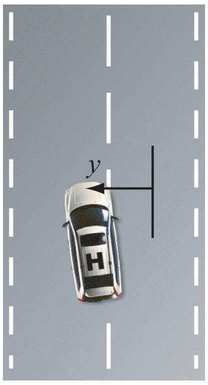


Figure 6.3: The y position is measured from the "original", old lane instead of the new lane.

For example, for a 1.6 m wide vehicle, the lateral position would increase to 1.5 + 1.6/2 m = 2.3 m and then jump to -1.5 + 1.6/2 m = -0.7 m.

6.4 Evaluation

6.4.1 Test scenario

This chapter will explain and give some results from the field tests that was carried out during the last phase of the project.

Figure 6.4 shows the test track that was used to tune and verify the ELA algorithm. A straight track of length 300 meters with two lanes of width 3.2 meters each was used.

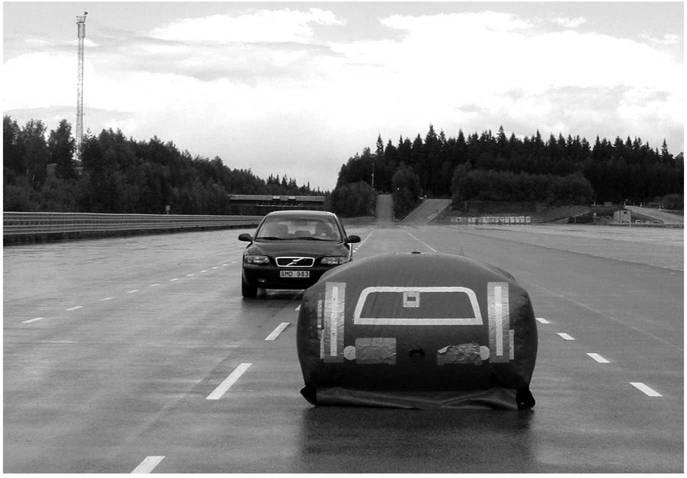


Figure 6.4: The test track and the inflatable dummy vehicle that was used in the ELA development.

An inflatable dummy vehicle was used to trig the intervention, also shown in Figure 6.4. It is the same type of test object that was used in for example the testing of the Collision Mitigation by Braking system described in [17]. The dummy is designed to resemble a real car, at least in the eyes of the sensors, but at the same time not damage the host vehicle in a collision.

The main restriction is of course that it is stationary which has limited the variation of test cases so far.

During a typical test, the dummy is representing a threat in the adjacent lane, for example an oncoming vehicle. The host vehicle is driving in the other lane, and as it approaches the dummy, a slow lane change manoeuver towards the threat is commenced.

Variations

The test case may seem simple, but there are still many ways the test can be varied. The most important parameters that can be changed are the following:

Host vehicle velocity The host vehicle velocity affects many aspects of the test. First of all, for higher speeds we need the sensors to pick up the obstacle at a much longer distance. The velocity also put different demands on the intervention module. At a high velocity, the torque that needs to be applied to the steering wheel in order to carry out the avoidance manoeuver is much lower.

Heading angle The heading angle, denoted by Ψ_{rel} in previous chapters, refers to the angle between the host vehicle and the lane and is highly connected to lateral velocity. If the magnitude of the heading angle is large, then the torque and time required to change the direction of the lateral velocity will be increased. Also, the time it takes to get back to the safe lane will be much longer.

Lateral displacement While the lateral displacement also affects the time it takes to get the vehicle back into the safe lane, it is also related to the lane change problem discussed in Section 6.3. If the vehicle gets to far into the other lane, according to the function description in Chapter 3, ELA is not supposed to intervene at all. Instead we expect some sort of forward collision system to be activated in such cases.

6.4.2 Test results

The system was tested and the different parameters from the previous section, velocity, heading angle and lateral displacement, was varied as systematically as possible. The general impression is that, for most cases, the system performance is satisfying. As long as the sensors detect the obstacle and the vehicle is on collision course, the decision algorithm always detects the threat and raises the ELA warning flag. In such cases the lateral control system is activated and has in this simple test scenario so far never failed to steer away from the threat unless the driver wishes to override the intervention by forcing the steering wheel in the other direction.

Furthermore, in almost all cases, the system is also able to align the vehicle straight ahead in the center of the original, safe lane again, before it lets go. Figure 6.5 shows a successful ELA avoidance manoeuver.

Many people have driven the system and most reactions are very positive. Many of the drivers felt that the intervention is very soft and not at all dramatic. Even drivers who before the test drive was afraid the intervention would be very dramatic agreed on this. Also, since the system brings the car back into the safe lane and leaves it in a safe position, it generally gave the drivers a positive feeling of security.

Many of the people who tested the system also believed in the usefulness of ELA as a safety system and could relate to incidents they had experienced themselves or knew someone who have had accidents that a system like this might have prevented.

Figure 6.5: Demonstration of a typical ELA intervention.

They also felt that the system fits well in maintaining the safety profile of the Volvo Car Corporation.

Where ELA needs improvement

Two main technical problems with ELA have been discovered so far. The first is in the intervention module. If the heading angle is to steep when the intervention is activated, the avoidance manoeuver requires a high yaw rate in order to reach the safe lane in time. During such high dynamic manoeuvres, we have learned that the vision system consequently loses track of the lane markings. So for a short period of time, perhaps 0.5 - 1 seconds, just when the control algorithm is active, we have no measurements of the lane markings. We have tried to solve this by just making time updates in the Kalman filter. The idea was that, for this short time, we could keep track of the heading angle and the lateral position, both used in the controller, just by studying the yaw rate sensor signal and the velocity. For some reason, though, this does not work, the signals drift away in no time. The result is that, in cases where the heading angle is too high, the vehicle is still able to avoid the obstacle, but it overshoots on the way back, ending up exiting the safe lane again on the other side. This needs further investigation, perhaps the bandwidth of the yaw rate sensor is not high enough, or we are using the signal incorrectly in some other way.

The second problem is also related to the sensors. Sometimes targets are detected too late or not at all, resulting in a collision with the dummy. This is most common during bad weather conditions such as rain, but can also happen in good visibility. Also note that late detection is an even greater problem for higher relative velocities, for example if the threat vehicle was an oncoming vehicle. If both vehicles are doing 90 km/h, a detection at 50 meters would only give one second for the system to respond which is too late in most cases.

People who succeeded in provoking the first error during a test drive generally felt very negative about this. However, in order to obtain such a high heading angle, a rather resolute and late lane change towards the threat has to be carried out. In such cases, perhaps the system should be totally disabled due to the fact that the driver is actually in control of the situation, as it was discussed in Section 6.2.3. Some people also experienced the second problem and actually collided with the dummy, naturally a rather unpleasant experience.

Another quite common reaction was that people wanted the system to steer back even though they had almost completed the lane change, which would actually disagree with the function description. After receiving that explanation, a majority of them did agree that activating some sort of forward collision system such as Collision Mitigation by Braking would be a better alternative. Some people, but actually surprisingly few, are naturally very skeptical towards systems that interfere with the driver, particularly the steering. They are also worried that it would be virtually impossible to test all possible scenarios required before a system like this could be put in production.

Chapter 7

Simulation environment

7.1 Introduction

Much of the development and verification of the different ELA components have been done in a dedicated simulation environment, which will be presented in this chapter. The development environment is a MATLAB software package which was developed during the initial phase of the project. The main purpose has been to speed up the development of ELA and allow for easier tuning of different parameters, both in the tracking system and in the decision algorithm.

The development environment has also made it easy to compare and evaluate different concepts, for example when it came to choosing geometric model in Section 5.4 or when evaluating the decision algorithm described in Section 6.1.

7.2 The software

7.2.1 Overview

The main idea of the development environment is to present the result of the different algorithms in a way that is easy to comprehend. As input, the software uses sensor signals. Then it runs the algorithm and records the output, both from the tracking system and from the decision algorithm. The result is then presented in a graphical user interface (GUI), together with the sensor signals and, if available, true positions of vehicles and lane markings. The software can also be run in batch mode, the output is then stored in a file instead of displayed in the GUI. The structure of the development environment is shown in Figure 7.1.

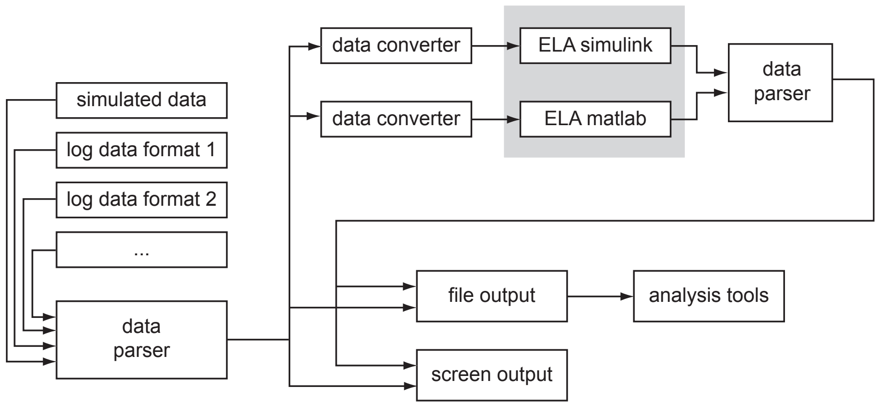


Figure 7.1: Structure of the ELA development software.

7.2.2 Data sources

The development environment accepts data from several different sources. There are for example different logging equipments which store data collected from the vehicle in different formats. To the left in Figure 7.1, the input data parser is illustrated. It also accepts input from a simulation tool that is used to generate artificial scenarios and sensor signals.

7.2.3 The simulation tool

An important part of the development environment is a simulation tool which can be used to customize and replay different scenarios. A basic script language is used to generate objects with different trajectories and also to generate the path of the host vehicle. All sensor signals can be distorted, for example missed measurements or false targets can be simulated, or measurement noise of some sort can be added. The simulation tool was very useful when developing the decision algorithm, but also during the initial development of the tracking system, before any real sensor data was available.

7.3 Matlab-Simulink connection

Most of the development has been done using MATLAB code. The main reason for this is that the time it takes to implement an idea or to make changes is very short compared other alternatives. When run on a normal PC, the processing speed of the algorithm has also been very good, always faster than real time.

In order to use the code in a vehicle, the MATLAB component Real Time Workshop was used. This implied that the code had to be implemented in Simulink. A separate Simulink interface was built to be able to run the Simulink version in the same development environment, see the shaded area in Figure 7.1. This way, the Simulink version could easily be verified by comparing the outputs.

7.4 Software operation

7.4.1 Graphical User Interface

The graphical user interface is shown in Figure 7.2. It can be used to adjust various settings, for example to select data sources or to select geometric model to be used in the Extended Kalman Filter.

The GUI then, as the algorithm runs, displays measurements of vehicles and lane geometry together with the output from the filter and the decision modules. It also displays the true values of these parameters if they are available. In Figure 7.2, the brighter lines and markings show the measured quantities while the darker figures show the output from the filter.

Much of the tuning of the filter and the algorithms has been done by simply studying the behavior of the output visually in the GUI.

7.4.2 Batch mode

The software can also be run in batch mode. This is convenient if a more systematic analysis is required. For example, batch mode makes it easy to run the algorithm on a data set many times while changing some tuning parameter a little bit between each run. The batch mode also makes it easy to carry out Monte Carlo simulations.

All settings are made in a script file where the desired tests are specified. When run in batch mode, the development software generates data files as outputs instead of displaying the results graphically. These data files can then be analyzed with a small set of data analysis tools that have also been developed.

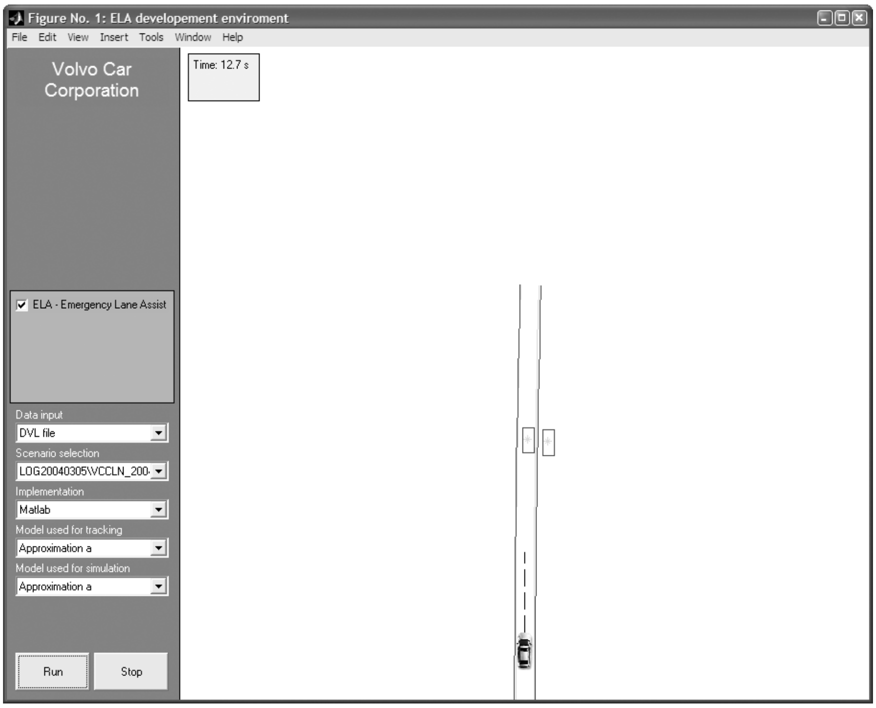


Figure 7.2: The graphical user interface that is connected to the development environment.

The batch mode was used, for example, to generate the tuning plots in Section 5.4.2.

Chapter 8

Concluding Remarks

8.1 Conclusions

In this thesis, the development of an automotive active safety function has been presented.

First, an evaluation method for such functions was developed. The method tries to estimate the function's impact on road traffic accident statistics and also component and development cost and then gives a score to each function based on the results. Functions with a high potential impact on the statistics and low component and development cost receives a high score. The method does not claim to be exact, but it can provide a help when trying to judge different active safety functions.

The evaluation method indicated that a function called "Emergency Lane Assist" should be developed. Emergency Lane Assist (ELA) is an active safety function which tries to prevent dangerous lane departure maneuvers by monitoring threats in adjacent lanes.

At an early stage it was noticed that the road geometry estimate, the curvature in particular, was not accurate enough. In the thesis, it has been shown that this estimate can be improved by using a centralized filter which also incorporates the motion of other vehicles. It was also shown that there might exist better alternatives than the established methods for doing this.

The centralized filter is based on a road aligned coordinate system, which also brings other advantages when it comes to modelling, prediction and developing applications.

A demonstrator was also built to evaluate the ELA concept and to test the performance of the tracking and decision modules. The variation of test scenarios has so far been quite small, but the performance of the demonstrator is promising. People who have been driving the system has generally felt very positive, although some drivers, but actually surprisingly few, were worried about liability questions.

ELA could potentially be associated with *less* liability problems than conventional lane guidance systems since the number of false alarms is reduced.

8.2 Future work

The first step is to start varying the test scenarios. So far, only one simple scenario has been tested and many other remain. Some examples are given here:

Moving objects Working with moving objects will the first step towards more realistic scenarios. The algorithm has of course been developed for movable objects, but it will certainly be a tougher challenge for the decision algorithm.

Multiple objects Running the system in a dense traffic environment will also be required, most of all to verify that the system does not give false alarms, especially during "active" driving with tight lane changes.

Curves First of all, curves are important for testing the performance of the lateral controller. Curvature information needs to be fed forward to the controller in order to avoid a stationary lateral control error, see [24]. Curves will also be required to see how the lane assignment accuracy described in Section 5.4.3 affects the performance of ELA.

Another thing that might be interesting to investigate is a more "physics based" decision algorithm. Inspired by the work in [17] where Jansson tries to find a state of unavoidable collision, a similar strategy could be studied in the ELA case. The difference is that ELA needs to intervene just *before* the collision unavoidable state. It would also be interesting to include and test some sort of algorithm to detect lane changes of leading vehicles.

Bibliography

[1] Vägutformning 94 version s-2. Technical report, Swedish National Road Administration, 1994.

[2] Automotive collision avoidance system field operational test. Technical report, National Highway Traffic Safety Administration, 2000.

[3] European road statistics 2004. Technical report, European Road Federation, 2004.

[4] N. H. Abel. Beweis der unmöglichkeit, algebraische gleichungen von höheren graden als dem vierten allgemein aufzulösen. *J. reine angew. Math.*, 1(65), 1826.

[5] S. Birch. Pre-Safe headlines S-Class revisions. *SAE Global Vehicles*, pages 15–18, January 2003.

[6] S. S. Blackman. *Multiple-Target Tracking with Radar Applications*. Artech House, Inc., 1986.

[7] S. Chaib, M. S. Netto, and S. Mammar. H_∞, adaptive, PID and fuzzy control: A comparison of controllers for vehicle lane keeping. In *Proceedings of the IEEE Intelligent Vehicles Symposium*, pages 139–144, Parma, Italy, June 2004.

[8] F. Dellaert and C. Thorpe. Robust car tracking using Kalman filtering and Bayesian templates. In *Proceedings of the SPIE conference on Intelligent Transportation Systems*, volume 3207, October 1997.

[9] E. D. Dickmanns and A. Zapp. A curvature-based scheme for improving road vehicle guidance by computer vision. In *Proceedings of the SPIE conference on Mobile Robots*, 1986.

[10] A. Eidehall and F. Gustafsson. Combined road prediction and target tracking in collision avoidance. In *Proceeding of IEEE Intelligent Vehicles Symposium*, pages 619–624, Parma, Italy, June 2004.

[11] D. M. Gavrila, J. Giebel, and S. Munder. Vision-based pedestrian detection: The protector system. In *Proceedings of the IEEE Intelligent Vehicles Symposium*, pages 13–18, Parma, Italy, June 2004.

[12] A. Gern, U. Franke, and P. Levi. Advanced lane recognition - fusing vision and radar. In *Proceedings of the IEEE Intelligent Vehicles Symposium*, Dearborn, MI, USA, October 2000.

[13] G. Grubb, A. Zelinsky, L. Nilsson, and M. Rilbe. 3D vision sensing for improved pedestrian safety. In *Proceedings of the IEEE Intelligent Vehicles Symposium*, pages 19–24, Parma, Italy, June 2004.

[14] F. Gustafsson. *Adaptive Filtering and Change Detection*. John Wiley & Sons, Ltd., 2000.

[15] T. Hessburg and M. Tomizuka. A fuzzy rule-based controller for automotive vehicle guidance. PATH research report UCB-ITS-PRR-91-18, California Partners for Advanced Transit and Highways, 1991.

[16] S. Ishida and J. E. Gayko. Development, evaluation and introduction of a lane keeping assistance system. In *Proceedings of the IEEE Intelligent Vehicles Symposium*, pages 943–344, Parma, Italy, June 2004.

[17] J. Jansson. Tracking and decision making for automotive collision avoidance. Licentiate thesis 965, Department of Electrical Engineering, University of Linköping, 2002.

[18] W. D. Jones. Keeping cars from crashing. *IEEE Spectrum*, pages 40–45, September 2001.

[19] T. Kailath, A. H. Sayed, and B. Hassibi. *Linear estimation*. Prentice Hall, 2000.

[20] R. E. Kalman. A new approach to linear filtering and prediction problems. *Transactions of the ASME-Journal of Basic Engineering*, 82:35–45, 1960.

[21] S. M. Kay. *Fundamentals of statistical signal processing*. Prentice Hall, 1993.

[22] A. Klotz, D. Hötzer, and J. Sparbert. Lane data fusion for driver assistance systems. In *Proceedings of the 7th International Conference on Information Fusion*, pages 657–663, Stockholm, Sweden, June 2004.

[23] K. Kodaka and J. E. Gayko. Intelligent systems for active and passive safety - collision mitigation brake system. In *Proceedings of the ATA 12th international symposium*, Parma, Italy, June 2004.

[24] J. Pohl and J. Ekmark. Development of a haptic intervention system for unintended lane departure. In *Proceedings of the 2003 SAE World Congress*, Detroit, MI, USA, March 2003.

[25] A. Polychronopoulos, U. Scheunert, and F. Tango. Centralized data fusion for obstacle and road borders tracking in a collision warning system. In *Proceedings of the 7th International Conference on Information Fusion*, pages 760–767, Stockholm, Sweden, June 2004.

[26] W. van Winsum, K. A. Brookhuis, and D. de Ward. A comparison of different ways to approximate time-to-line crossing (TLC) during car driving. In *Accident Analysis and Prevention 32*, pages 47–56, 2000.

[27] K. Weiss, N. Kaempchen, and A. Kirchner. Multiple-model tracking for the detection of lane change maneuvers. In *Proceedings of the IEEE Intelligent Vehicles Symposium*, pages 937–942, Parma, Italy, June 2004.

[28] W. W. Wierwille, L. A. Ellsworth, S. S. Wreggit, R. J. Fairbanks, and C. L. Kirn. Research on vehicle-based driver status/performance monitoring: Development, validation, and refinement of algorithms for detection of driver drowsiness. final report. Technical Report DOT HS 808 247, National Highway Traffic Safety Administration, 1994.

[29] Z. Zomotor and U. Franke. Sensor fusion for improved vision based lane recognition and object tracking with range-finders. In *Proceedings of the IEEE Conference on Intelligent Transportation Systems*, November 1997.

Tekn. lic. Dissertations
Division of Automatic Control and Communication Systems
Linköping University

P. Andersson: Adaptive Forgetting through Multiple Models and Adaptive Control of Car Dynamics. Thesis No. 15, 1983.

B. Wahlberg: On Model Simplification in System Identification. Thesis No. 47, 1985.

A. Isaksson: Identification of Time Varying Systems and Applications of System Identification to Signal Processing. Thesis No 75, 1986.

G. Malmberg: A Study of Adaptive Control Missiles. Thesis No 76, 1986.

S. Gunnarsson: On the Mean Square Error of Transfer Function Estimates with Applications to Control. Thesis No. 90, 1986.

M. Viberg: On the Adaptive Array Problem. Thesis No. 117, 1987.

K. Ståhl: On the Frequency Domain Analysis of Nonlinear Systems. Thesis No. 137, 1988.

A. Skeppstedt: Construction of Composite Models from Large Data-Sets. Thesis No. 149, 1988.

P. A. J. Nagy: MaMiS: A Programming Environment for Numeric/Symbolic Data Processing. Thesis No. 153, 1988.

K. Forsman: Applications of Constructive Algebra to Control Problems. Thesis No. 231, 1990.

I. Klein: Planning for a Class of Sequential Control Problems. Thesis No. 234, 1990.

F. Gustafsson: Optimal Segmentation of Linear Regression Parameters. Thesis No. 246, 1990.

H. Hjalmarsson: On Estimation of Model Quality in System Identification. Thesis No. 251, 1990.

S. Andersson: Sensor Array Processing; Application to Mobile Communication Systems and Dimension Reduction. Thesis No. 255, 1990.

K. Wang Chen: Observability and Invertibility of Nonlinear Systems: A Differential Algebraic Approach. Thesis No. 282, 1991.

J. Sjöberg: Regularization Issues in Neural Network Models of Dynamical Systems. Thesis No. 366, 1993.

P. Pucar: Segmentation of Laser Range Radar Images Using Hidden Markov Field Models. Thesis No. 403, 1993.

H. Fortell: Volterra and Algebraic Approaches to the Zero Dynamics. Thesis No. 438, 1994.

T. McKelvey: On State-Space Models in System Identification. Thesis No. 447, 1994.

T. Andersson: Concepts and Algorithms for Non-Linear System Identifiability. Thesis No. 448, 1994.

P. Lindskog: Algorithms and Tools for System Identification Using Prior Knowledge. Thesis No. 456, 1994.

J. Plantin: Algebraic Methods for Verification and Control of Discrete Event Dynamic Systems. Thesis No. 501, 1995.

J. Gunnarsson: On Modeling of Discrete Event Dynamic Systems, Using Symbolic Algebraic Methods. Thesis No. 502, 1995.

A. Ericsson: Fast Power Control to Counteract Rayleigh Fading in Cellular Radio Systems. Thesis No. 527, 1995.

M. Jirstrand: Algebraic Methods for Modeling and Design in Control. Thesis No. 540, 1996.

K. Edström: Simulation of Mode Switching Systems Using Switched Bond Graphs. Thesis No. 586, 1996.

J. Palmqvist: On Integrity Monitoring of Integrated Navigation Systems. Thesis No. 600, 1997.

A. Stenman: Just-in-Time Models with Applications to Dynamical Systems. Thesis No. 601, 1997.

M. Andersson: Experimental Design and Updating of Finite Element Models. Thesis No. 611, 1997.

U. Forssell: Properties and Usage of Closed-Loop Identification Methods. Thesis No. 641, 1997.

M. Larsson: On Modeling and Diagnosis of Discrete Event Dynamic systems. Thesis No. 648, 1997.

N. Bergman: Bayesian Inference in Terrain Navigation. Thesis No. 649, 1997.

V. Einarsson: On Verification of Switched Systems Using Abstractions. Thesis No. 705, 1998.

J. Blom, F. Gunnarsson: Power Control in Cellular Radio Systems. Thesis No. 706, 1998.

P. Spångéus: Hybrid Control using LP and LMI methods – Some Applications. Thesis No. 724, 1998.

M. Norrlöf: On Analysis and Implementation of Iterative Learning Control. Thesis No. 727, 1998.

A. Hagenblad: Aspects of the Identification of Wiener Models. Thesis no 793, 1999.

F. Tjärnström: Quality Estimation of Approximate Models. Thesis no 810, 2000.

C. Carlsson: Vehicle Size and Orientation Estimation Using Geometric Fitting. Thesis no 840, 2000.

J. Löfberg: Linear Model Predictive Control: Stability and Robustness. Thesis no 866, 2001.

O. Härkegård: Flight Control Design Using Backstepping. Thesis no 875, 2001.

J. Elbornsson: Equalization of Distortion in A/D Converters. Thesis No. 883, 2001.

J. Roll: Robust Verification and Identification of Piecewise Affine Systems. Thesis No. 899, 2001.

I. Lind: Regressor Selection in System Identification using ANOVA. Thesis No. 921, 2001.

R. Karlsson: Simulation Based Methods for Target Tracking. Thesis No. 930, 2002.

P-J. Nordlund: Sequential Monte Carlo Filters and Integrated Navigation. Thesis No. 945, 2002.

M. Östring: Identification, Diagnosis, and Control of a Flexible Robot Arm. Thesis No. 948, 2002.

C. Olsson: Active Engine Vibration Isolation using Feedback Control. Thesis No. 968, 2002.

J. Jansson: Tracking and Decision Making for Automotive Collision Avoidance. Thesis No. 965, 2002.

N. Persson: Event Based Sampling with Application to Spectral Estimation. Thesis No. 981, 2002.

D. Lindgren: Subspace Selection Techniques for Classification Problems. Thesis No. 995, 2002.

E. Geijer Lundin: Uplink Load in CDMA Cellular Systems. Thesis No. 1045, 2003.

M. Enqvist: Some Results on Linear Models of Nonlinear Systems. Thesis No. 1046, 2003.

T. Schön: On Computational Methods for Nonlinear Estimation. Thesis No. 1047, 2003.

F. Gunnarsson: On Modeling and Control of Network Queue Dynamics. Thesis No. 1048, 2003.

S. Björklund: A Survey and Comparison of Time-Delay Estimation Methods in Linear Systems. Thesis No. 1061, 2003.

M. Gerdin: Parameter Estimation in Linear Descriptor Systems. Thesis No. 1085, 2004.