

Towards environment perception for highly automated driving : with a case study on roundabouts

Raaijmakers, M.

Published: 26/06/2017

Document Version

Publisher's PDF, also known as Version of Record (includes final page, issue and volume numbers)

Please check the document version of this publication:

- A submitted manuscript is the author's version of the article upon submission and before peer-review. There can be important differences between the submitted version and the official published version of record. People interested in the research are advised to contact the author for the final version of the publication, or visit the DOI to the publisher's website.
- The final author version and the galley proof are versions of the publication after peer review.
- The final published version features the final layout of the paper including the volume, issue and page numbers.

[Link to publication](#)

Citation for published version (APA):

Raaijmakers, M. (2017). Towards environment perception for highly automated driving : with a case study on roundabouts Eindhoven: Technische Universiteit Eindhoven

General rights

Copyright and moral rights for the publications made accessible in the public portal are retained by the authors and/or other copyright owners and it is a condition of accessing publications that users recognise and abide by the legal requirements associated with these rights.

- Users may download and print one copy of any publication from the public portal for the purpose of private study or research.
- You may not further distribute the material or use it for any profit-making activity or commercial gain
- You may freely distribute the URL identifying the publication in the public portal ?

Take down policy

If you believe that this document breaches copyright please contact us providing details, and we will remove access to the work immediately and investigate your claim.

Towards Environment Perception for Highly Automated Driving

with a case study on roundabouts

PROEFSCHRIFT

ter verkrijging van de graad van doctor aan de Technische
Universiteit Eindhoven, op gezag van de rector magnificus
prof.dr.ir. F.P.T. Baaijens, voor een commissie aangewezen
door het College voor Promoties, in het openbaar te
verdedigen op maandag 26 juni 2017 om 16:00 uur

door

Marvin Raaijmakers

geboren te Heerlen, Nederland

Dit proefschrift is goedgekeurd door de promotoren en de samenstelling van de promotiecommissie is als volgt:

voorzitter: prof.dr. J.J. Lukkien
1^e promotor: prof.dr. M.T. de Berg
copromotoren: dr. H.J. Haverkort
Dr.-Ing. M.E. Bouzouraa (Audi AG)
leden: Prof.Dr.-Ing. C. Stiller (KIT)
prof.dr. M.J. van Kreveld (UU)
prof.dr. E.R. van den Heuvel
prof.dr.ir. M. Steinbuch

Het onderzoek dat in dit proefschrift wordt beschreven is uitgevoerd in overeenstemming met de TU/e Gedragscode Wetenschapsbeoefening.

This thesis is the result of research I did as an employee of Audi AG. This research was supported by the German Federal Ministry for Economic Affairs and Energy, in the context of the research initiative URBAN, on the basis of a decision by the German Bundestag.

© Marvin Raaijmakers. All rights are reserved. Reproduction in whole or in part is prohibited without the written consent of the copyright owner.

Cover photograph: Wangspitze (2420 m), Austria, by Marvin Raaijmakers.
Printing: Dereumaux, Eindhoven

ISBN: 978-90-386-4285-7

Acknowledgements

This thesis is the result of research I did as an employee of Audi AG. I would like to thank Sayed Bouzouraa for supervising my work at Audi and for the valuable discussions and feedback. Thanks go to Herman Haverkort for supervising this work on behalf of the Eindhoven University of Technology, our enjoyable discussions and for providing different viewpoints for my research. My gratitude goes to professor Mark de Berg for being the formal academic supervisor of my research and providing valuable feedback on my thesis.

I would also like to thank professors Christoph Stiller, Marc van Kreveld, Edwin van den Heuvel and Maarten Steinbuch for their interest in my research topic and their willingness to be part of my doctorate committee.

My work took place within the Audi department for pre-development of automated driving systems. I am happy to have been a part of this very open, cooperative and innovative team led by Karl-Heinz Siedersberger and Klaus Verweyen. I would like to thank all of my great colleagues for making me have such a great time and for supporting me during my research. Special thanks go to colleagues Max Schmidt, Frank Dierkes, Tobias Weiherer, Martin Kellner, Christopher Demiral, Caterina Vitadello, Waldemar Winter and Jens Storz for assisting me during my experiments, helping me with technical problems and the fruitful discussions with them. Furthermore, I would like to express my gratitude to Ulrich Hofmann and Stephan Neumaier for their valuable discussions with me and for sharing their highly appreciated experience and expertise within the field of automated driving.

Thanks go to Virgilio Gruppelaar who developed, as part of his internship, a set of scripts that I used for the evaluation of my roundabout perception algorithms.

And last, but certainly not least, big thanks go to my uncle Wil Koenen for being the person who triggered my fascination for science and engineering.

Summary

Highly automated vehicles are defined as automated vehicles where the human driver does not have to monitor the vehicle's behaviour and its environment. This requirement makes the environment perception system one of the most crucial components of such a vehicle. This thesis addresses some of the challenges in the development of environment perception methods for highly automated driving.

Research objectives. In order to realize highly automated driving in urban environments, a lot of research still has to be done on making environment perception more robust. The rather limited capabilities of the current state of the art sensor-based environment perception for urban environments, motivates research objective 1: *Improve the capabilities of sensor-based perception in urban environments using close-to-market sensors.*

However, it is unrealistic to assume that in the near future all elements in the environment, relevant for highly automated driving, will be perceived based on only sensor data. Instead, detailed digital maps will play an important role for the perception of those elements for which no robust sensor-based methods are available yet. Thus, a near-future perception system for highly automated driving must be able to combine sensor-based perception with data from digital maps. This requirement defines objective 2: *Develop methodology to combine sensor-based perception with data from an a priori map in order to perceive all relevant elements.*

Unfortunately all automated vehicles demonstrated thus far assume the contents of a digital map to be correct. This may be acceptable for a partially automated system where a human driver has to monitor the system, but not for a highly automated vehicle. Since this problem has been rarely a topic of scientific research, although it is considered to be very significant, it forms objective 3: *Formulate and investigate the problem of verifying the correctness of a priori map data before the vehicle acts on it.*

The three research objectives described above are too hard to investigate in their full generality within a single thesis. Therefore these objectives are studied in a specific context, namely the perception of roundabouts. Roundabouts form an increasingly important part of urban road systems, and they have very different characteristics from regular intersections. Despite their increasing importance, the perception of roundabouts has not been studied so far by the automated driving research community. Hence, the study of roundabouts not only serves as a test bed for the research objectives above, it is also interesting in its own right. Therefore research objective 4 is: *Develop methods for the perception of roundabouts based on sensor data and a priori digital map data.*

Contributions of the thesis. One of the problems that arises if we want to replace the fully-map-based perception of urban roads by sensor-based perception, is ambiguity about the structure of the road. This problem is addressed in the thesis where a multi-hypothesis road representation is presented as a contribution to research objective 1. Instead of letting the perception system choose the most likely hypothesis about the road and send this hypothesis to the behaviour generation system, the idea behind the presented representation is to send the set of all plausible road hypotheses.

A contribution to both research objectives 1 and 4 is made by the presentation of a method

for detecting central islands of roundabouts using a close-to-market laser scanner. An extensive evaluation of this method is done where not only the detection rate is evaluated, but also the quality of the input data, the error distributions of the detected central islands, and causes of missed detections.

Based on the central-island detector, a method is presented for the perception of all relevant elements of a roundabout, which is a contribution to research objectives 2 and 4. The output of the central-island detector and a road marking detector are fused with a detailed a priori digital map to achieve a complete perception of the roundabout. The method localizes itself relative to this digital map without the need of a precise GNSS or additional landmarks in the map.

Verifying the correctness of a digital map, before the automated vehicle acts on it, is a topic discussed in the thesis which contributes to research objective 3. First, possible requirements on map verification for highly automated driving are discussed. Then the map verification problem is formally specified in a probabilistic way. This specification is followed by a proposal for a generic map verification model, and a description of how it can be applied to verify the correctness of the map of a roundabout's central island (thus it also contributes to objective 4). The presented verification model is analysed based on this application and its discussion identifies the difficulties and challenges that arise for approaches like the proposed verification method.

Contents

1	Introduction	1
1.1	Introduction to Automated Driving	1
1.2	Motivation	4
1.3	Related Work	6
1.3.1	Infrastructure Perception and the Use of Digital Map Data	6
1.3.2	Roundabout Perception and Circle Detection	7
1.3.3	Map-Relative Localization	8
1.4	Research Objectives	9
1.5	Main Contributions and Outline of the Thesis	10
2	Conventions, Definitions and Background Knowledge	13
2.1	The Environment Coordinate System	13
2.2	The Roundabout Model	14
2.3	Laser Scanner	15
2.3.1	Laser Scanner Principles	15
2.3.2	Specifications of Used Laser Scanner	15
2.4	Notation of Probabilistic Concepts	16
3	A Multi-Hypothesis Road Representation	19
3.1	Motivation	19
3.2	A Single-Hypothesis Roadway Model	20
3.3	Multi-Hypothesis Representation	21
3.4	Recommendations for Future Research	25
4	Roundabout Central Island Detection	27
4.1	Introduction	27
4.2	Problem Statement	27
4.3	Algorithm Overview	30
4.4	Semi-Convex Partitioning	33
4.4.1	Convexity and Semi-Convexity	33
4.4.2	Semi-Convexity of Sequences on the Central Island	34
4.4.3	Semi-Convex Partitioning Algorithm	37
4.5	Curvature-Based Partitioning	38
4.6	Evaluation	41
4.6.1	Evaluation Goals and Experimental Setup	41
4.6.2	Input Data Quality	42
4.6.3	Detection Performance	44
4.6.4	Detection Rate Depending on the Distance to the Central Island	49
4.6.5	Error of Fitted Circles	50
4.7	Conclusions and Recommendations for Future Research	57

5	Roundabout Perception Supported by A Priori Map Data	59
5.1	Introduction	59
5.2	Digital Map Data	59
5.3	The Roundabout Perception Algorithm	60
5.3.1	Algorithm Overview	60
5.3.2	Stages of Operation	61
5.3.3	The Roundabout Tracker	62
5.3.4	Roundabout State Hypothesis Derivation	65
5.3.5	Computing the In-Roundabout Probability	65
5.4	Evaluation	66
5.5	Conclusions and Recommendations for Future Research	68
6	Map Verification	73
6.1	Introduction	73
6.2	Requirements on Map Verification	73
6.3	Problem Statement	76
6.4	A Generic Probabilistic Map Verification Model	77
6.5	Implementation of the Verification Model for Central Islands	80
6.6	Integral Computation	82
6.6.1	Partitioning the Integral's Space into a Grid	83
6.6.2	Integration over A	84
6.6.3	Integration over $A \cap B$	89
6.6.4	Integration over $A \cap B \cap V$	97
6.7	Analysis of the Method	98
6.7.1	Analysis of the Central Island Implementation	98
6.7.2	Evaluation	105
6.7.3	Causes of Unsatisfactory Results	107
6.7.4	Why the problems are hard to solve	109
6.8	Conclusions and Recommendations for Future Work	110
7	Conclusions and Recommendations for Future Research	113
	Bibliography	117
	A Roundabouts Used for Evaluation	123
	B Good-to-Total Central Island Detection Ratio	127
	C $\beta_{i,j}$ Sets	131
	Curriculum Vitae	141
	Index	143

Chapter 1

Introduction

1.1 Introduction to Automated Driving

The first ideas about automated driving date back to the 1950s. According to a survey by Fenton [Fen70], the General Motor Corporation was the first to start research on an automatic vehicle system for “speed control” and “variable spacing”. The vision of automated driving was also picked up by the public, as shown in the 1956 advertisement in figure 1.1 by America’s Independent Electric Light and Power Companies. This advertisement also describes what the advantages of automated driving may be, which are still the objectives of its development today: increased safety, more comfort, being able to spend your time on things different from driving and the reduction of traffic jams.

Due to the limited capabilities of computers and imaging sensors, the earliest concepts, such as those by the General Motors Company, mainly relied on infrastructure modifications (e.g. magnetic cables integrated in road) for the guidance of the vehicles [Fen70]. With the increasing performance of computers and digital imaging sensors, significant progress was made in the late eighties. One important milestone was the demonstration of the VaMoRs vehicle [Sie03] in 1987 by the group of Ernst Dickmanns at the Bundeswehr University Munich. VaMoRs drove automated on a part of the German A92 motorway, which had not yet been released for public traffic, based on computer vision for the recognition of the road.

Another important event was the Urban Challenge organized by DARPA in 2007 [UAB⁺08], where automated vehicles without any occupants had to drive 97 km through an urban environment (closed for the public), interacting with other moving vehicles and obeying the California Driver Handbook. In this challenge, 53 teams from around the world had the opportunity to show the automated driving capabilities of their cars. Only 11 of these teams qualified for the final event of the challenge. Most of the finalists, such as the winning vehicle of the Carnegie Mellon University [UAB⁺08], based their system mainly on high-resolution laser scanners, radars and high-precision GPS in combination with a digital map provided by the event’s organization. After the challenge, many of the participating research groups continued the research and development of their vehicles, thus forming the basis of automated driving demonstrations that followed the challenge [BDF⁺14].

The architecture of automated vehicles comprises three main components, as illustrated in figure 1.2: sensors, a data processing component and actuators. The data processing component is usually divided in three subcomponents: an environment perception system, a behaviour planning system and a vehicle control system. Sensors generate measurements, of elements of the environment or the vehicle itself. First, the *environment perception system* processes these measurements in order to detect and quantify all elements of the environment that are relevant for the automated functionality. Optionally, the perception system may also use a priori knowledge about the environment such as a digital map. The result is an environment representation that serves as the input to the behaviour planning system. The *behaviour planning system* decides how the vehicle should behave in terms of manoeuvres and plans the vehicle’s trajectory and velocity

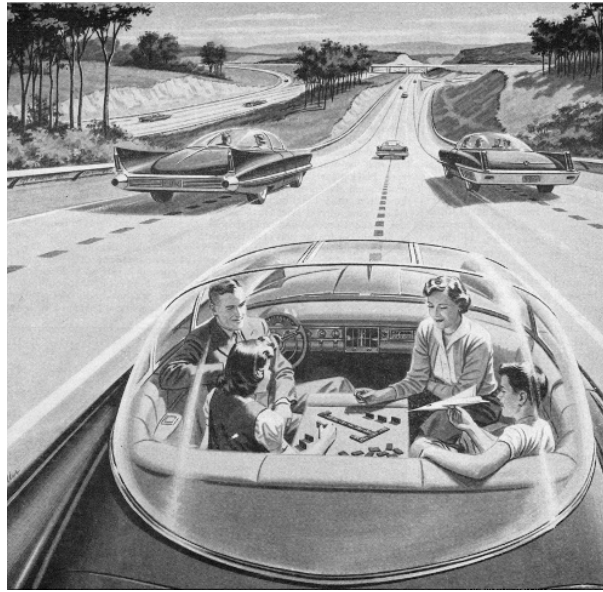


Figure 1.1: An image from a 1956 advertisement that was accompanied with the text “One day your car may speed along an electric super-highway, its speed and steering automatically controlled by electronic devices embedded in the road. Travel will be more relaxing and enjoyable. Highways will be made safe – by electricity! No traffic jams... no collisions... no driver fatigue.” [Ame56]

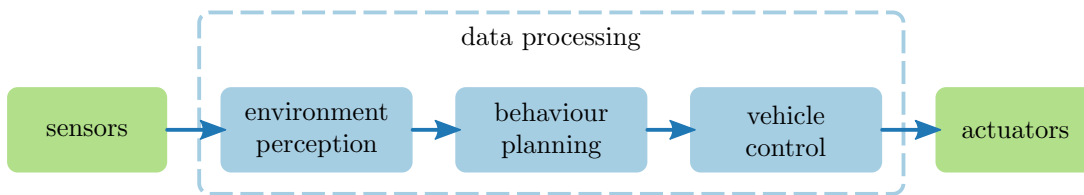


Figure 1.2: The major components in which an automated driving system is commonly subdivided.

profile. Finally, the *vehicle control system* translates this planned trajectory and velocity profile into the actual movement of the vehicle through its actuators.

Although to date there are no production vehicles capable of automated driving, some of the technology that was developed for the ultimate goal of automated driving found its way to production vehicles as part of driver assistance systems and active safety systems. By combining electronic throttle control, an electronic braking system and radar sensors that allow measuring the distance to an upfront vehicle and its velocity, adaptive cruise control made it possible to automate the longitudinal control of a vehicle. Being introduced in production vehicles in 1999, this made adaptive cruise control the first form of vehicle automation that responds to its environment. A few years later, the same sensors and actuators were used in production vehicles for long-range collision mitigation systems. The computer vision methods for road recognition were used for lane departure warning. As the performance of these methods increased, they were used for active lane keeping assistance. In 2013 the first production vehicle was introduced that features traffic jam assistance. This system combines the automated longitudinal and lateral control of the vehicle in traffic jams and allows the driver to take his/her hands of the steering wheel while the system is active for velocities up to 30 km/h. A comprehensive overview of the development of driver assistance systems is given in a review by Bengler et al. [BDF⁺14].

With the introduction of traffic jam assistance, the boundaries between driver assistance systems and automated driving became more vague. Although the driver can take his/her hands of the steering wheel and let the vehicle drive on its own, he/she still has to monitor the behaviour

Level	Name	Narrative definition	Vehicle control	Monitoring	Fallback	SAE level
0	Driver Only	The human driver performs full-time both lateral and longitudinal control of the vehicle.	human driver	human driver	human driver	No Automation
1	Assisted	The system performs either lateral or longitudinal control of the vehicle while the human driver is in control of the other aspects of the driving task.	human driver and system	human driver	human driver	Driver Assistance
2	Partially Automated	The system performs lateral and longitudinal control of the vehicle, while the human driver must monitor the system and environment and must intervene when the system's capability limits are reached.	system	human driver	human driver	Partial Automation
3	Highly Automated	The system performs all aspects of driving and detects when it reaches the limits of its capabilities such that the human driver is given enough time to intervene.	system	system	human driver	Conditional Automation
4	Fully Automated	The system performs all aspects of driving and brings itself to a safe state when it detects that the limits of its capabilities will be reached.	system	system	system	High and Full Automation

Table 1.1: A summary of the automation levels as defined by BaSt [GAA⁺12]. The “Monitoring” column describes whether the human driver or the system itself monitors the environment and behaviour of the system. The “Fallback” column indicates whether the human driver or the system takes control of the vehicle when the limits of the system’s capabilities are reached. The corresponding names according to the SAE definitions [SAE14] are also listed.

of the system. In order to eliminate confusion and to improve understanding across multiple disciplines (e.g. engineering, legal, media and public discourse), the German Federal Highway Research Institute (abbreviated as BaSt) and SAE International (Society of Automotive Engineers) defined different levels of automation [GAA⁺12, SAE14]. Both the BaSt and SAE taxonomies are summarized in table 1.1. Note the difference in naming for the automation levels between those two taxonomies. This thesis follows the classification and terminology defined by BaSt. Furthermore, in this work the popular terms “autonomous vehicle” or “self-driving vehicle” are avoided.

The current state of the art allows us to build partially automated vehicles where the human driver still has to monitor the system and must be prepared to take over control of the vehicle. The next step will be the development of highly automated driving where the human driver no longer has to monitor the system and will be given sufficient time to take over the vehicle once the system detects the limits of its capabilities. While this may sound like a minor difference, the step from partial automation to high automation requires an enormous increase in robustness of the perception system. This is one of the challenges researchers within the field of automated driving are currently faced with.

The first highly automated driving systems will only be able to drive in relatively simple environments and scenarios, such as highway traffic jams. A research challenge is to improve environment perception in a such way that it can handle more complex environments and situations with the ultimate goal of perceiving the highly dynamic and diverse urban environments.

1.2 Motivation

Since the beginning of this decade there have been a number of demonstrations of research vehicles that drove automated in real traffic:

- In 2010, as part of a project called “Stadt-pilot” [WSM10] (German for “city pilot”) of the Braunschweig University, research vehicle Leonie drove automated on a 2.5 km long part of the inner ring road around the Braunschweig (Germany) city centre [SWLM11]. On this two-lane road the vehicle was able to make lane changes, merge into moving traffic, and performed manoeuvres at signalized intersections that did not require consideration of right of way.
- In July 2013, research company VisLab demonstrated their vehicle BRAiVE [BCD⁺15]. During this demonstration in Parma (Italy), BRAiVE drove automated on highways and urban roads. Besides simple driving manoeuvres, the vehicle had to cope with roundabouts, junctions, crosswalks and traffic lights.
- In August 2013 Daimler’s research vehicle Bertha [ZBS⁺14] drove automated from Mannheim to Pforzheim (Germany). The course had a total length of 103 km, included rural roads and passed through 23 small villages and major cities. The vehicle demonstrated its abilities to handle intersections with and without traffic lights, roundabouts, and narrow passages with oncoming traffic.
- In July 2014 Ulm University demonstrated their automated vehicle by letting it drive along a 5 km route in an urban environment [KNW⁺15]. The route included the universities’ campus and contained traffic lights, crosswalks, intersections, roundabouts and a lot of pedestrians and other vulnerable road users.
- In 2015 BMW Group Research and Technology published a paper [ARB⁺15] about their experience of testing automated vehicles on German highways since they started working on this in 2011. They developed a prototype vehicle that is able to drive automated on highways up to 130 km/h, can perform lane changes and can merge into moving traffic when entering the highway via a ramp.
- One of the most publicly noticed projects is Google’s Self-Driving Car Project that started in 2009. At the time of writing, Google has a fleet of 58 automated vehicles which have driven in total around 3.4 million testing kilometres in “autonomous mode” (2.1 million kilometres were driven in “manual mode”) in various scenarios on the streets of Mountain View (California), Kirkland (Washington), Phoenix (Arizona) and Austin (Texas) [Goo16a]. Unfortunately, there are no scientific publications about the methods that are used in this project and therefore it will not be a part of the discussion of related work in this thesis.

Although the ultimate goal of the automated driving research area is to develop a fully automated car, none of the vehicles developed thus far can be classified as a fully or even highly automated vehicle. Each of the vehicles still requires a safety driver to monitor the behaviour of the vehicle to guarantee safe operation. This is because the environment perception methods are not robust enough to ensure all relevant elements of the environment are correctly perceived, and because the systems are not capable yet of recognizing their own driving limits early enough to inform the driver to take over control or to automatically bring the vehicle in a safe state.

All of the vehicles listed above, one more than the other, rely on a priori digital maps for perception of the environment. A digital map may be used as a redundant source of information to improve the robustness of a sensor-based perception system. However, the main reason thus far for using a digital map is to obtain information about certain elements of the environment for which no sensor-based perception methods exist yet. The inherent problem is that the contents of a digital map may be outdated or otherwise erroneous. If such outdated data are used by the vehicle, then it will have an erroneous perception of the environment, that may result in dangerous behaviour.

Current state-of-the-art sensor-based perception methods will probably be sufficiently robust for the first highly automated driving systems that are to be introduced in production vehicles, since these first systems will be restricted to operate in simple environments. However, much research has to be done in order to perceive more elements (e.g. intersections) in complex environments based on sensors. Therefore, if digital maps can be safely used by an automated vehicle for the perception of certain environment elements, then this will accelerate the introduction of highly automated driving, in more complex environments, to production vehicles.

The importance of digital maps is also recognized by the automotive industry. A striking example of this is the joint acquisition of digital map company Here (formerly known as Nokia Here, of which the origins lie in Navteq) by car manufacturers Audi AG, BMW Group and Daimler AG in 2015: the importance of digital maps for automated driving was one of the main motivations for this acquisition.

With the aim of contributing to an early introduction of highly automated driving in urban environments, this thesis addresses the following topics:

- Improving the capabilities of sensor-based environment perception in urban environments using close-to-market sensors.
- Combining sensor-based perception with data from an a priori digital map in order to perceive all relevant elements of the environment.
- Robust use of not fully reliable a priori digital map data for environment perception.

These topics are treated within the context of roundabout perception. Roundabouts can be considered a special type of intersection mainly used in urban environments. Compared to regular intersections, roundabouts are known to increase the traffic flow and decrease the number of accidents [BSU09]. This is because at roundabouts the number of points where vehicle trajectories can cross or merge, is much less than for regular intersections (single-lane roundabouts even have zero points where trajectories can cross), the vehicles on the circulatory roadway all drive in the same direction and the right-of-way rules at a roundabout are simple. Furthermore, roundabouts offer better visibility between the traffic participants and the structure of a roundabout is easier to understand than that of a regular intersection. Not only do these properties of a roundabout lead to fewer accidents, but they also indicate that environment perception for automated driving may be simpler at a roundabout than at a regular intersection. Therefore, roundabouts have the potential for allowing highly automated driving earlier than regular intersections. This can be of high importance, since cities wanting to promote highly automated driving in its early stages, could consider this when planning their infrastructure. Since main roads in cities (e.g. ring roads) are the least complex urban environments, an early introduction of urban highly automated driving may be realized by creating a main road network where the roads are connected through roundabouts.

Considering the above arguments on roundabouts, it may not come as a surprise that, within the research field of automated driving, roundabouts have already received some attention. In 2011 Pérez Rastelli et al. [PRMdPV11] published a paper on vehicle path planning in roundabouts. Muffert et al. [MPF13] presented an approach that tracks other vehicles on a roundabout, makes a prediction about them and finally decides whether it is safe to enter the roundabout. Also the work by Gritschneider et al. [GHT⁺16] and Okumura et al. [OJK⁺16] focuses on decision making at roundabouts. Dickmann et al. [DKH⁺15] discuss future requirements on automotive radar sensors for perception of other vehicles at roundabouts. Furthermore, the automated vehicles Bertha [ZBS⁺14], BRAiVE [BCD⁺15] and the Ulm vehicle [KNW⁺15] were able to cope with roundabouts on their route. However, for the perception of the roundabout itself, all of these approaches rely solely on a detailed digital map that is assumed to be correct. Therefore, roundabout perception can be identified as a research gap that is relevant for automated driving.

1.3 Related Work

1.3.1 Infrastructure Perception and the Use of Digital Map Data

Road perception has been a research topic since the first automated driving projects. For example, in the late 1980s Dickmanns et al. [DMC90] demonstrated their camera-based road perception method in an automated vehicle. Their vehicle was able to drive automated on highways and could perform lane keeping and lane change manoeuvres. The method was able to perceive the geometry of the ego lane (i.e. the lane in which the own vehicle is driving) and neighbouring lanes. Since then, much effort has been spent on making lane perception algorithms more robust. As a result, they formed the basis of the perception systems for lane keeping assistance that can be found in production vehicles today [FPR⁺13].

A good description of the state of the art of road and lane perception is the survey by Hillel et al. [HLLR12]. One important observation they make, is that most research on sensor-based road perception has been done for relatively simple road structures. Most of the work is limited to the perception of the ego lane and neighbouring lanes on highway-like road geometries with very little curvature. Higher complexity in roads can occur in the form of complex lane geometries, splitting or merging of lanes and intersections. This makes it more challenging for a system to perceive the geometry of the lanes and understand the structure and topology of the road. As Hillel et al. [HLLR12] state, vehicles that demonstrated automated driving on complex roads relied on the use of a priori digital map data for the perception of the road. Consequently, they identified the sensor-based perception of complex roads as a main research gap.

With the demonstration of the BRAiVE vehicle, Broggi et al. [BCD⁺15] claim that the “test demonstrates that it is not necessary to use precise 3D maps: world reconstruction, including both road features and road users, can be performed during the driving”. However, the validity of this claim is highly questionable since they do not explain how their vehicle perceived the intersections and roundabouts that were on its route.

The research vehicles for urban environments, Leonie [WSM10], Bertha [FPR⁺13], the Ulm car [KNW⁺15] and Google’s Self-Driving Car [Goo16b] are known to use an a priori digital map for the “perception” of the road. Bertha and the Ulm car both detect road markings in camera images, but instead of using these features for road recognition, the features are used for localization in the digital map. Even on the highway, where road perception is considerably easier than in urban environments, the BMW vehicle [ARB⁺15] relied on a priori digital maps for the perception of the road.

Based on experience with the BMW vehicle, Aeberhard et al. [ARB⁺15] recognize that outdated digital maps form a serious problem for their system. They state that, in the future, an automated driving system must be able to detect that a map is not valid anymore and that their current research is focussed on this topic (no results were published to date).

To date, automatic map verification has been a topic of interest mainly for map makers with the goal of detecting whether a map is outdated. Internet connectivity of navigation devices allow navigation map makers to detect map errors, caused by changes in the road network, by processing GNSS (global navigation satellite system) trajectories of a fleet of vehicles (see for example the method presented by Ekpenyong et al. [EPBB09]). Unfortunately, since these methods rely on trajectories of many vehicles, which can also be outdated, they are not suitable for automated vehicles to verify the correctness of the map’s current state.

Zinoune [ZBIG12b, Zin14] presented a method in which a vehicle verifies the correctness of a digital map based on its own GNSS trajectory. This method detects errors in the map, but this can only be done after the vehicle passed the part of the map that was incorrect. This makes the method not suitable for the verification of map data before the system uses the map data. Furthermore it only allows a verification of the centre line of the road, whereas an automated vehicle requires a much more detailed geometric description of the road.

A method to detect errors in the map before the vehicle has reached an erroneous location in the map, was proposed by Hartmann et al. [HGS⁺14]. They defined the map verification problem as a classification problem for which they trained a neural network that outputs a probability that

the map contains an error. Their digital map is a traditional navigation map where the road's geometry is described by its centreline. As a ground truth centreline they recorded the vehicle's trajectory with DGPS (differential GPS). For each time instant in the training data, the map was labelled as incorrect if and only if within a 100 m range ahead of the vehicle the Hausdorff distance between the ground truth centreline and the map centreline exceeded a maximum value. Using these labels, the neural network was trained to classify the correctness of the map based on sensor data from a near infra-red camera, radars and the output of an optical lane recognition system. One disadvantage of the presented method is that it can only recognize that the map contains an error somewhere between 0 m and 100 m ahead of the vehicle. It cannot identify the exact location of the error. Another drawback of the method is its low flexibility: the neural network is trained for a specific combination of the map's uncertainty (which is expressed as the maximum Hausdorff distance in order to classify the map as correct), lookahead (100 m in the presented paper), sensor set and individual sensor characteristics. Although the authors motivate their work by the fact that automated vehicles require detailed digital maps and that for those vehicles unnoticed errors may lead to dangerous situations, the method they presented is designed for the verification of traditional centreline maps. However, highly automated vehicles require a much more detailed geometric description of the road.

To the best of my knowledge, there are no scientific publications on onboard map verification for automated vehicles to date. Therefore this can be considered a research gap that has been recognized [ARB⁺15, HGS⁺14] to be significant.

1.3.2 Roundabout Perception and Circle Detection

Sensor-based perception of roundabouts for automated driving has not been a topic of scientific research thus far. In this section we provide an overview of literature related to roundabout detection for other kinds of applications and literature related to circular shape detection.

A number of approaches [ZBIG12a, FCR09] exist that detect missing roundabouts in a digital map, based on the trajectory driven by a vehicle. Because these methods are purely based on the driven trajectory, no precise geometry of the roundabout elements can be derived. Furthermore, these methods can only detect a roundabout after the vehicle passed through it and thus it cannot detect a roundabout the vehicle is approaching.

In the field of image processing the Hough transform is a well-known method for detecting circles in rasterized images. The original idea of the Hough Transform was described and patented by Hough and Paul [HP62] for detection and extraction of straight lines in rasterized images. Duda and Hart [DH72] generalized this method to detect curves and explained how it could be used to detect and extract circles. The main idea behind the circular Hough transform is to evaluate for each of the possible values of the circle parameters (i.e. centre and radius) how many data points from the input support these parameter values. For cartographic purposes, image-processing methods have been developed to recognize roundabouts in aerial images using the Hough transform [MFZ09, BVC00]. The Hough transform is known to be a very expensive algorithm in terms of computation time and memory usage. The Hough transform performance for offline processing of aerial images is acceptable, but this is not the case for a system that should process the data online in real-time and where the circle parameter space is big.

A different approach to detect roundabouts in aerial images was presented by Ravanbakhsh and Fraser [RF09]. Their algorithm requires the complete central island circle to be visible in the image, whereas on-board vehicle sensors can only observe an arc of the circle. This makes their approach unsuitable for our application.

Núñez et al. [NVMdT⁺06] presented a method to segment a laser scan into line and curve segments. However they do not specify what a "curve" segment means. From the provided examples and the description of how the algorithm works, the reader could conclude that a curve segment is a segment for which the curvature is constant (i.e. the segment is a circle). Unfortunately, it is not clear from the description of the method how these curve segments are found.

Xavier et al. [XPC⁺05] also present an approach to detect circles in laser scans. The method splits a laser scan, which consists of a sequence of measured locations, into segments at each

point where the distance between two neighbouring points is above a certain threshold. In those laser scan segments the algorithm then detects line segments. Then for all remaining laser scan segments they verify whether the segment is a circle segment or not. This approach will not work for our application mainly for two reasons. The first reason is that in some cases the laser scan will not make a big jump in distance when going from the ground surface to the curb. The second reason is that the part of the laser scan on the ground is not in general a straight line. This means that the part of the laser scan hitting the roundabout's curb might be in the same final laser scan segment as some parts hitting the ground and therefore the former will not be identified as a circle.

1.3.3 Map-Relative Localization

In order to use a priori digital map data, the ego vehicle has to be localized relative to the data in the digital map. If the map data is geo-referenced then this localization can be achieved by measuring the global position of the ego vehicle with a GNSS like GPS. The disadvantage of a GNSS is that a connection with a sufficient number of satellites is not always available and that the localization accuracy is not acceptable for highly automated vehicles. Combining GPS with correction data, like with Differential GPS, may solve the accuracy issue but it does not solve the availability issue.

To improve localization accuracy without relying on the availability of a GNSS, methods have been developed to localize the vehicle by matching sensor observations with geo-referenced data stored in an a priori digital map. One family of these methods uses an a priori map that contains a high resolution 2D grid where each grid cell stores information about the corresponding location on the ground plane. For example the method by Levison et al. [LMT07] uses an a priori grid map, with a 5 cm cell resolution, of the ground reflectivity in the infrared spectrum. For localization of the ego vehicle, a particle filter correlates infrared LIDAR measurements with this map. Although this group of methods achieves good localization results, their main disadvantage is the enormous storage space required by the high resolution a priori grid maps. A second family of localization methods uses, instead of an a priori grid map, an a priori map with landmarks. The map contains for each landmark its location and a description that helps the system to recognize it. Landmarks can be for example posts or walls of houses as in the approach presented by Weiss et al. [WKD05] or points on the surface of an object with a distinguishable curvature characteristic as in the approach of Madavan and Durrant-Whyte [MDW04]. For localization the ego vehicle will extract landmarks from raw sensor data and match these with the landmarks in the a priori map to estimate the position of the ego vehicle.

In many applications localization is performed because the system has to know its position relative to objects described in the a priori map. For both of the above explained families of sensor-based localization methods, a fraction of the data stored in the a priori map will only be used for localization and is of no further interest for the rest of the system. That means that the a priori map contains two kinds of data: data for localization (e.g. landmarks) and data about the objects the system is actually interested in (e.g. elements of the infrastructure). A drawback of this is that a map containing both kinds of data needs to be maintained. This does not only imply that more information needs to be stored, but also that the map becomes useless (or less trustworthy) if the localization data is outdated even when the object data is still up-to-date.

In the method proposed by Nedevschi et al. [NPM⁺10] the a priori map does not contain additional localization data but only the object data the automated vehicle is interested in. This object data is used to localize the ego vehicle relative to the object. The goal of their system is to perceive a detailed description of a road intersection. They have extended a navigation map with a detailed intersection description that includes the stop lines and for each entry lane the boundaries and the turn-direction of the lane. When the ego vehicle is approaching an intersection, their system detects the boundary of the entry lane and the position of the stop line using a stereo camera. The algorithm also derives the turn-direction of the ego lane from detected arrows in the camera image. The system aligns these observations with the detailed intersection description stored in the a priori map in order to have a complete intersection geometry described relative to the ego vehicle.

For localization on roads with a simple geometry (i.e. low curvatures and no intersections), methods like the one proposed by Chausse et al. [CLC05], localize the vehicle relative to a digital map where the boundaries of lanes are used as landmarks. Based on camera data, the presented method computes the lane width, the road's lateral curvature, the vehicle's distance to the left side of the lane and the vehicle's heading angle relative to the lane. These values are used to update the estimation of the vehicle's location on the map. Because of the relatively great ambiguity of lane boundary observations for low curvature roads, methods like these are known to have a greater position uncertainty along the longitudinal direction of the road compared to the uncertainty in the lateral direction. Furthermore if the system only recognizes the ego lane, then it is possible that the system localizes itself in the wrong lane or at least has multiple ego lane hypotheses. However, Rabe et al. [RNS16] showed that by incorporating radar measurements of other vehicles and stationary objects, these approaches can be used to identifying on which of the lanes of a multi-lane road the vehicle is driving. This makes their approach suitable for driver assistance systems, like lane-level navigation. These method proved to be useful for localization when map data is used to enhance the performance of camera-based lane keeping assistance systems [TPA07]. It seems that this group of methods is useful for driver assistance systems, but that, due to the low accuracy in longitudinal direction, it is not suitable for localization in highly accurate and detailed maps for automated vehicles.

In order to improve the accuracy of localization methods based on lane markings, Mattern et al. [MSW10] and Schreiber et al. [SKF13] proposed approaches that rely on a more detailed map. Instead of having a map only representing the geometry of lane boundary, their approaches use a map where the geometry of each individual road marking is described. That is, when a lane boundary is a dash road marking, then the geometry of each individual dash is described. Furthermore markings at intersections or symbols (e.g. arrows) painted on the road are also represented. Obviously, the mapping efforts and required storage space for the map are higher, but these localization methods proved to result in a high localization accuracy.

1.4 Research Objectives

As can be concluded from the previous sections, much research still has to be done on robust environment perception in order to realize highly automated driving in urban environments. The rather limited capabilities of the current state-of-the-art sensor-based environment perception for urban environments, motivates the first objective of this thesis:

Research objective 1. *Improve the capabilities of sensor-based perception in urban environments using close-to-market sensors.*

However, it is unrealistic to assume that in the near future all elements in the environment, relevant for highly automated driving, will be perceived based on only sensor data. Instead, detailed digital maps will play an important role for the perception of those elements for which no robust sensor-based methods are available yet. Thus, a near-future perception system for highly automated driving must be able to combine sensor-based perception with data from digital maps. This requirement leads to the following objective:

Research objective 2. *Develop methods to combine sensor-based perception with data from an a priori map in order to perceive all relevant environment elements.*

Unfortunately, all automated vehicles demonstrated thus far assume the contents of a digital map to be correct. This may be acceptable for a partially automated system where a human driver has to monitor the system, but not for a highly automated vehicle. Since this problem has been rarely a topic of scientific research, although it is considered to be very significant, it forms the third objective:

Research objective 3. *Formulate and investigate the problem of verifying the correctness of a priori map data before the vehicle acts on it.*

The three research objectives described above are too broad to investigate in their full generality within a single thesis. Therefore these objectives are studied in a specific context, namely the perception of roundabouts. Despite the importance of roundabouts for an early introduction of urban highly automated driving (see section 1.2), the perception of roundabouts has not been studied thus far by the automated driving research community. Hence, the study of roundabouts not only serves as a test bed for research objectives 1–3, it is also interesting in its own right.

Research objective 4. *Develop methods for the perception of roundabouts.*

1.5 Main Contributions and Outline of the Thesis

The following list summarizes the main contributions of this thesis and relates them to the research objectives that were defined in the previous section:

- The introduction of a road representation that allows the representation of multiple road hypotheses in order to cope with ambiguities and uncertainties in the road’s perception. (contributes to research objective 1)
- The first results on on-board sensor-based roundabout perception in the form of a roundabout-central-island detector based on laser scanner measurements. (contributes to research objectives 1 and 4)
- The presentation of a roundabout perception approach that combines partial sensor-based observations of the roundabout with a detailed digital map of the roundabout. This approach can also be considered as a novel way for localization relative to a digital map without requiring additional landmarks. (contributes to research objectives 2 and 4)
- A formulation of the requirements for robust use of a priori digital map data for environment perception. (contributes to research objective 3)
- Identification of difficulties and challenges that may arise when digital map verification is formulated as a probabilistic problem. (contributes to research objective 3)

Chapter 2 provides background knowledge, conventions and definitions that are required for a good understanding of the contents of this thesis.

On urban roads the great variety of road and lane boundaries, bad quality road markings, missing road markings, occlusions and patched roads are some of the reasons that can make the structure of the road ambiguous for a road perception system. This is one of the problems that arises if we want to replace the fully-map-based perception of urban roads by sensor-based perception. This problem is addressed in chapter 3 where a multi-hypothesis road representation is presented. Instead of letting the perception system choose the most likely hypothesis about the road and send this hypothesis to the behaviour planning system, the idea behind the presented representation is to send the set of all plausible road hypotheses. Besides presenting the multi-hypothesis road representation, chapter 3 further motivates the use of it.

Chapter 4 presents a method for detecting central islands of roundabouts using a close-to-market laser scanner. An extensive evaluation is done where not only the detection rate is evaluated, but also the quality of the input data, the error distributions of the detected central islands, and causes of missed detections. Some of the resulting evaluation data, describing the characteristics of the detection method, are used in the following chapters that present methods utilizing the detector.

In chapter 5 a method is presented for the perception of all relevant elements of a roundabout. The output of the central-island detector and a road marking detector are fused with a detailed a priori digital map (which for the purpose of this chapter we assume to be correct) to achieve a complete perception of the roundabout. The method localizes itself relative to this digital map without the need of a precise GNSS or additional landmarks in the map.

Verifying the correctness of a digital map, before the automated vehicle acts on it, is the topic of chapter 6. First, possible requirements on map verification for highly automated driving are discussed. Then the map verification problem is formally specified in a probabilistic way. This specification is followed by a proposal for a generic map verification model, and a description of how it can be applied to verify the correctness of the map of a roundabout's central island. The presented verification model is analysed based on this application.

Final conclusions and recommendations for future research are presented in chapter 7.

Chapter 2

Conventions, Definitions and Background Knowledge

2.1 The Environment Coordinate System

Each sensor measures some quantity in its own coordinate system. Since the perception system must perceive the environment based on different sensors, it must have a common coordinate system in which all information about the environment can be represented. Furthermore, the output of the perception system should be in a coordinate system that is convenient for behaviour planning, which is most likely not the same coordinate system as an arbitrary sensor coordinate system. In this thesis, the perception system therefore represents the environment in a so-called environment coordinate system, both for internal representation and output. The *environment coordinate system* (see also figure 2.1) is defined to be the coordinate system fixed to the vehicle that

- is a right-handed orthogonal coordinate system,
- has the local ground plane on which the vehicle is standing as the xy -plane,
- has the orthogonal projection of the vehicle rear axle's centre as its origin,
- has an x -axis in the same direction as the orthogonal projection of the vehicle's forward-pointing longitudinal axis on the xy -plane,
- and has an upward pointing z -axis.

When in this thesis we talk about coordinates or axes (i.e. x , y or z) without explicitly mentioning their coordinate system, then these are coordinates or axes in the environment coordinate system. When we say that a location or geometry is described *relative to the ego vehicle*, then we mean that the location or geometry is described in the environment coordinate system.

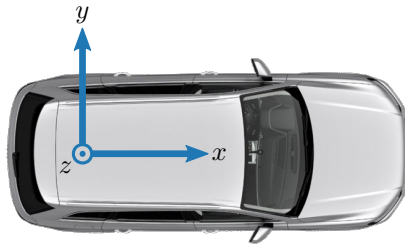


Figure 2.1: The environment coordinate system seen from above the ego vehicle

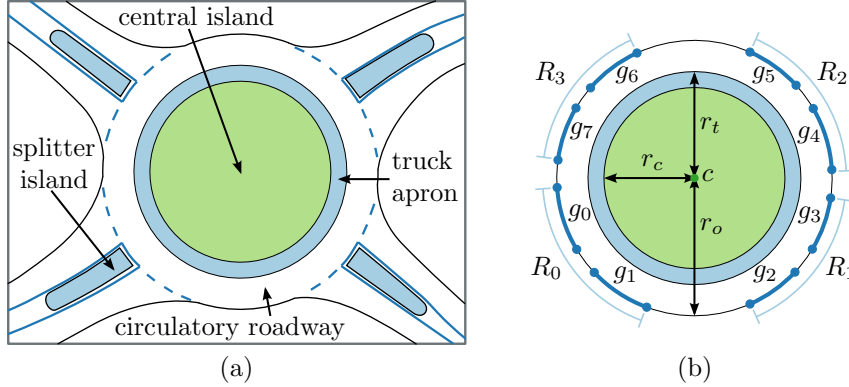


Figure 2.2: (a) A roundabout and its elements; the blue lines are painted road markings. (b) A visualization of the perception model of this roundabout.

2.2 The Roundabout Model

The research objectives of this thesis are mainly applied to roundabout perception. This section describes all relevant elements and terminology concerning roundabouts and presents a model that is used for the output of an environment perception system to describe a perceived roundabout. Figure 2.2(a) is used to clarify the structural elements of a roundabout, while figure 2.2(b) illustrates the proposed roundabout model.

Based on German road construction guidelines [For12] we identify those elements of a roundabout (see figure 2.2(a)) that are relevant for the task of driving highly automated through it. The *central island* is in the centre of the roundabout. Crossing it is prohibited and is made impossible and/or discouraged because it has a road curb. In most cases the island itself is elevated from the road. The *truck apron* is a ring around the central island and may only be crossed by long vehicles like trucks or buses. The truck apron is an optional element. The *circulatory roadway* is the ring-shaped part of the roundabout on which the vehicle should drive. At those parts where roads are connected to the roundabout, the outer border of the roundabout is marked by a road marking (see the dashed road markings in figure 2.2(a)). If a road connected to the roundabout has an entry lane and an exit lane, then those lanes are separated by a *splitter island*.

In this thesis, only single-lane roundabouts are considered, since this is the most common type of roundabout and it is preferred over a multi-lane roundabout in road-design [BSU09, FHW10]. The presented perception methods may work for large multi-lane roundabouts (with an adapted roundabout model), but these roundabouts are out of the scope of this thesis. Furthermore, there are non-circular roundabouts, or small roundabouts where the central island does not have a curb. Also these kinds of roundabouts are out of the scope of this thesis.

Now that we have identified the relevant elements of a roundabout, we describe a model for representing a roundabout (illustrated by figure 2.2(b)). Because the central island, truck apron and circulatory roadway are concentric circles, they have a common centre that is described by position c in the roundabout model. The geometric description of central island, truck apron and circulatory roadway is completed by describing their radii r_c , r_t and r_o respectively. Since the truck apron is an optional element, the model contains an indicator which tells us whether or not a truck apron is present. So-called *gates* model the parts where entry and exit lanes are connected to the roundabout (see g_0, \dots, g_7 in figure 2.2). Each gate is geometrically described by its two end points on the roundabout's outer border (i.e. the circle with radius r_o). The gates are grouped according the roads (see R_0, \dots, R_3 in figure 2.2) to which they belong.

Although road construction guidelines specify requirements on the roads connected to a roundabout, a description of these roads (including their splitter islands) is not part of the presented roundabout model. The reason for this is that, in this thesis, the perception of these roads are not considered to be part of the roundabout perception problem.

2.3 Laser Scanner

2.3.1 Laser Scanner Principles

The detection of roundabout central islands as presented in this thesis is based on measurements by a laser scanner. A laser scanner is a device that is able to make three-dimensional position measurements based on the time-of-flight principle. The device emits short laser pulses that are reflected by an object. In case of diffuse reflection of the laser pulse, the reflected light reaches the device and is detected by a photodetector. Based on the time difference between emission and detection, the distance to the point where the pulse was reflected by the object can be computed. Using a mirror, the laser beam can be deflected in a specific direction. The combination of the known angle in which the beam is deflected and the measured distance, allows us to compute the three-dimensional position of the point where the pulse was reflected by the object. By continuously rotating the mirror (or the emitter-detector unit itself) around a single axis, the device can scan a whole range of angles around the rotation axis. The measurements collected during one rotation are called a *laser scan*. The angle between a laser beam and the rotation axis is fixed. Scanning the environment in different directions simultaneously can be achieved by emitting multiple laser beams with different angles relative to the rotation axis. The emitter-detector unit for each of these beams is referred to as a *layer* of the laser scanner.

Usually a laser scanner is mounted in such way that the axis of rotation is orthogonal (or almost orthogonal) to the ground plane. Therefore, the field of view and angular resolution around the axis of rotation are called the *horizontal field of view* and *horizontal resolution* respectively. Because in such case the layers of laser scanner are “stacked” on top of each other, the field of view and angular resolution in the direction where the directions of the layers vary are called the *vertical field of view* and *vertical resolution* respectively.

There are laser scanners with a large number of scan layers and thereby a high vertical resolution and large vertical field of view. For example, the winning car of the DARPA Urban Challenge used a laser scanner with 64 layers [UAB⁺08]. Because of the price and the dimensions this kind of laser scanner to date, they are far from being a component that can be integrated in mass-production vehicles. Laser scanners with only a few layers are soon to be introduced in production vehicles [Sch15].

2.3.2 Specifications of Used Laser Scanner

The experiments presented in this thesis were done with such a close-to-production laser scanner with 6 scan layers. This device is very similar to the laser scanner evaluated by Zeisler and Maas [ZM15]. The laser scanner is constructed in such way that layers 0–2 scan simultaneously and layers 3–5 scan simultaneously. During one cycle of the laser scanner, first layers 0–2 perform a scan followed by layers 3–5. Layers 1 and 3 have the same direction and layers 2 and 4 have the same direction, as illustrated in figure 2.3. The vertical angular resolution is 0.80° and the vertical field of view is 3.20° . The laser scanner has a horizontal resolution of 0.25° and a horizontal field of view of 145° , meaning that a single layer scan consists of 581 distance measurements. The horizontal and vertical divergence of the laser beams are 0.10° and 0.80° respectively. The distance resolution is 0.04 m and the distance measurement error is less than 0.1 m. The maximum measurable distance is more than 100 m. Laser scans are produced with a frequency of 25 Hz. By this we mean that a scan of layers 0–2 takes 40 ms, a scan of layers 3–5 takes 40 ms and that a complete laser scan cycle, which comprises the scans of layers 0–5, takes 80 ms.

The laser scanner is built into the centre of the vehicle’s front grille, as illustrated in figure 2.3. The sensor’s origin is 3.82 m in front of the vehicle’s rear axis and is approximately 0.3 m above the ground. It is tilted in such way that on a flat road the lowest layer (layer 0) measures a point at the road at approximately 12 m in front of the laser scanner along the x -axis.

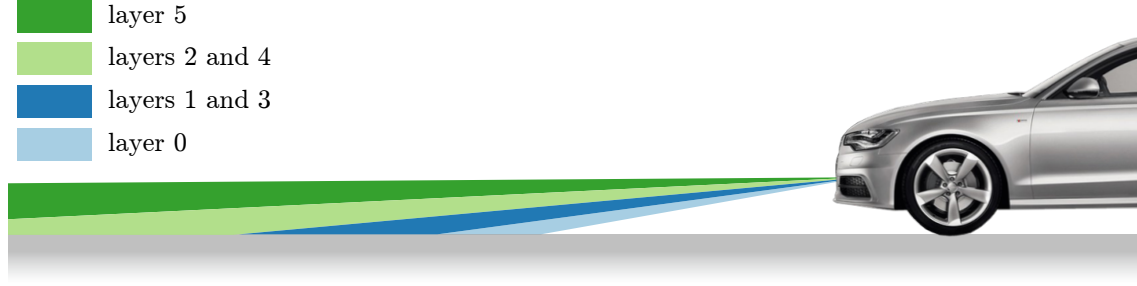


Figure 2.3: An illustration of how the different scan layers of the used laser scanner are positioned relative to each other. For illustrative reasons, the proportions in this figure do not agree with reality.

2.4 Notation of Probabilistic Concepts

The reader of this thesis is assumed to have basic knowledge about probability theory which includes being familiar with the concepts of random variables, (joint) probability distributions, (joint) probability density distributions and statistical dependence. This thesis adopts a notation for probabilities and probability densities that is regularly used within the field of robotics (e.g. in the book by Thrun et al. [TBF05]). Since this notation is different from the notation taught in a classical university course on probability theory, this section explains the notation.

Classical notation. We first consider the classical notation. Let X be a random variable and let x be a realisation of X . Then the probability of the event that X takes value x is written as:

$$\Pr(X = x)$$

Usually this probability is referred to as “the probability that X is x ”. Let Y be another random variable and y be a realisation of Y . Then the probability of the event where X takes value x and Y takes value y is written as:

$$\Pr(X = x, Y = y)$$

and the probability of the event where X takes value x , given that Y took value y , is written as:

$$\Pr(X = x \mid Y = y)$$

Now if X is a continuous random variable then usually $\Pr(X = x) = 0$ for x and speaking of “the probability that X is x ” is not of much use. Usually for a continuous random variable X , symbol F_X is used to denote the cumulative distribution function of X :

$$F_X(x) = \Pr(X \leq x)$$

If Y is another continuous random variable, then symbol $F_{X,Y}$ is to denote the joint cumulative probability function of X and Y :

$$F_{X,Y}(x, y) = \Pr(X \leq x, Y \leq y)$$

The joint cumulative probability density function of X given Y (which can be discrete or continuous) is denoted by symbol $F_{X|Y}$:

$$F_{X|Y}(x, y) = \Pr(X \leq x \mid Y = y)$$

Instead of probability values, we make use of probability density values for continuous random variables. Usually for a continuous random variable X , symbol f_X is used to denote the probability density function of X :

$$f_X(x) = \frac{dF_X(x)}{dx}$$

In case of two continuous random variables (this example extends in a similar way to more continuous random variables) X and Y , symbol $f_{X,Y}$ is used to denote the joint probability density function of X and Y :

$$f_{X,Y}(x, y) = \frac{\partial^2 F_{X,Y}(x, y)}{\partial x \partial y}$$

The conditional probability density function of a continuous random variable X , given a random variable Y (which can be discrete or continuous), is usually denoted using symbol $f_{X|Y}$:

$$f_{X|Y}(x, y) = \frac{dF_{X|Y}(x, y)}{dx}$$

Notation in this thesis. Now for a random variable X we use the following notation to express its probability or probability density for value x :

$$p(X = x)$$

That is, if X is a discrete random variable, then:

$$p(X = x) = \Pr(X = x)$$

If X is a continuous random variable, then:

$$p(X = x) = f_X(x)$$

If two random variables X and Y are both discrete, then $P(X = x, Y = y)$ means:

$$p(X = x, Y = y) = \Pr(X = x, Y = y)$$

Similarly, if two random variables X and Y are both continuous, then $P(X = x, Y = y)$ means:

$$p(X = x, Y = y) = f_{X,Y}(x, y)$$

If X is discrete then:

$$p(X = x | Y = y) = \Pr(X = x | Y = y) \tag{2.1}$$

If X is continuous then:

$$p(X = x | Y = y) = f_{X|Y}(x, y) \tag{2.2}$$

Note that in equations 2.1 and 2.2 Y can be discrete or continuous.

To improve readability, the “ $X =$ ” part in the notation may be omitted in case the symbol used for the realisation of the random variable is the lower-case variant of the symbol used for the random variable. For example:

$$p(x) = p(X = x)$$

The notation that involves \Pr is still used for writing the probability of a random event that does not involve the equality of a random variable to a given value. For example, to express the probability that random variable X takes a value between a and b , we write

$$\Pr(a < X < b),$$

or if we want to express the probability that random variable X takes a value from a set A , we write

$$\Pr(X \in A).$$

Random variables introduced in this thesis are continuous unless stated otherwise.

Chapter 3

A Multi-Hypothesis Road Representation

Whereas the rest of this thesis treats the research objectives within the context of roundabout perception, this chapter focusses on the perception of roadways. Here a *roadway* is defined to be the portion of a road in-between junctions, that consists of at least one lane and is in the lateral direction bounded by a shoulder line (edge of the pavement), a curb line, a guard rail or a barrier. Traffic islands are considered to be part of a roadway. This chapter presents a road representation that, in the presence of perception uncertainties, allows multiple hypotheses about a roadway. The reason for the chapter’s focus on roadways only, is that purely-sensor-based perception, which might require such a multi-hypothesis representation, will most likely be possible for roadways before it is possible for junctions. The development of the representation was part of the PhD research projects of Frank Dierkes, Max Theo Schmidt and the research presented by this thesis. The concept was first presented in a joint publication in 2015 [DRS⁺15].

3.1 Motivation

In order to drive highly automated, a system must be aware of all relevant elements in its environment. Understanding the roadway is an important part of this. By roadway understanding we mean understanding what portions of the roads are considered to be lanes or other relevant parts of the road (e.g. shoulders, splitter islands), and the relations between them. As already mentioned in section 1.3.1, Hillel et al. [HLLR12] state that most of the work on sensor-based road perception thus far has been done for relatively simple roads such as highways. For roads that are more challenging to perceive, most systems used an a priori map. For the simpler roads, there are sensor-based perception methods, but most of these existing methods are restricted to the perception of the ego lane or ego lane plus neighbouring lanes. Therefore, Hillel et al. identify full roadway understanding based on sensor data as a research gap.

Environment perception is and will always be subject to uncertainty. For the perception of simple roadways, this uncertainty may only lie in the geometry of lane boundaries. However, when we want to perceive more complex roadways, another kind of uncertainty will arise: uncertainty about the structure of the roadway. That is, there is uncertainty about how many lanes (and roadway parts of other kinds) there are, and about which of the detected road markings, curbs, etcetera define a lane. This kind of uncertainty may be caused by imperfections of the perception system that results in false positive or false negative detections of lane/road boundaries. However, even with a perfect perception of these boundaries, uncertainty about the roadway’s structure can persist. Uncertainty can be the result of ambiguity caused by the roadway itself, for example in construction zones or by poorly removed old road markings. Bad visibility (e.g. caused by fog, rain or sun glare) or occlusions can also cause uncertainty.

Behaviour planning methods developed thus far only consider uncertainty about a lane’s geometry or about the future behaviour of other road users (e.g. [UM13, BGD14]), if uncertainties are considered at all. However, each of these methods is based on an unambiguous understanding of the road. Such methods assume that solely the perception system copes with uncertainties about the roadway structure as discussed above. Even though increasing robustness of the perception capabilities of automated vehicles may significantly reduce the uncertainty about the roadway, some uncertainty will remain. In general, requiring an unambiguous representation of the roadway for behaviour planning, forces the perception system to make decisions that may affect the behaviour of the vehicle in a crucial way.

If there are multiple plausible hypotheses about the roadway, then it may be beneficial or even necessary for the behaviour planning to know all of them instead of just one. As an example, consider the start of an off-ramp, which is easily falsely recognized as a widening lane (for instance because at the start, the perception system has not accumulated enough evidence for lane marking that separates both lanes). Instead of picking one hypothesis —the widening lane or the lane bounded by a marking with little evidence— the perception system can present both hypotheses to the behaviour planning system. The behaviour planning system might then decide to follow the lane with the constant width, since it is also a portion of the widening lane.

The considerations above lead us to present a multi-hypothesis road representation. The goal of our representation is to eliminate the requirement on the perception system to present a single roadway hypothesis to the behaviour planning system. Consequently, with this representation the behaviour planning system can consider all plausible (possibly conflicting) hypotheses. The representation can represent multiple hypotheses about the roadway and is designed in such a way that it is easy to infer where and how the hypotheses about the roadway differ and where they are the same.

3.2 A Single-Hypothesis Roadway Model

Before presenting the multi-hypothesis road representation, this section discusses how we would model a single hypothesis roadway in the absence of uncertainty. This is done for two reasons. First, it clarifies which modelling concepts are introduced in order to model a roadway in general (i.e. without ambiguity) and which concepts are introduced in order to model uncertainties. Second, it allows us to introduce some terminology about the roadway that applies to both a single-hypothesis representation and a multi-hypothesis representation.

A roadway consists of at least one lane. A lane is bounded in lateral direction by a left and right boundary. However, a roadway does not only consist of lanes as can be seen in the lateral cross section of a road in figure 3.1; there can be shoulders, traffic islands and bicycle lanes for example. In a lateral cross section of the roadway, these parts too are bounded by a left and right boundary. We generalize these kinds of parts (including lanes) by modelling them as *strips*. A strip is a part of the roadway that is bounded in lateral direction by two boundaries. Boundaries can appear in different forms: road markings (of different kinds), Botts’ dots, shoulder lines, curbs, etcetera.

For an automated vehicle it is important to know which strips are next to each other, and therefore a roadway model should describe these neighbour relations. These relationships are not constant over the complete length of the roadway (see for example figure 3.2), which is the reason for introducing the concept of *roadway segments*. In the longitudinal direction the roadway is partitioned in roadway segments. Roadway segments begin and end at those locations where the neighbour relations between strips change, as illustrated in figure 3.2. That is, within a roadway segment the strip neighbour relations are constant. A roadway is modelled as a sequence of roadway segments.

A *strip segment* and *boundary segment* are the portion within a roadway segment of a strip and boundary, respectively. A roadway segment is modelled as a sequence of strip segments ordered from one side of the road to the other side. The order of this sequence describes neighbour relations (in lateral direction). A strip segment itself is described by its two boundary segments.

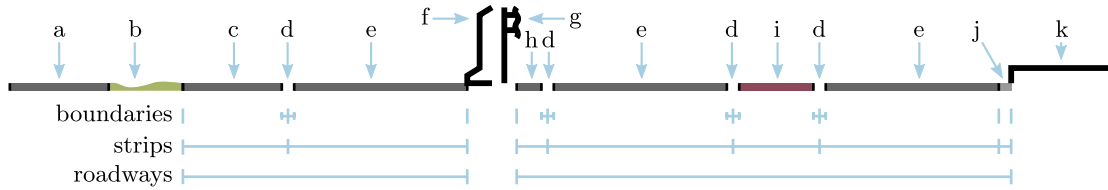


Figure 3.1: The structure of a roadway in lateral direction illustrated by a cross-section of the roadway. The roadway is composed of strips. A strip is defined by its lateral boundaries and its type. There are no gaps between strips and no overlaps of strips. A road may have more than one roadway. Here, an example of a rather rare cross-section of a road is pictured. It consists of (a) a side path, (b) a green strip, (c) a shoulder, (d) painted lane markings, (e) travel lanes, (f) a barrier, (g) a guard rail, (h) a separating strip, (i) a bicycle lane, (j) a gutter and (k) a footway.

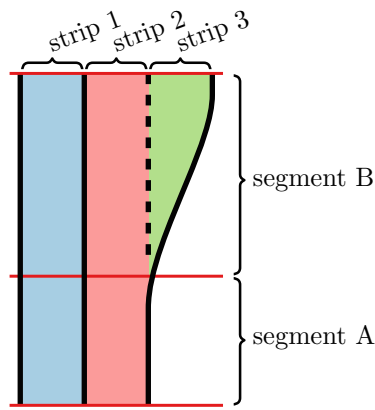


Figure 3.2: In this example roadway is divided into two roadway segments at the point where strip 3 starts. Strip segments 1A and 2A are connected longitudinally to strip segments 1B and 2B, respectively. Strip segment 3B is only connected laterally to strip segment 2B.

Furthermore, the connections in longitudinal direction between consecutive strip segments are described.

3.3 Multi-Hypothesis Representation

To explain the multi-hypothesis road representation in this section we make use of the example in figure 3.3. As illustrated on the left side of figure 3.3, in this example there exists a road with road markings that have been detected only partially by the perception system as well as false road marking detections. Since the perception system cannot distinguish between true and false detections of road markings, it generates multiple hypotheses about the roadway. Here we define a *roadway hypothesis* as a hypothesis about the roadway over the complete represented longitudinal range of the roadway. Such roadway hypotheses can be generated based on, for example, the quality of the road marking detections, knowledge about road models based on road construction guidelines, or a priori map knowledge. Since the criteria and methods for hypothesis generation are not within the scope of this chapter, some possible hypotheses are presented without trying to present a complete set of possible hypotheses. The main point of the example is to show how multiple roadway hypotheses are represented in our representation.

The goal of the multi-hypothesis representation is to represent the set of different roadway hypotheses, such that it is easily interpretable and is efficient in terms of memory usage. The presented representation is easily interpretable in the sense that it is easy for the user to determine where and how the roadway hypotheses differ. There may be some longitudinal ranges where

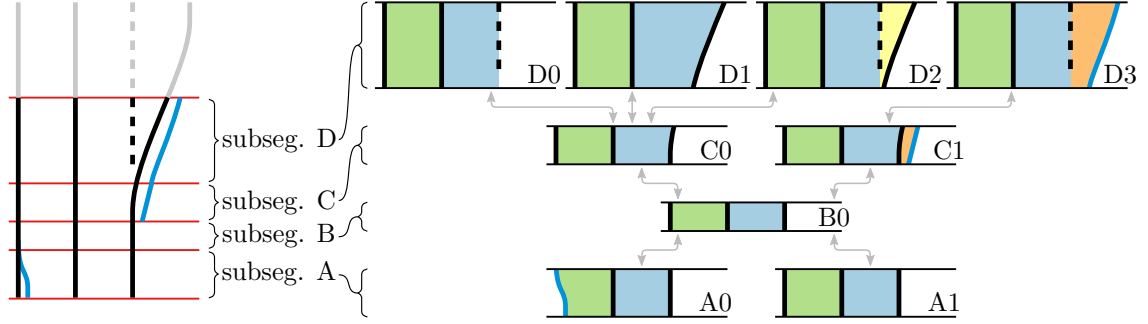


Figure 3.3: Left side: Road markings as they appear in the real world are visualized by the black/grey lines. The black part of these markings indicate the part that is observed by the perception system and are considered as road marking candidates. The blue lines visualize further candidates for road markings that were detected by the perception system. The red lines indicate the borders of the roadway subsegments in the constructed road representation instance. Right side: For each of the roadway subsegments the roadway subsegment hypotheses are visualized. The arrows indicate the longitudinal connections between these road subsegment hypotheses. If two strip subsegment hypotheses of neighbouring subsegments have the same colour, then that means that there exists a strip subsegment hypothesis connection between them.

multiple roadway hypotheses are the same. The presented representation is memory efficient in the sense that it describes a hypothesis about the roadway within each such longitudinal range only once, instead of describing it multiple times for each roadway hypothesis separately. This kind of memory efficiency and interpretability is achieved by partitioning the roadway into *roadway subsegments*. A roadway subsegment is defined as a range in longitudinal direction of the road with the following properties:

1. There are no two roadway hypotheses such that within the subsegment there is one position in the longitudinal direction where the lateral structure of the roadway hypotheses differ and another position where they are the same.
2. For each roadway hypothesis, the neighbour relations between strips are the same for each longitudinal position within the subsegment.
3. The range is maximal. That is, there exists no other range in longitudinal direction that includes the subsegment and satisfies properties 1 and 2.

Property 1 ensures the memory efficiency and interpretability requirement described above. Property 2 ensures a convenient interpretation of strip neighbour relations just as roadway segments do in the single-hypothesis case. Property 3 minimizes the number of roadway subsegments. As a result of this definition of roadway subsegments, the roadway is split into subsegments at those places where roadway hypotheses start or end to differ, or where there is at least one roadway hypothesis for which the neighbour relations of its strips change.

The data structure of our multi-hypothesis representation is illustrated by an UML class diagram in figure 3.4. Here a roadway is represented by the `Roadway` class and its partitioning into roadway subsegments is represented by a list of `RoadwaySubseg` objects. Each roadway subsegment has a set of *roadway subsegment hypotheses* (represented by class `RoadwaySubsegHyp`) that each describe a hypothesis about the roadway *within* that roadway subsegment.

On the right side of figure 3.3 we see for each roadway subsegment in the example, the possible roadway subsegment hypotheses. In roadway subsegment A there is an uncertainty about the left border of the left lane because there are two possible candidates for this boundary. Therefore there are two roadway subsegment hypotheses for A: hypothesis A0 and hypothesis A1. Within roadway subsegment B, all roadway hypotheses are the same and thus there is only one hypothesis B0. Because of this change in uncertainty about the roadway when going from subsegment A to

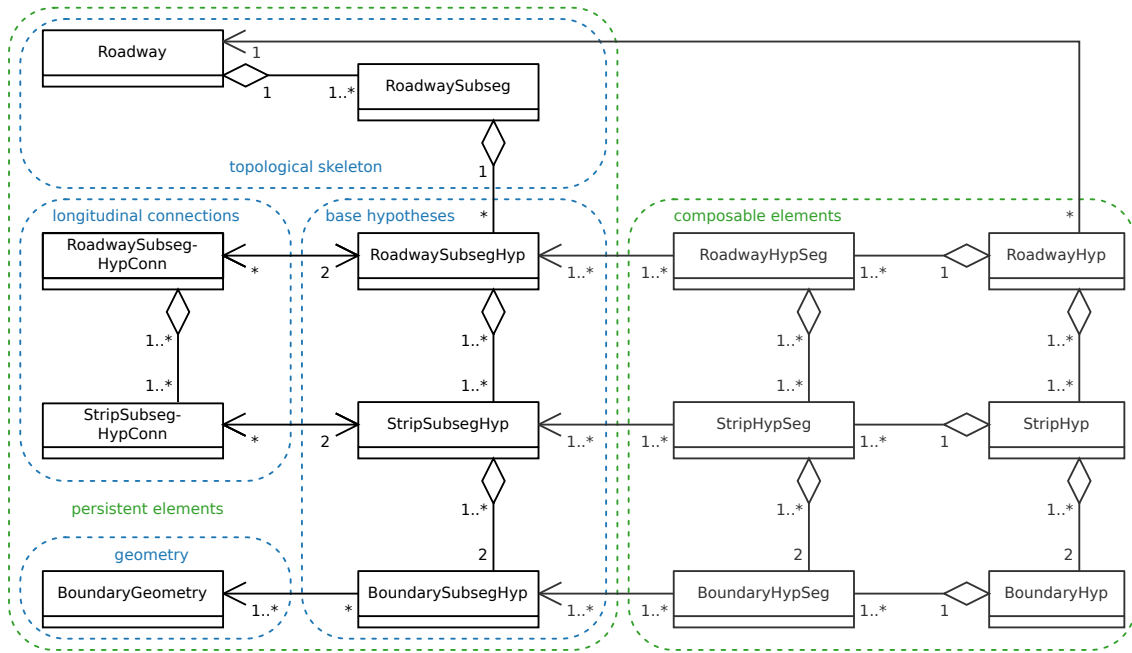


Figure 3.4: UML class diagram of the multi-hypothesis representation. The persistent elements are mandatory. The roadway (with its location line) and its subsegments are the topological skeleton for the composition of roadway hypotheses. For each subsegment, roadway subsegment hypotheses can be composed of contiguous strip subsegment hypotheses. In longitudinal direction, compatible roadway subsegment hypotheses can be connected. A roadway subsegment hypotheses connection defines which strip subsegment hypotheses of the corresponding roadway subsegment hypotheses are connected. A strip subsegment hypothesis is defined by its boundary subsegment hypotheses. A boundary subsegment hypothesis refers to a subsegment of the geometric description of a detected or virtual boundary. The composable elements can be used to describe a single roadway hypothesis or a continuous lane hypothesis, for example.

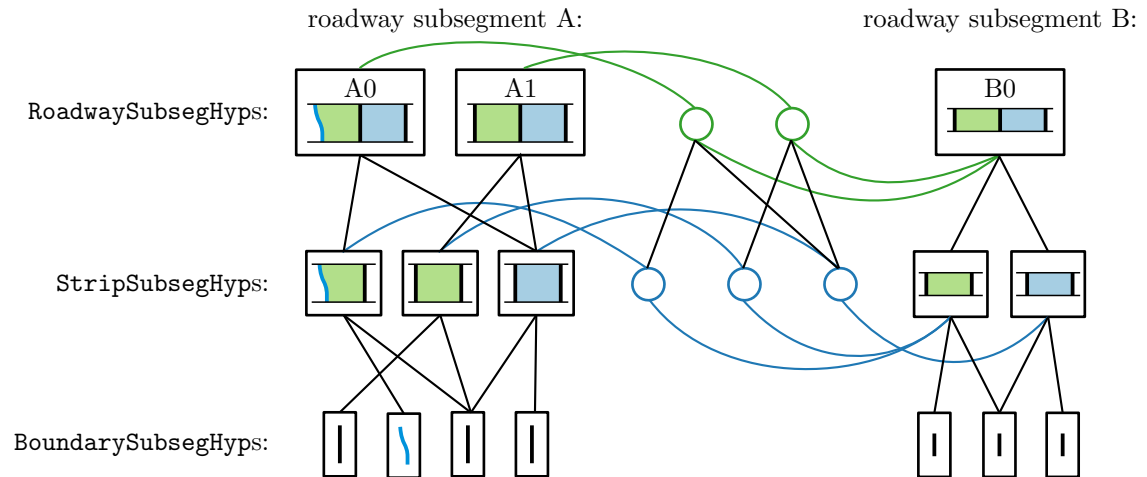


Figure 3.5: A visual representation of the associations between objects of the classes RoadwaySubsegHyp, StripSubsegHyp, BoundarySubsegHyp, RoadwaySubsegHypConn (the green circles) and StripSubsegHypConn (the blue circles), corresponding to the roadway subsegments A and B of the example depicted by figure 3.3.

B, property 1 enforces us to split up the roadway into subsegments A and B. Since the perception system does not know whether the falsely detected marking on the right is correct or not, there are two roadway subsegment hypotheses (C0 and C1) for roadway subsegment C. Apart from property 1, a split up between subsegments B and C is also enforced by property 2 because there are roadway hypotheses (namely all roadway hypotheses that contain roadway subsegment hypothesis C1) that get a new lane in subsegment C. When we take a look at the hypothesis for roadway subsegment D, we see that there is doubt about the borders of the second and third lane and the existence of a third lane.

The part of a hypothesis about a strip within a certain roadway subsegment, is called a *strip subsegment hypothesis* (represented by the class `StripSubsegHyp`). A `RoadwaySubsegHyp` object contains an ordered list of references to those `StripSubsegHyp` objects that are part of the roadway subsegment hypothesis. The order of this list encodes the neighbour relations between the strips: two strips are neighbours if and only if they are neighbours in the list. The list is a list of references since one `StripSubsegHyp` can belong to multiple hypotheses of the same roadway subsegment. This makes it possible for the user to determine whether different roadway subsegment hypotheses of the same roadway subsegment have strips in common. For example, for roadway subsegment A there are three different strip subsegment hypotheses (see also figure 3.5); roadway subsegment hypotheses A0 and A1 share the strip subsegment hypothesis for the blue coloured strip, but have different hypotheses for the green coloured strip.

The class `RoadwaySubsegHypConn` models possible connections in longitudinal direction between two roadway subsegment hypotheses. Thus there exists a connection between two roadway subsegment hypotheses if and only if they belong to consecutive subsegments and there exists a roadway hypothesis that contains both roadway subsegment hypotheses. For example, there is a connection between roadway subsegment hypotheses C0 and D2 because there exists a hypothesis about the roadway that contains both of them. On the other hand there is no connection between C0 and D3 since, for some reason, these hypotheses are incompatible with each other (this depends on the method and criteria used to construct hypotheses which is outside the scope of this work).

Similar to connections between roadway subsegment hypotheses, connections in longitudinal direction between strip subsegment hypotheses are modelled by the class `StripSubsegHypConn`. There exists a connection between two strip subsegment hypotheses if and only if they are in consecutive subsegments and there exists a roadway hypothesis in which both strip subsegment hypotheses are part of the same strip. In order to find out how the strip subsegment hypotheses of two given roadway subsegment hypotheses are connected, a list with references to the corresponding `StripSubsegHypConn` objects is stored in the `RoadwaySubsegHypConn`. These references are visualized in figure 3.5 by the lines between the green and blue circles.

A strip subsegment hypothesis is defined by its two lateral boundaries and its strip type. These boundaries are called *boundary subsegment hypotheses* (modelled by the class `BoundarySubsegHyp`) and describe the hypothesis of a boundary in its corresponding roadway subsegment. A `StripSubsegHyp` refers to the two `BoundarySubsegHyp` objects that correspond to its two boundaries. There are two cases where a `BoundarySubsegHyp` belongs to multiple `StripSubsegHyp` objects. First, the case of two strip subsegment hypotheses of the same roadway subsegment hypothesis that are neighbours and therefore have a common boundary. Second, the case of two strip subsegment hypotheses of different roadway subsegment hypotheses that share a common boundary. The latter case applies in our example, for instance, to the right boundary of the green coloured lane in subsegment A: the two different strip subsegment hypotheses of this lane have the right boundary in common as can also be seen in figure 3.5. A `BoundarySubsegHyp` contains a geometric description of the boundary, the type of the boundary (lane marking, road curb, shoulder, etc.) and other attributes of the boundary that may be relevant for the application.

The elements on the right side of figure 3.4 can be used to compose elements that describe a single hypothesis. An object of type `RoadwayHyp` describes a single roadway hypothesis, for example.

3.4 Recommendations for Future Research

The multi-hypothesis road representation presented in this chapter has been implemented and used in an automated vehicle as an interface between the perception system and behaviour planning system. Unfortunately, in this application the added value of multiple hypotheses could not be evaluated since the behaviour planning system only uses the most likely roadway hypothesis. The reason for this is that no behaviour planning methods that consider multiple hypotheses have been developed yet. Behaviour planning under consideration of multiple roadway hypotheses is a part of the current PhD research by Frank Dierkes. The first publications about these new methods, which use the multi-hypothesis road representation, are expected to be published in the near future.

The representation, described in this chapter, can deal with uncertainty about the structure of a roadway by representing multiple hypotheses. What is not yet possible is to express the degree of uncertainty of each hypothesis. For example, assigning probabilities to hypotheses would allow us to compare which of hypotheses are more likely to be true. Developing a probabilistic model for the multi-hypothesis road representation is a non-trivial task and the possibilities for quantifying the degree of uncertainty are not limited to probability values (for example, the Dempster-Shafer theory [YL08] defines so-called belief functions and plausibility functions to express the degree of uncertainty). Therefore the development of a mathematical model to express the degree of uncertainty, is a topic of future research. The development of this model should consider how the perception system generates hypotheses and the way the behaviour planning system handles different hypotheses.

Currently, only roadways can be represented. In order to model a complete road network, it must also be possible to model intersections. Because of the great variety in the possible structures for an intersection, specifying a generic model for even a single hypothesis of an intersection, is a non-trivial task; specifying a multi-hypothesis intersection model is even more difficult. Also the connection between a roadway and an intersection may be ambiguous: e.g. is at a certain location the roadway connected to an intersection, or does the roadway simply continue? Since sensor-based perception of intersections has hardly received any attention within the research community, it also unclear yet what ambiguities might occur at intersections. Nevertheless, as soon as the multi-hypothesis representation proves itself useful (or maybe even necessary) at roadways, future research on extending the representation with the representation of intersections will be an important next step. Probably the most fruitful next step is to extend it first with a representation of roundabouts instead of intersections in general, for the same reasons that motivated the choice of roundabouts as a topic of this thesis (see section 1.2).

Chapter 4

Roundabout Central Island Detection

4.1 Introduction

As mentioned in chapter 1, sensor-based perception of roundabouts for highly automated driving has not been a topic of research so far. This chapter contributes to this topic, by presenting a method for detecting roundabout central islands in single-layer laser scans. A main advantage of the method is that it does not rely on very costly and bulky high resolution laser scanners as used on some research vehicles. Instead, the presented method uses a laser scanner that only has a few scanning layers, which is a close-to-production automotive component (see section 2.3.2).

The principle by which one can measure (a part of) the central island with a single layer of a laser scanner, is illustrated by figures 4.1 and 4.2. This principle is based on the fact that a central island is bounded by a road curb. The laser scanner is built into the front bumper of the vehicle and all of its layers are slightly tilted downwards. While approaching the roundabout, at some point the vehicle is close enough such that within a single scan the laser is reflected by a part of this curb. That is, in such a scan a part of the central island's border is visible. The presented algorithm detects this part of the scan and estimates the circle parameters of the central island.

The application of the presented method is not limited to roundabout perception. In general it can be used to detect circular (parts of) objects in laser scans, or to detect circular arcs in point sequences. The method is robust against measurement noise and imperfections of the circular shape of the real-world object.

This chapter is based on our paper that was published in the proceedings of the *17th International IEEE Conference on Intelligent Transportation Systems (ITSC)* [RB14]. The major difference with the paper is a more extensive evaluation of the method.

Section 4.2 gives a formal specification of the detection problem. An overview of the algorithm is presented in section 4.3. One of the main contributions of this algorithm is the way it partitions a laser scan into segments by two partitioning steps: the semi-convex partitioning step and the curvature-based partitioning step, which are discussed in detail in sections 4.4 and 4.5 respectively. Section 4.6 discusses how the algorithm was evaluated as a central-island detector and presents experimental results. Based on the experimental results, conclusions are drawn in section 4.7.

4.2 Problem Statement

The goal is to detect the border of the central island of a roundabout through the detection of circles in single-layer laser scans.

For many laser scanners with only a few layers, the differences in elevation between the curb of the central island's border and the road next to it cannot be observed, because of the uncertainty

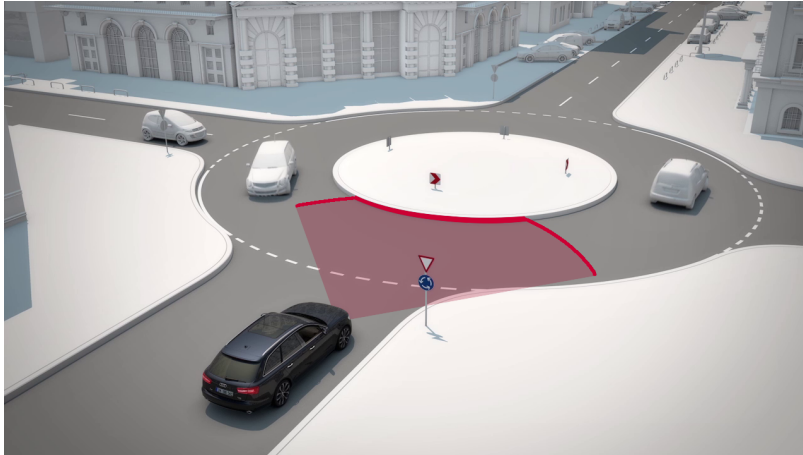


Figure 4.1: The red area is scanned by one of the laser scanner's layers. The highlighted red line visualises the locations where the laser is reflected by an object (which is the road surface or the central island's curb in this figure).

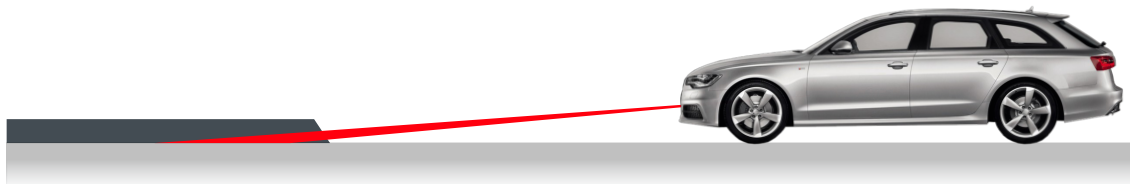


Figure 4.2: A side view where the scanning surface of one of the laser scanner's layers intersects with the central island's curb.

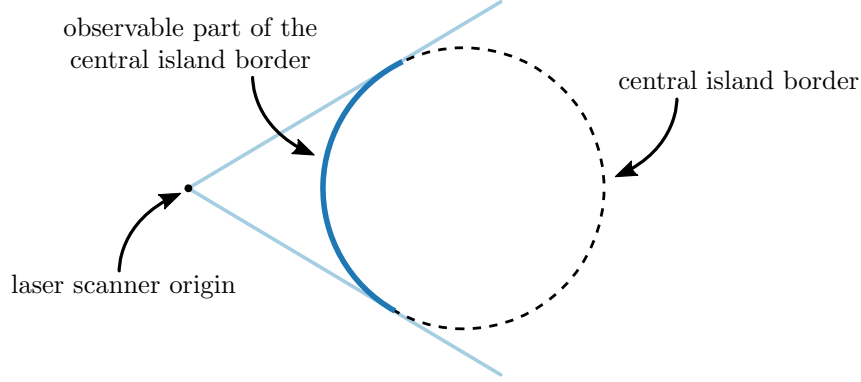


Figure 4.3: An illustration of the part of the central island's border that can be observed by the laser scanner: the border can only be observed from the outside.

for the elevation of a measured point caused by the divergence of the laser beam. We therefore restrict ourselves to the detection of circles in the projection of the 3d laser scan onto the xy plane.

The input to the algorithm is a sequence of point measurements generated by a single scan of a laser scanner layer. We denote this input sequence by $P = [p_1, \dots, p_n]$. We assume that the laser scanner scans in clockwise direction (as seen from a top-view). Consequently p_i is the i -th measurement in clockwise order around the laser scanner's position. (Note that, because of symmetry, the presented algorithm can be easily modified for counter-clockwise laser scans.) We furthermore assume that the tangential uncertainty of a measurement is small enough to ensure that clockwise order of the points p_1, \dots, p_n is the same as the clockwise order of the corresponding real-world points that were measured. For simplicity, we make no distinction between a point measurement's uncertainty in radial direction and tangential direction. Instead, we assume the absolute error of a point to be at most ε_m (which is assumed to be much smaller than the radii of the central islands we want to detect). As mentioned, due to the measurement's precision the elevation of a point will not help us in distinguishing points on a curb from points on the road. Hence, we will ignore the elevations from now on, and we consider the measurements p_i to be points in the two-dimensional xy -plane. We assume that the central island border is only observable from the outside as illustrated by figure 4.3. That is, a point p on the border (in the xy -plane) can only be measured by the laser scanner that is at position o (in xy -plane) if in the line segment \overline{op} does not intersect the central island border except at p . For every point sequence discussed in this chapter we silently assume that it is the output of a laser scanner and thereby has the properties described above.

The problem we wish to solve is to detect the border of the roundabout's central island using the point sequence P . To make the problem statement more precise, we first present a model of this border. Intuitively, the border of the central island is a circle in the two-dimensional plane in which we assume the measurements lie. However, in practice the border will not be an exact mathematical circle. To model this, we need the concepts of *star-shaped curves* and *ε -circle-approximations*, we define next.

Recall that a Jordan curve, is a closed curve in the plane without self-intersections.

Definition 1 (star-shaped). A Jordan curve Γ is called *star-shaped with respect to a point q* lying in the interior of Γ if and only if the following holds: for each point $p \in \Gamma$, the line segment \overline{pq} does not intersect Γ except at point p . The curve Γ is *star-shaped* if there exists a point q such that Γ is star-shaped with respect to q .

Definition 2 (ε -circle-approximation). A Jordan curve Γ is called an *ε -circle-approximation* of a circle with centre c if and only if Γ is star-shaped with respect to c and for every point on Γ the closest distance to the circle is at most ε .

We assume that the real border of the central island is a Jordan curve B_r . Furthermore we

assume that there exists a circle C_r such that B_r is an ε_c -circle-approximation of C_r for a certain value of ε_c . That is, ε_c is used as a parameter to describe the maximum deviation of a central island's border from a perfect circle.

Intuitively, our goal is now to compute a circle C from the measurement sequence P that is a good approximation of the circle C_r . This is complicated by the fact that the measurements p_i do not lie on C_r and, in fact, they do not even lie on B_r : First of all, the sequence P not only contains measurements of points on the central island's curb but also of points on other parts of the environment, like the road (as shown in figure 4.1), the interior of the central island or other vehicles. Second, the measurements are not exact, so even the measurements of points that should lie on B_r do not lie exactly on B_r .

Definition 3 (subsequence). Let $Q = [q_1, \dots, q_{|Q|}]$ and $Q' = [q'_1, \dots, q'_{|Q'|}]$ be two sequences with $|Q'| \leq |Q|$, where $|Q|$ denotes the number of elements of a sequence $|Q|$. We then call Q' a subsequence of Q if and only if there exists some $j \in \mathbb{N}$, with $j \leq |Q| - |Q'|$, such that for each q'_i in Q' we have $q'_i = q_{j+i}$.

The algorithm will detect (possibly more than one) potential circles in the input. More precisely, it will report a number of subsequences of P such that each reported subsequence contains points that are “almost co-circular”, as defined below.

We define a *least-square circle* and a *good circle fit* as follows:

Definition 4 (least-square circle). A *least-square circle* of a point (sub)sequence $Q = [q_1, \dots, q_{|Q|}]$ is a circle C for which the sum $\sum_{i=1}^{|Q|} d(q_i, C)^2$ is minimal, where $d(q_i, C)$ denotes the orthogonal distance from point q_i to circle C .

Definition 5 (good fit). Given a parameter $f_{\min} \in [0, 1]$, a circle C is said to be a *good fit* for a point (sub)sequence if at least $n \cdot f_{\min}$ points in this sequence have a distance to circle C not more than $\varepsilon_c + \varepsilon_m$.

The goal of our algorithm is now to find subsequences P' of P , together with a least-square circle $C_{\text{lsq}}(P')$ of each such P' , with the following properties:

1. P' should have at least n_{\min} points, where n_{\min} is a parameter.
2. The arc of $C_{\text{lsq}}(P')$ covered by P' should be of length at least $2\pi r l_{\min}$, where r is the radius of circle $C_{\text{lsq}}(P')$ and l_{\min} is a parameter specifying the minimal fraction of the circle that needs to be covered.
3. $C_{\text{lsq}}(P')$ is a good fit for P' .
4. P' is maximal in the sense that for no other subsequence of P that contains P' , the constraints above are satisfied for any circle C .

Note that above we specified the ideal behaviour of the algorithm. The algorithm presented in this work approximates this behaviour in the sense that it might report only a subsequence of such a maximal subsequence P' . Thus our algorithm will report a collection of subsequences P' of the input sequence P and corresponding least-square circles such that each subsequence P' satisfies properties 1–3 above; the subsequences are not guaranteed to be maximal (and the algorithm may even miss subsequences).

4.3 Algorithm Overview

Before discussing the algorithm we provide the following definition:

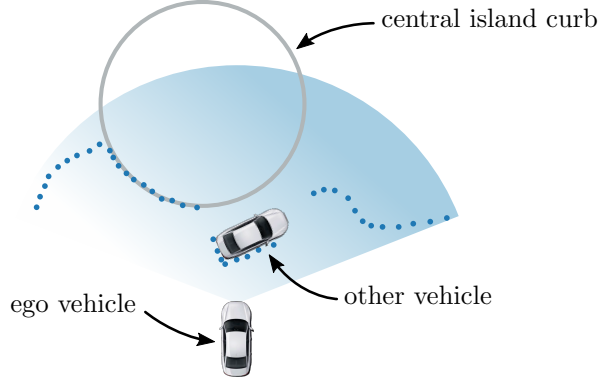


Figure 4.4: A schematic top-view drawing of a scenario where the laser scanner scans the curb of a central island, another vehicle partially occluding the central island and the road. The blue points are points measured by the laser scanner.

Definition 6 (partitioning of a sequence). Let Q be a point sequence. We say that $\mathcal{S} = [S_1, S_2, \dots, S_m]$ is a partitioning of a point sequence Q if each S_i is a subsequence of Q ; for each i , $1 \leq i < m$, the last element of S_i is the first element of S_{i+1} ; the first element of S_1 is the first element of Q ; and the last element of S_m is the last element of Q . The elements of \mathcal{S} are also called the segments of partitioning \mathcal{S} .

The algorithm processes its input point sequence by the following steps (see also figure 4.5):

1. **Neighbour-Based Splitting.** Split the laser scanner's point sequence into new sequences based on neighbour distance: two neighbouring points p_i, p_{i+1} of the input sequence are in the same new sequence if and only if the distance between two those points is smaller than a specified maximum d_{nb} , where d_{nb} is a given parameter. We perform this splitting because a relatively large distance between neighbouring points only occurs if the laser ray is almost parallel to the object's tangent at the measured point, or at a discontinuity of the scanned surface. For subsequences corresponding to a central island, the former can only occur at the beginning or end of the sequence. A discontinuity can be caused if one object partially occludes another object that was scanned (e.g. see figure 4.4 where the central island is partially occluded by another vehicle).
2. **Semi-Convex Partitioning.** Split each sequence at the points where it is turning towards the vehicle (i.e. the points where it is concave). The sensor will only see the circular border from the outside and therefore the observable part of the central island's border is convex from the sensor's point of view. Intuitively we want to split each sequence at those points where it is concave, so that after this step we have a partitioning with only convex segments. To make this step robust against noise and the fact that the real central island is not a perfect circle, a small (parametrizable) amount of concavity is allowed. This step is explained in detail in section 4.4.
3. **Curvature-Based Partitioning.** Since a circle has a constant curvature, we split each semi-convex segment, generated in the previous step, further into segments at those points where the curvature of the corresponding object cannot be constant. This step is discussed in detail in section 4.5.
4. **Circle Fitting & Constraint Checking.** After the curvature-based partitioning, we are left with a point sequence partitioning of which each segment is semi-convex and the curvature change is small. After discarding all segments that contain less than n_{min} points, we compute the least-square circle for the remaining segments using the Newton-based implementation of the Taubin fit which is described by Chernov [Che10]. For each of the computed

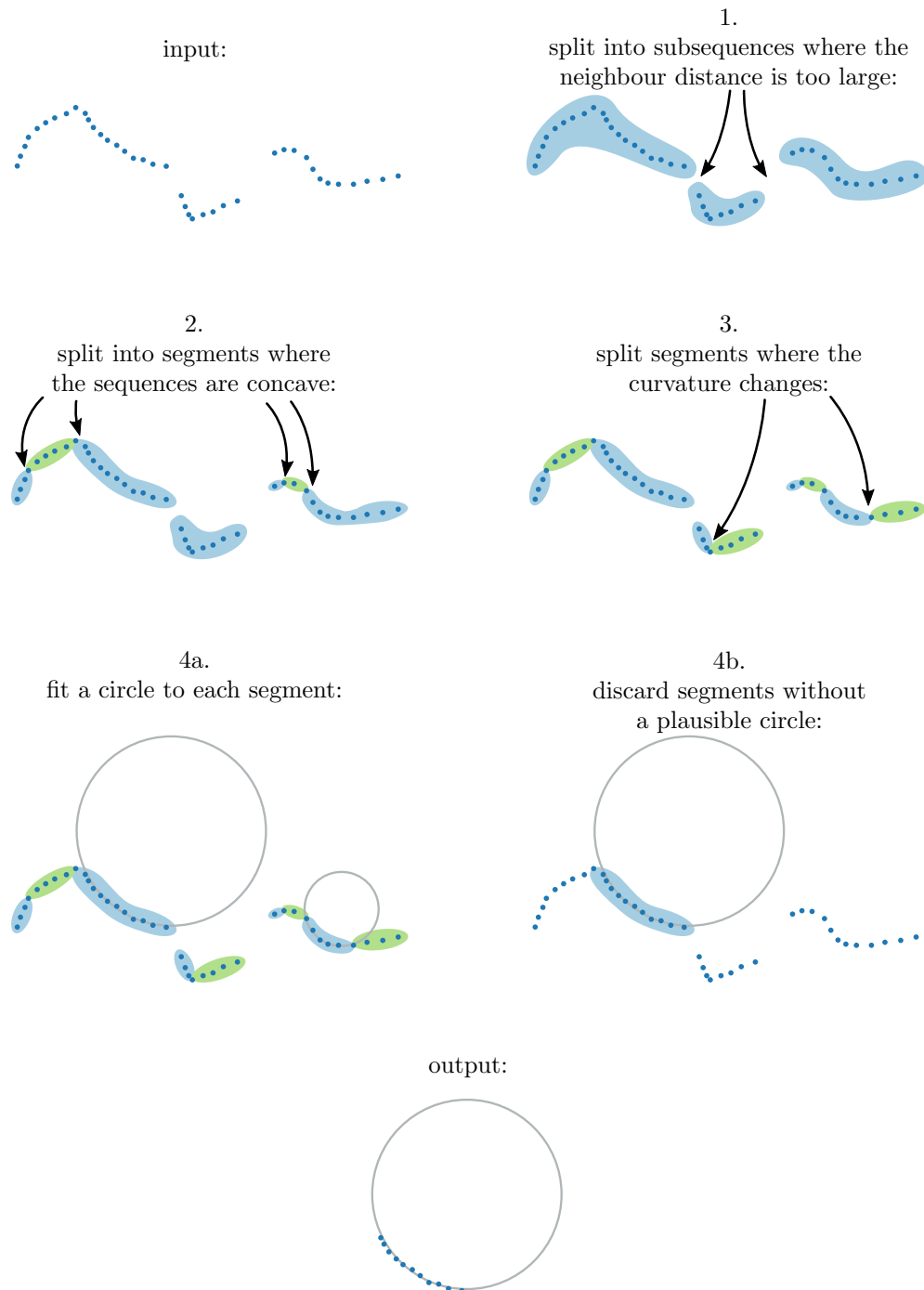


Figure 4.5: An overview of the algorithm.

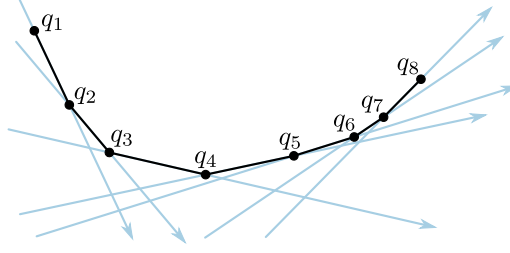


Figure 4.6: A convex point sequence $[q_1, \dots, q_8]$. Note how for each of the blue directed lines passing through two neighbouring points, all points of the sequence are at the left or on the directed line.

circles, constraints 2 and 3 (see section 4.2) are checked. If these constraints are satisfied then the segment and its circle is reported as a detected circle.

For the application of roundabout detection there is an additional step which discards those detected circles for which it is not plausible that they are a roundabout. This step uses an a priori digital street map (in the experiments a map was used with the quality of regular consumer navigation systems to date) and GPS localization, which provide us a rough estimate for the centre and radius of the central island ($\pm 15\text{m}$ for the centre and $\pm 5\text{m}$ for the radius). All circles that are outside the plausible ranges for the centre and radius are discarded.

4.4 Semi-Convex Partitioning

The first step, as described in section 4.3, of the central detection algorithm splits the input point sequence P into subsequences. The semi-convex partitioning step, which is discussed in this section, is applied to each of these subsequences. So the semi-convex partitioning step operates on a subsequence of P , and the complete laser scan point sequence P will not be of any relevance anymore. We therefore reuse $P = [q_1, \dots, q_n]$ to denote the point sequence to which the semi-convex partitioning algorithm is applied.

4.4.1 Convexity and Semi-Convexity

For closed curves there exists a definition of what a *convex* closed curve means. For the explanation of the following algorithm we define the meaning of the word convex in the context of laser scan point sequences. We also define the lower convex hull of a laser scan point sequence. Recall that a laser scan point sequence is ordered in clockwise direction around the position of the laser scanner.

Definition 7 (convex point sequence). A point sequence $Q = [q_1, q_2, \dots, q_{|Q|}]$ (see the example in figure 4.6) is a convex point sequence if and only if we do not make right turns when travelling over all points of Q in the order of the sequence. That is, for each triple of subsequent points q_i, q_{i+1}, q_{i+2} , point q_{i+2} is to the left or on the line through q_i and q_{i+1} directed from q_i to q_{i+1} .

Definition 8 (lower convex hull). For a point sequence $Q = [q_1, q_2, \dots, q_{|Q|}]$ the lower convex hull of Q , denoted as $\text{LCH}(Q)$, is a convex point sequence $[h_1, h_2, \dots, h_m]$ with the following properties (see figure 4.7): (i) $h_1 = q_1$ and $h_m = q_n$, (ii) for each $i < m$ there exist a and b with $a < b$ such that $h_i = q_a$ and $h_{i+1} = q_b$, and every point q_c , with $a < c < b$, is to the left or on the line through h_i and h_{i+1} directed from h_i to h_{i+1} .

In the step of the algorithm discussed in this section, we make use of the fact that a circle is convex and so we also expect many of the measured points to be in convex position. We use this as a constraint on the partitioning of the laser scanner sequence. Let us assume for a moment

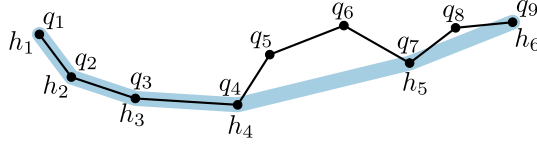


Figure 4.7: A point sequence $Q = [q_1, \dots, q_9]$ and its lower convex hull $\text{LCH}(Q) = [h_1, \dots, h_6]$. Note that points q_5 and q_6 are to the left of the line through h_4 and h_5 directed from h_4 to h_5 , and that point q_8 is to the left of the line through h_5 and h_6 directed from h_5 to h_6 .

that the central island’s border is a perfect circle and that no measurement errors are made by the laser scanner. Let P' be a subsequence of P . Then all points of P' are on a circle if all these points correspond to points on the central island’s border. As a consequence, P' must be convex from the laser scanner’s view point. So if three subsequent points p_i, p_{i+1}, p_{i+2} of P are not convex, then these three points cannot all be on the central island’s border.

We say that a partitioning \mathcal{S} of P is a *minimal convex partitioning* of P if all segments of \mathcal{S} are convex and the number of segments of \mathcal{S} is minimal under this convexity condition. It can be shown that there is exactly one minimal convex partitioning of P and therefore we can speak of *the* minimal convex partitioning of P . Let $\mathcal{S} = [S_1, S_2, \dots, S_m]$ be the minimal convex partitioning of P . Then if all points of a subsequence P' of P correspond to points on the central island’s border, there must be a segment S_i such that P' is a subsequence of S_i .

So by computing the minimal convex partitioning of P , one partitions P into segments satisfying the convexity condition without splitting subsequences of P for which all points correspond to points on the central island’s border. This is what we want to achieve in the step described in this section. Unfortunately, in reality the central island’s border is not an exact circle and the measurements made by the laser scanner are subject to measurement errors. As a consequence the subsequences of P for which all points correspond to points on the central island’s border, will not be convex in general. However, since the central island’s border is an ε_c -circle-approximation of a circle, it will have approximately the form of a circle and will therefore be approximately convex. Also because there is an upper bound on the laser scanner’s measurement error, a subsequence of P will be approximately convex if all its points correspond to points on the central island’s border. To make “approximately convex” concrete we introduce the concept of semi-convexity.

Definition 9 (ε -semi-convex). Let $Q = [q_1, \dots, q_{|Q|}]$ be a point sequence generated by a laser scanner and let $\text{LCH}(Q)$ be its lower convex hull. We say that Q is ε -semi-convex if and only if the following holds: For any two points q_i and q_j that are consecutive on $\text{LCH}(Q)$, and any point q_k with $i < k < j$, the distance from q_k to the line through q_i and q_j is at most ε . We refer to ε as the semi-convexity error.

As an example of this definition, the sequence in figure 4.7 is ε -semi-convex if the points q_5 and q_6 lie at distance at most ε of the line through q_4 and q_7 , and point q_8 lies at distance at most ε of the line through q_7 and q_9 .

4.4.2 Semi-Convexity of Sequences on the Central Island

For semi-convexity error $\varepsilon = 2\varepsilon_c + 2\varepsilon_m$ we can show that if a subsequence P' of P is not ε -semi-convex then not all of its points can be on the central island’s border. A proof of this directly follows by applying the following lemma.

Lemma 1. Let $P = [p_1, p_2, \dots, p_n]$ be a laser scan point sequence such that for each point p_i in P the distance between p_i and true point \hat{p}_i that induced measurement p_i is at most ε_m . Take two points p_k, p_m of P such that $k < m$ and for every i , with $k < i < m$, the point p_i is to the left or on the line through p_k and p_m directed from p_k and p_m . Let there be a point p_l in P such that $k < l < m$ and the shortest distance between p_l and the line passing through p_k and p_m is more than $2\varepsilon_c + 2\varepsilon_m$. Then there exists no circle C and Jordan curve Γ , such that Γ is an ε_c -circle-approximation of C and the true points $\hat{p}_k, \hat{p}_{k+1}, \dots, \hat{p}_{m-1}, \hat{p}_m$ are on Γ .

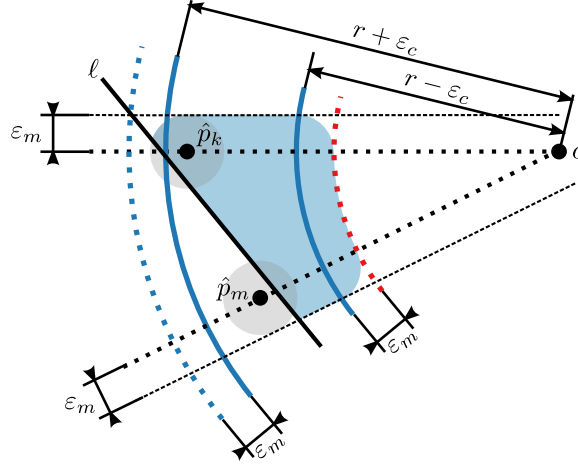


Figure 4.8: The blue area illustrates the area of possible locations of p_l . The two grey circles represent the possible locations of p_k and p_m given \hat{p}_k and \hat{p}_m .

Proof. Assume that there exists a circle with centre c and radius r , denoted as $\mathcal{C}(c, r)$, and an ε_c -circle-approximation Γ of this circle. Also assume that all the true points $\hat{p}_k, \hat{p}_{k+1}, \dots, \hat{p}_{m-1}, \hat{p}_m$ are on Γ . We now prove the correctness of the lemma by showing that for every $p_l, k < l < m$, the shortest distance $d(p_l, \ell)$ between p_l and the line ℓ passing through p_k and p_m is at most $2\varepsilon_c + 2\varepsilon_m$.

Note that since Γ is an ε_c -circle-approximation of circle $\mathcal{C}(c, r)$, the points $\hat{p}_k, \hat{p}_l, \hat{p}_m$ must be inside the annulus with centre c , inner radius $r - \varepsilon_c$ and outer radius $r + \varepsilon_c$ (see figure 4.8 where the border of this annulus is drawn with solid blue lines). Since Γ is star-shaped with respect to c , the points \hat{p}_l and \hat{p}_m must both be on the same side of the line passing through c and \hat{p}_k . Similarly \hat{p}_l and \hat{p}_k must both be on the same side of the line passing through c and \hat{p}_m . In figure 4.8 this means that \hat{p}_l must be on or between the two black dotted lines. Note that p_l and c must be on the same side of ℓ , because otherwise point \hat{p}_k must have been measured from the inside of the circle's border. All these constraints, in combination with the constraint that the distance between p_l and \hat{p}_l is at most ε_m , define an area R for all possible values for p_l . This area is highlighted in blue in figure 4.8. To find an upper bound for $d(p_l, \ell)$ we will search for the point p_{\max} in this area that has the largest shortest distance to line ℓ . Observe that such a point must be on the border of R .

We first search for a point on the part of R 's border that intersects with circle $\mathcal{C}(c, r - \varepsilon_c - \varepsilon_m)$ (this circle is drawn in figure 4.8 with a red dotted line), that has the largest shortest distance to line ℓ . We call the intersection of R with the circle, arc A . Let q be the point on circle $\mathcal{C}(c, r - \varepsilon_c - \varepsilon_m)$ for which the distance to line ℓ is maximal. Observe that point q cannot be on arc A . That implies that one of the two endpoints of arc A is closer to q on $(c, r - \varepsilon_c - \varepsilon_m)$ than all other points of A . Therefore one of A 's endpoints is farthest to ℓ of all points on A . Consequently if p_{\max} is on A , then p_{\max} is one of A 's endpoints.

Observe that one endpoint of A is on the line passing through \hat{p}_k and c and the other endpoint is on the line passing through \hat{p}_m and c (see figure 4.8). In the former (or latter) case \hat{p}_{\max} must be on the line passing through \hat{p}_k (or \hat{p}_m respectively) and c .

Note that p_{\max} will not be on the intersection between line ℓ and R . So what remains of the possible set of candidates for p_{\max} are those points on the border for which the real point \hat{p}_{\max} is either on the line passing through \hat{p}_k and c or on the line passing through \hat{p}_m and c . We now analyse the former case, which is symmetric to the latter case.

Let us assume that $\hat{p}_k, \hat{p}_{\max}, c$ are collinear. The maximum possible distance between any point in the circle $\mathcal{C}(\hat{p}_k, \varepsilon_m)$ and any point in the circle $\mathcal{C}(\hat{p}_{\max}, \varepsilon_m)$ is maximal if the distance is between \hat{p}_k and \hat{p}_{\max} is maximal. This is the case when \hat{p}_{\max} is on circle $\mathcal{C}(c, r - \varepsilon_c)$ and \hat{p}_k is on circle $\mathcal{C}(c, r + \varepsilon_c)$ as illustrated in figure 4.9. The maximum possible distance is then $2\varepsilon_c + 2\varepsilon_m$. The maximum possible shortest distance between any point in circle $\mathcal{C}(\hat{p}_{\max}, \varepsilon_m)$ and any line passing through the circle $\mathcal{C}(\hat{p}_k, \varepsilon_m)$ is at most the maximum possible distance between any point in circle

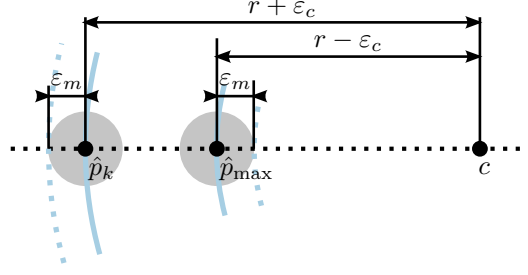


Figure 4.9: The case where \hat{p}_k , \hat{p}_{\max} and c are collinear.

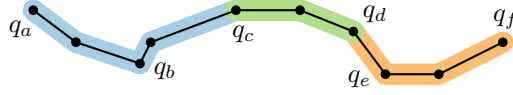


Figure 4.10: A point sequence (the black dots ordered from left to right) and two of its ε -semi-convex subsequences from q_a to q_d (the blue and green part) and from q_c to q_f (orange and green part). The part where the two ε -semi-convex sequences overlap is highlighted in green.

$\mathcal{C}(\hat{p}_{\max}, \varepsilon_m)$ and any point in circle $\mathcal{C}(\hat{p}_k, \varepsilon_m)$. We therefore can conclude that the distance between p_{\max} (which is in circle $\mathcal{C}(\hat{p}_{\max}, \varepsilon_m)$) and line ℓ (which passes through circle $\mathcal{C}(\hat{p}_k, \varepsilon_m)$) is at most $2\varepsilon_c + 2\varepsilon_m$.

Therefore the shortest distance $d(p_l, \ell)$ between p_l and ℓ is at most $2\varepsilon_c + 2\varepsilon_m$. \square

Note that it can be shown that the upper bound $2\varepsilon_c + 2\varepsilon_m$ in the previous lemma is tight (i.e. if we replace $2\varepsilon_c + 2\varepsilon_m$ by a smaller value, the lemma will not be true anymore).

For the ideal case of no measurement errors and a perfectly circular central island, discussed in section 4.4.1, we are able to generate a convex partitioning $\mathcal{S} = [S_1, S_2, \dots, S_m]$ of P in such way that if all points of a subsequence P' of P correspond to points on the central island's border, then there exists a segment S_i of \mathcal{S} such that P' is a subsequence of S_i . This is because every possible convex subsequence of P is a subsequence of one of the segments of the partitioning \mathcal{S} . This is not always true when we consider semi-convexity as stated by the following lemma.

Lemma 2. *Not for every point sequence $Q = [q_1, \dots, q_{|Q|}]$ and semi-convexity error ε it is possible to have an ε -semi-convex partitioning \mathcal{S} of sequence Q such that every possible ε -semi-convex subsequence of Q is a subsequence of one of the segments of \mathcal{S} .*

Proof. An ε -semi-convex subsequence $[q_i, \dots, q_j]$ of Q is called a *maximal ε -semi-convex subsequence* of Q if subsequences $[q_{i-1}, \dots, q_j]$ (if $i > 1$) and $[q_i, \dots, q_{j+1}]$ (if $j < |Q|$) are not ε -semi-convex. Consider the configuration illustrated by figure 4.10 where, for a specific ε , the sequences $[q_a, \dots, q_d]$ and $[q_c, \dots, q_f]$ are maximal ε -semi-convex. Note that the two ε -semi-convex sequences $[q_a, \dots, q_d]$ and $[q_c, \dots, q_f]$ overlap in $[q_c, \dots, q_d]$. Let us assume that there exists an ε -semi-convex partitioning \mathcal{S} of Q such that every ε -semi-convex subsequence of Q is a subsequence of one of the elements of \mathcal{S} . Then that means that the ε -semi-convex subsequence $[q_a, \dots, q_d]$ must be a segment of \mathcal{S} . However that would mean that no other segment of \mathcal{S} can contain the subsequence $[q_c, \dots, q_d]$ and therefore there will be no segment of \mathcal{S} of which $[q_c, \dots, q_f]$ is a subsequence. This contradiction completes the proof. \square

Lemma 2 tells us that if we partition into ε -semi-convex segments then it might be the case that we separate two subsequences of P that are both on the central island's border, because it is not always possible to create a partitioning of P of which each segment is maximal ε -semi-convex. This is not very desirable since in all following steps of the algorithm we only consider the points within a single segment as candidates for circle detection.

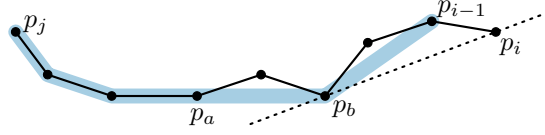


Figure 4.11: The semi-convex partitioning algorithm adds p_i to the segment $[p_j, \dots, p_{i-1}]$ if for each point between p_b and p_i the distance to the dashed line is at most $2\varepsilon_c + 2\varepsilon_m$.

4.4.3 Semi-Convex Partitioning Algorithm

As an alternative for creating a partitioning of P of which each segment is maximally ε -semi-convex, we can create a partitioning $\mathcal{S} = [S_1, S_2, \dots, S_m]$ of P of which each segment S_i corresponds to a maximal ε -semi-convex subsequence P' of P . Let P'_a be the overlap of subsequence P' and the maximal ε -semi-convex subsequence corresponding to segment S_{i-1} . Similarly let P'_b be the overlap of subsequence P' and the maximal ε -semi-convex subsequence corresponding to segment S_{i+1} . Instead that S_i contains all points of P' it contains the part of P' that does not overlap with other maximal ε -semi-convex subsequences, it contains P'_a if it is not in S_{i-1} and it contains P'_b if it is not in S_{i+1} . This means that each overlap of two maximal ε -semi-convex subsequences of P , appears in exactly one segment of \mathcal{S} .

With such a partitioning we make an “error” when we split a maximal ε -semi-convex subsequence at the point where it overlaps with another maximal ε -semi-convex subsequence. In practice, the length of the overlapping part of two maximal ε -semi-convex subsequences is expected to be small. Therefore the effect of the error we make using such a partitioning is expected to be rather limited (this is confirmed by the experimental results in section 4.6.3).

A partitioning as specified above can be computed by algorithm 4.1 which we will discuss now. The algorithm iterates over the points of point sequence P in the order of the sequence. While doing so, it keeps track of the lower convex hull of the segment that it is currently constructing (represented by variable S_{curr}). Thus, if the algorithm is at point p_i and the current segment starts at point p_j ($j < i$), then it has a list H containing the indices of those points in P that are on the lower convex-hull (marked blue in figure 4.11) of the segment $[p_j, \dots, p_{i-1}]$. When handling point p_i the algorithm decides whether p_i should be added to the current segment $S_{\text{curr}} = [p_j, \dots, p_{i-1}]$. The algorithm first searches (see the loop at line 7) for the last two subsequent points p_a and p_b in the lower convex hull for which p_i is not to the right of the line through p_a and p_b directed from p_a to p_b . After these two points are found, the algorithm verifies (see the loop at line 12) for each point on P between p_b ($b = H(k)$ after the execution of the loop at line 7) and p_i , whether the shortest distance between the point and the line passing through p_b and p_i (the dashed line in figure 4.11) is at most $2\varepsilon_c + 2\varepsilon_m$. If this is the case, then $[p_j, \dots, p_{i-1}, p_i]$ is ε -semi-convex and p_i is added to the current segment S_{curr} . Moreover all indices corresponding to points after point p_b are removed from H and index i of point p_i is appended to H . If not, then $[p_j, \dots, p_{i-1}, p_i]$ is not ε -semi-convex which forces the algorithm to terminate current segment at p_{i-1} and to start a new segment that is initialized by $[p_{i-1}, p_i]$.

Lemma 3. *The time complexity of the semi-convex partitioning algorithm is $O(n^2)$ where n is the number of points of P .*

Proof. The algorithm iterates over all n points of P . During the iteration for a point p_i , finding p_a and p_b requires in the worst case (in which $H = [p_1, p_2, \dots, p_{i-1}]$) i constant-time operations. So the complexity of finding p_a is $O(i)$. After p_a and p_b are found the algorithm scans through all points between p_b and p_i and computes their distance to the line passing through p_b and p_i . In the worst case $p_a = p_1$ for which this scan consists of $O(i)$ steps. So the total time complexity of one iteration step of the algorithm is $O(i)$, which gives us an $O(n^2)$ time complexity for the total algorithm. \square

```

Algorithm ComputeSemiConvexPartitioning( $[p_1, \dots, p_n]$ ,  $\varepsilon_c$ ,  $\varepsilon_m$ )
1.  $\varepsilon \leftarrow 2\varepsilon_c + 2\varepsilon_m$ ;  $\mathcal{S} \leftarrow []$ ;  $S_{\text{curr}} \leftarrow [p_1]$ ;  $H \leftarrow [1]$ ;  $i \leftarrow 2$ 
2. {Invariant:  $\mathcal{S} \text{ ++ } [S_{\text{curr}}]$  is an  $\varepsilon$ -semi-convex partitioning of  $[p_1, \dots, p_{i-1}]$ }
3. {Invariant:  $\text{LCH}(S_{\text{curr}}) = [p_{H(1)}, p_{H(2)}, \dots, p_{H(|H|)}]$ }
4. while  $i \leq n$ 
5.   do  $k \leftarrow |H|$ 
6.     {Invariant: points  $p_{H(k)+1}, p_{H(k)+2}, \dots, p_{i-1}$  are not in  $\text{LCH}(S_{\text{curr}} \text{ ++ } [p_i])$ }
7.     while  $k > 1$  and  $p_i$  is to the right of  $\vec{\ell}(p_{H(k-1)}, p_{H(k)})$ 
8.       do  $k \leftarrow k - 1$ 
9.        $b \leftarrow \text{true}$ 
10.       $r \leftarrow H(k) + 1$ 
11.      {Invariant:  $b = (\forall s : H(k) < s < r : d(p_s, \vec{\ell}(p_{H(k)}, p_i)) \leq \varepsilon)$ }
12.      while  $r < i \wedge b$ 
13.        do  $b \leftarrow (d(p_r, \vec{\ell}(p_{H(k)}, p_i)) \leq \varepsilon)$ 
14.           $r \leftarrow r + 1$ 
15.      if  $b$ 
16.        then  $H \leftarrow [H(1), \dots, H(k)] \text{ ++ } [i]$ 
17.           $S_{\text{curr}} \leftarrow S_{\text{curr}} \text{ ++ } [p_i]$ 
18.        else  $\mathcal{S} \leftarrow \mathcal{S} \text{ ++ } S_{\text{curr}}$ 
19.           $S_{\text{curr}} \leftarrow [p_{i-1}, p_i]$ 
20.           $H \leftarrow [i - 1, i]$ 
21.       $i \leftarrow i + 1$ 
22. return  $\mathcal{S} \text{ ++ } S_{\text{curr}}$ 

```

Algorithm 4.1: The semi-convex partitioning algorithm. Given a point sequence $[p_1, \dots, p_n]$, and parameters ε_c and ε_m , this algorithm computes a partitioning of the point sequence of which each segment is ε -semi-convex (with $\varepsilon = 2\varepsilon_c + 2\varepsilon_m$) and *corresponds* to a maximal ε -semi-convex subsequence of the point sequence. Here $\vec{\ell}(a, b)$ denotes the line through points a and p directed from a to b , and $H(t)$ denotes the t -th element of H .

4.5 Curvature-Based Partitioning

After partitioning the laser scan point sequence into ε -semi-convex segments, these ε -semi-convex segments are further partitioned into segments for which the change in curvature is small. We do this because a circle has a constant curvature and therefore two subsequences with a different curvature cannot correspond to points of the same circle. Note that a laser scanner subsequence corresponding to points of the central island's border will not be exactly on a circle. First because the border itself is not an exact circle but an ε_c -circle-approximation. Second because the point measurements of the border are subject to measurement errors. However, a subsequence corresponding to points on the border will deviate no more than $\varepsilon_c + \varepsilon_m$ from a circle.

In this subsection we describe an algorithm which splits a point sequence into segments at points where the curvature changes. This algorithm is applied to each segment of the partitioning generated by the semi-convex partitioning algorithm described in section 4.4. We redefine $P = [p_1, \dots, p_n]$ for this section to be the input point sequence to which the curvature-based partitioning algorithm is applied.

For a continuous planar curve the definition of the curvature at a given point on the curve can be described in multiple ways [WS93]. One of these ways is to define the curvature of a point p on a curve Γ as the reciprocal of the radius of the osculating circle of Γ at point p . The *osculating circle* at a point p on a curve Γ is the circle that passes through p and the points of Γ that are infinitesimally close to p . Another way is to define the curvature of a point p on a curve Γ as the rate of change of the direction of the tangent vector of Γ at point p . Note that these two

definitions of curvature are equivalent.

For a sequence of points it is not directly clear what curvature means. If we assume that a point sequence $Q = [q_1, \dots, q_{|Q|}]$ sufficiently approximates a curve Γ then we can say that the curvature of Q at one of its points q_i is equal to the curvature of Γ at q_i . So if Q is a good approximation of Γ then we can obtain Q 's curvature at q_i by computing the osculating circle that passes through the of points Q that are near q_i . Let us specify which points are considered to be near q_i . If the sequence Q does not contain any noise (i.e. all its points are on the curve Γ) then the osculating circle can be obtained by computing the circle passing through q_{i-1} , q_i and q_{i+1} . However in practice Q contains noise and the osculating circle must be approximated. To approximate the osculating circle at a point q_i we fit a circle to the points $[q_l, \dots, q_i, \dots, q_r]$, where l is the smallest index such that the distance between each point in $[q_l, \dots, q_{i-1}]$ and q_i is at most w and r is the largest index such that the distance between each point in $[q_{i+1}, \dots, q_r]$ and q_i is at most w . Here w is a given parameter which depends on the range of curvatures we wish to detect and characteristics of the input point sequence like the noise distribution or sampling density. Accompanied with w there is a second parameter ρ_Δ which specifies how accurately the curvature can be approximated given w and the input characteristics. Also in this step of the algorithm, circle fitting is done using the Newton-based implementation of the Taubin fit [Che10].

Let us assume that $\mathcal{S}' = [S'_1, S'_2, \dots, S'_m]$ is a given partitioning of the input point sequence P such that for each subsequence S'_i there exists a straight line ℓ for which the distance from each point of S'_i to ℓ is at most a specified maximum ε_s . We can then say that the tangent of sequence P does not change within a subsequence S'_i if we ignore deviations of size ε_s . We use only those points of P , which are the last point of one of the segments of \mathcal{S}' , as candidate points for splitting P as follows. Let p be the last point of a subsequence S'_i . Then there are two cases in which we want to split P at p . First if the curvature of P at p differs from the curvature at the last point of the previous subsequence S'_{i-1} . Second if the curvature of P is not constant within the combination of subsequences S'_i and S'_{i+1} . We start by verifying whether the second case holds. According to what we wrote above, the osculating circle of P at p is found by fitting a circle to the points within the window specified by w around p . However instead of doing that (since we know that within a segment the curvature does not change significantly) we extend the window such that it also covers S'_i and S'_{i+1} . If the resulting circle is not a good fit to the points within that window, then P is split at p since we know that the curvature of P is not constant within the combination of subsequences S'_i and S'_{i+1} . If the resulting circle is a good fit, then this circle will be a good approximation of the osculating circle at p . If the curvature of this circle differs more than ρ_Δ from the curvature at the last point of the preceding subsequence S'_{i-1} (i.e. when the first case holds), then P is split at p .

Above we used the partitioning \mathcal{S}' as a simplification (controlled by parameter ε_s) of P in order to speed up the algorithm by computing only the curvature of P at the reduced set of points, provided by the simplification, instead of all points of P . Furthermore this simplification introduces a desirable smoothing effect on the original point sequence. The right value for ε_s can be found empirically and may require some fine tuning. Choosing ε_s too large will decrease the accuracy of locating the points in P where the curvature changes. Due to the sensor resolution and noise characteristics there is a limit on how accurate curvature changes can possibly be detected. So choosing ε_s too small may not improve the accuracy but it will increase the computation time.

Until now we assumed that a partitioning \mathcal{S}' was given, but of course it also has to be computed. In the implementation used for the evaluation of the presented approach, we used the iterative end-point algorithm [Ram72] to compute \mathcal{S}' . This algorithm takes the vertices of a polyline as an input and produces a polyline consisting of a subset of the input vertices, such that the smallest distance of each input vertex to the output polyline is at most ε_s .

Algorithm 4.2 shows the pseudo code for the curvature-based partitioning algorithm.

Lemma 4. *The time complexity of the presented algorithm that, given the partitioning \mathcal{S}' , splits P at points where its curvature changes, is $O(mw/d_{\max} + n)$. Here n is the number of points of P , m is the number of segments of \mathcal{S}' and d_{\max} is the upper bound on the point density of P .*

Algorithm *ComputeCurvatureBasedPartitioning* ($[p_1, \dots, p_n]$, ε_s , w , ρ_Δ)

1. Compute a partitioning $[S'_1, \dots, S'_m]$ such that for each subsequence S'_i there exists a straight line ℓ for which the distance from each point of S'_i to ℓ is at most ε_s .
2. $\mathcal{S} \leftarrow []$; $S_{\text{curr}} \leftarrow S'_1$; $i \leftarrow 1$; $j \leftarrow |S'_1|$; $b \leftarrow \text{true}$
3. {Invariant: $[S'_1, \dots, S'_i]$ is a partitioning of point sequence $[p_1, \dots, p_j]$ }
4. {Invariant: $\mathcal{S} \text{ ++ } [S_{\text{curr}}]$ is a partitioning of point sequence $[p_1, \dots, p_j]$ }
5. {Invariant: $b = ((i > 1) \wedge (S'_{i-1} \text{ a subsequence of } S_{\text{curr}}))$ }
6. **while** $i < m$
7. **do** $l \leftarrow 0$
8. {Invariant: $(\forall e : j - l \leq e < j : d(p_j, p_e) \leq w)$ }
9. **while** $j - l > 1 \wedge d(p_j, p_{j-l-1}) \leq w$
10. **do** $l \leftarrow l + 1$
11. **if** $l < |S'_i| - 1$
12. **then** $l \leftarrow |S'_i| - 1$
13. $r \leftarrow 1$
14. {Invariant: $(\forall e : j < e \leq j + r : d(p_j, p_e) \leq w)$ }
15. **while** $j + r < n \wedge d(p_j, p_{j+r+1}) \leq w$
16. **do** $r \leftarrow r + 1$
17. **if** $r < |S'_{i+1}| - 1$
18. **then** $r \leftarrow |S'_{i+1}| - 1$
19. $c \leftarrow \text{FitCircle}([p_{j-l}, \dots, p_{j+r}])$
20. $\rho_c \leftarrow \text{curvature of } c$
21. **if** c is a good fit and $(-b \vee |\rho - \rho_{\text{prev}}| \leq \rho_\Delta)$
22. **then** $S_{\text{curr}} \leftarrow S_{\text{curr}} \text{ ++ } [p_{j+1}, \dots, p_{j+|S'_{i+1}|-1}]$
23. $\rho_{\text{prev}} \leftarrow \rho$
24. $b \leftarrow \text{true}$
25. **else** $\mathcal{S} \leftarrow \mathcal{S} \text{ ++ } [S_{\text{curr}}]$
26. $S_{\text{curr}} \leftarrow [p_j, \dots, p_{j+|S'_{i+1}|-1}]$
27. $b \leftarrow \text{false}$
28. $j \leftarrow j + |S'_{i+1}| - 1$ {minus one since S'_i and S'_{i+1} have one point in common}
29. $i \leftarrow i + 1$
30. **return** $\mathcal{S} \text{ ++ } S_{\text{curr}}$

Algorithm 4.2: The curvature-base partitioning algorithm.

Proof. At each segment S'_i of \mathcal{S}' we have to compute the circle that fits the points within the window around the last point of S'_i . The upper bound d_{max} on the point density also gives us the upper bound $\lfloor w/d_{\text{max}} \rfloor$ on the number of points to the left/right of the last point of S'_i within a window with width $2w$. If this window does not cover all points of segments S'_i and S'_{i+1} , then the algorithm extends the window such that it covers both segments. So the number of points, within the window around the last point of S'_i , is at most $1 + \max(\lfloor w/d_{\text{max}} \rfloor, |S'_i|) + \max(\lfloor w/d_{\text{max}} \rfloor, |S'_{i+1}|) \leq 1 + 2\lfloor w/d_{\text{max}} \rfloor + |S'_i| + |S'_{i+1}|$. Since the circle fitting algorithm is a linear-time algorithm, the total time complexity of the algorithm is:

$$O\left(\sum_{i=1}^{m-1} \left(1 + 2\lfloor \frac{w}{d_{\text{max}}} \rfloor + |S'_i| + |S'_{i+1}| \right)\right) = O\left(\frac{wm}{d_{\text{max}}} + n\right) \quad \square$$

Note that d_{max} can be computed given the laser scanner's horizontal resolution and the minimal distance between the laser scanner and the points of P . Computing \mathcal{S}' using the iterative end-point algorithm has a time complexity of $O(mn)$.

4.6 Evaluation

4.6.1 Evaluation Goals and Experimental Setup

Section 4.2 gives a formal specification for the problem of detecting circles in point sequences. The goal of the evaluation in this section is not only to evaluate how well the proposed algorithm works as a circle detector according to this specification, but also how well the system as a whole performs as a central-island detector. The system could have an excellent performance as a circle detector, but may be very bad as a central-island detector. This would, for example, be the case when the laser ray of the laser scanner never hits the central island's curb. Therefore the goal of this evaluation is to answer the following questions:

- How good is the input data of the circle detection algorithm? Does it contain point sequences that are on the central island's border?
- If the input contains a sequence on the central island's border, is it detected by the detector?
- How often is a circle detected that does not correspond to a central island?
- What are the characteristics of the detected circles?
- Are there any differences among the detection results for different roundabouts and/or gates?

The following subsections specify these questions and their answers in more detail.

The evaluation is based on sensor recordings by a testing vehicle done at six different roundabouts. An overview of the roundabouts can be found in appendix A. Each of these roundabouts has four entrance gates, where each entrance gate was approached three times. This gives a total of 72 recordings.

The geometry of each of the roundabouts was measured, in global coordinates, using a highly accurate DGPS device. Not only the geometry of the central island was measured, but also that of the outer border, the gate endpoints and the splitter islands. For the evaluation in this chapter, however, only the central island's geometry is used. For each central island multiple points were measured at different locations on the top of the central island's curb. The global uncertainty of each measured point never exceeded 2 cm. These points were used to fit a circle, which gave a radius, a centre in global coordinates and a fit error. For all roundabouts, the root mean square error of the fitted circle was at most 0.02 m. For the whole set of roundabouts the largest distance of a measured point to the fitted circle was 0.06 m.

The testing vehicle used for the recordings was equipped with a highly accurate positioning system, consisting of DGPS and an inertial measurement unit, that logged the vehicle's global position and orientation during the recording at a frequency of 100 Hz. The orientation and position measurement error by this system never exceeded 0.08° and 4 cm respectively. Using the geometry of the central island in global coordinates and the vehicle's global pose, the central island's geometry relative to the vehicle was computed for each time instant where a laser scan was done. These relative geometries are used as the ground truth in this evaluation.

Since we want to compare the detection performance for different roundabouts and different gates we tried to keep two of the recording aspects the same across all recordings. First, the velocity at which the vehicle approached the roundabout was approximately 20 km/h. This was done because a difference in velocity may result in a difference in the density in which an area is scanned by the laser scanner (i.e. if the vehicle drives slower, there will be more laser scans per area). Second, the view to the central island was kept free in the sense that no other vehicles occluded the view to the central island's border for the laser scanner. Of course a robust central-island detector should also be able to detect a central island when other vehicles are on the roundabout, but having recordings free of occlusions by other vehicles allows us to do a fair comparison on the performance at different roundabouts and gates. The performance of the detector for cases where the roundabout is (partially) occluded by other vehicles, could be a topic of future evaluations and some existing measurements have already shown that also in these cases the central island is detected.

For this evaluation the algorithm was parametrized as follows: $l_{\min} = 0.1$, $\rho_{\Delta} = 0.03 \text{ m}^{-1}$, $\varepsilon_c + \varepsilon_m = 0.3 \text{ m}$, $\varepsilon_s = 0.35 \text{ m}$, $n_{\min} = 50$, $d_{\text{nb}} = 0.5 \text{ m}$, $f_{\min} = 0.95$, $w = 1 \text{ m}$. Details about the laser scanner that was used are given in section 2.3.2.

Section 4.4 tells us that for the semi-convex partitioning algorithm, the parameter ε , defining the maximum semi-convexity error, should be $2\varepsilon_c + 2\varepsilon_m$. However, practice showed that using value $\varepsilon = \varepsilon_c + \varepsilon_m$, the number of false central island detections decreases drastically while the decrease in the number of true detections is negligible (this will be shown in section 4.6.3). Therefore for this evaluation $\varepsilon = \varepsilon_c + \varepsilon_m$ is used.

The computation time of the algorithm on single-layer laser scanner point sequences of 581 points never exceeded 2 ms on an Intel Core i7 PC. This means that less than 6 ms is needed to process the scans by three layers, which is a small fraction of the 40 ms cycle time at which the laser scanner produces laser scans for three layers simultaneously. These computation times indicate that the algorithm could be used in a real-time system for online roundabout detection.

4.6.2 Input Data Quality

Before evaluating the performance of the presented algorithm, the quality of the algorithm’s input is evaluated by investigating how often the laser scanner sequence contains a subsequence corresponding to the central island. Knowing the ground truth for the central island’s geometry relative to the vehicle (something that is not known to the algorithm), *benchmark sequences* are extracted from the laser scan with the following specifications:

1. A benchmark sequence should have at least the minimum number of points n_{\min} .
2. The arc of the ground truth circle, with radius r , covered by a benchmark sequence should be at least of length $2\pi r l_{\min}$.
3. The distance between neighbouring points in a benchmark sequence is at most d_{nb} .
4. At least 95% of a benchmark sequence’s points has a distance of at most $\varepsilon_c + \varepsilon_m = 0.3 \text{ m}$ to the ground truth circle.
5. All points of a benchmark sequence have a distance of at most 0.6 m to the ground truth circle.
6. A benchmark sequence is maximal in the sense that there exists no other subsequence of the benchmark sequence, satisfying all the constrains above.

Note that this specification largely corresponds to the specification in section 4.2 of circles that should be detected by the presented algorithm. The main difference however is that we allow at most 5% of the points to be further away than $\varepsilon_c + \varepsilon_m$ from the reference circle. This is done because of cases like the one illustrated in figure 4.12. In such cases for the majority of the points the distance to the reference circle is less than $\varepsilon_c + \varepsilon_m$, but for all points the distance to the sequence’s least-square circle is less than $\varepsilon_c + \varepsilon_m$. According to the specification in section 4.2, the circle detector should detect such a circle and since the point sequence still corresponds to the central island, we also want such a point sequence to be a benchmark sequence.

Table 4.1 gives an overview of the number of detected benchmark sequences for each of the studied roundabout gates. Compared to all other roundabouts the recordings at all gates of the Casino roundabout contain the highest number of benchmark sequences with an average of approximately 100 per recording. The most plausible explanation for this is that the Casino roundabout has the highest central island curb and therefore it is more likely that the laser ray reflects at the curb. A second roundabout where the laser scanner performed well at “hitting” the central island curb at all gate, is the Gaimersheim Large roundabout. The Gaimersheim Small roundabout has the smallest central island among the roundabouts of the experiment and has only a few benchmark sequences per approach. A plausible explanation for the low number of benchmark sequences may be the relatively low curb height of 8 cm. The reason why almost no

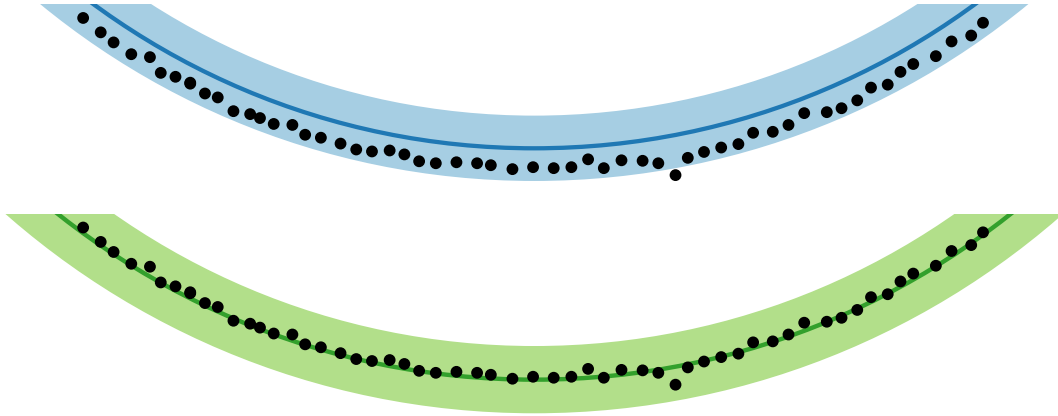


Figure 4.12: Above you see the ground truth circle, coloured dark blue, and the area for which the distance to the ground truth circle is not more than $\varepsilon_c + \varepsilon_m$, coloured bright blue. Also a set of points can be seen in which there is one point with a distance larger than $\varepsilon_c + \varepsilon_m$ to the ground truth circle. Below the dark green circle is the least square circle of that set of points and the bright green area is the area for which the distance to the least square circle is not more than $\varepsilon_c + \varepsilon_m$. One can see that for all points the distance to the least square circle is less than $\varepsilon_c + \varepsilon_m$.

benchmarks were measured at gate 1 of Gaimersheim Small, can be explained by the fact that the road that approaches the gate goes uphill before it reaches the roundabout.

A possible reason for having no benchmarks at gate 0 of the Gymnasium roundabout could be that the road bends just before the roundabout and because the road goes slightly down hill. The absence of benchmarks at gate 1 of the Gymnasium roundabout could be because the approach lane is far from perpendicular to the roundabout. However further analyses of the sensor in combination with this roundabout need to be done to find an explanation for the overall bad performance at the Gymnasium roundabout.

When visually inspecting the laser scans at roundabout MX303, it appears that the laser reflects mostly at the hill on the roundabout instead of the central island's curb. Maybe the curb does not have such a good reflectivity as its surrounding area (the hill behind it and asphalt in front of it), which could be an explanation for these results. Future analyses are required to find out whether this is the cause.

Although the curbs of roundabouts MX303 and Gymnasium are higher than the curb of the Lana Grossa roundabout, there are more benchmarks at gate 1 and 2 of Lana Grossa than at the gates of MX303 and Gymnasium. The photos of the curbs in appendix A show that the curbs of roundabouts MX303 and Gymnasium are less steep compared to the other roundabouts. This means that the angle of incidence of the laser beam will be higher for those curbs, which may be a cause for the lower number of benchmark sequences.

The laser scanner performs quite well at gates 1 and 2 of the Lana Grossa roundabout. The road approaching gate 3 is slightly rolled in relation to the roundabout (i.e. the plane of the road's surface is not parallel to the plane of the roundabout) which could be an explanation for its bad performance there, but whether this is the main cause of the bad performance at gate 0 can only be answered after further analysis.

It should be noted that in previous experiments done at the Lana Grossa roundabout with an older version of the laser scanner, the measurements contained significantly more benchmark sequences. The older version of the laser scanner and the one used for the experiment presented in this chapter differ slightly in their optical properties, that are known to make the older version more sensitive for measuring points on the ground. These measurements are not included in this evaluation since they were only done at the Lana Grossa roundabout and therefore cannot be used for a fair comparison with the performance at other roundabouts. However these differences in performances between two versions of the sensor show that a significant part of the performance

Roundabout	Gate	Total # Bench	Avg # Bench	Min # Bench	Max # Bench
Lana Grossa	0	0	0	0	0
Lana Grossa	1	67	22.3333	17	27
Lana Grossa	2	82	27.3333	23	36
Lana Grossa	3	0	0	0	0
MX303	0	2	0.66667	0	2
MX303	1	40	13.3333	11	15
MX303	2	53	17.6667	13	26
MX303	3	19	6.3333	3	9
Gymnasium	0	0	0	0	0
Gymnasium	1	0	0	0	0
Gymnasium	2	32	10.6667	9	14
Gymnasium	3	37	12.3333	11	15
Casino	0	301	100.3333	99	103
Casino	1	304	101.3333	92	110
Casino	2	326	108.6667	92	117
Casino	3	280	93.3333	88	97
Gaimersheim Small	0	36	12	8	17
Gaimersheim Small	1	2	0.66667	0	2
Gaimersheim Small	2	42	14	11	18
Gaimersheim Small	3	28	9.3333	8	11
Gaimersheim Large	0	216	72	68	77
Gaimersheim Large	1	252	84	78	91
Gaimersheim Large	2	336	112	100	120
Gaimersheim Large	3	196	65.3333	63	67

Table 4.1: This table shows for the three recordings of each studied roundabout gate the total, average, minimum and maximum number of bench mark sequences.

in detecting the central island depends on the characteristics (e.g. sensitivity, low-level processing of the raw sensor data) of the sensor instead of only on the geometry of the roads and roundabout and geometrical properties of the scan layers.

Despite different performances at different roundabouts and different gates, we see that at each gate the results are very well reproducible since per gate the variations in number of benchmark sequences is very small.

It can be concluded that the performance of the laser scanner measuring the central island’s curb, varies among the different roundabouts. The exact reasons for a low number of benchmark sequences at certain roundabouts and gates cannot be given in general and should be subject of future research. However, it is shown that the principle of measuring the central island curb with a laser scanner works. Since it makes no sense to evaluate the performance of the central island detection algorithm for input where there is nothing to detect, only the following roundabouts were selected for further evaluation in the rest of this section: Casino, Gaimersheim Large and gates 1 and 2 of Lana Grossa.

4.6.3 Detection Performance

In this part we evaluate how well the detection algorithm performs as a central-island detector. Ideally, as a central-island detector, the algorithm detects a circle for each benchmark sequence and no other circles. In order to analyse how much the algorithm deviates from this ideal behaviour, we count the number of benchmark sequences for which the algorithm detected no circle. In other words: we count the number of times the algorithm did not detect the central island although it should have detected it. We also count the total number of detections by the algorithm in order

Roundabout	Num Bench	Num Circles	B w/o C
Lana Grossa	149	440	15
Casino	1211	1599	29
Gaimersheim Large	1000	1601	6
Total	2360	3039	50
Total ⁽¹⁾	2144	3039	16
Total ⁽²⁾	2360	4186	39

Table 4.2: For each roundabout the number of benchmark sequences, number of circles detected by the algorithm and the number of benchmarks sequences for which the algorithm did not detect a circle (B w/o C). The row “Total⁽¹⁾” shows the total values if we do not allow benchmark sequences to have points with a distance more than 0.3m from the ground truth circle. Row “Total⁽²⁾” shows the total values if the $\varepsilon = 2\varepsilon_c + 2\varepsilon_m$ is used for the semi-convex partitioning step.

to get an idea how many circles are detected that do not correspond to the central island.

Although ideally the detector should not detect any other circles than the central island, in section 4.2 the problem of detecting central islands is specified as the problem of finding circles in point sequences generated by a laser scanner. So it will be inevitable that other circular objects are detected by the algorithm as well. Since the last step of the algorithm performs a circle fit and checks the quality of the fitted circle, the algorithm will guaranteed never report detections that do no correspond to circles with the criteria specified in section 4.2. This means that all circles that are detected by the algorithm and that do not correspond to the central island, are not false circle detections but are false central island detections.

We say that a benchmark was detected by the algorithm if and only if the algorithm has detected a circle of which the corresponding point sequence contains at least 80% of the points of the benchmark sequence. Since it is not expected from the algorithm to split the input sequence at exactly the endpoints of the benchmark sequence, this empirical value of 80% is chosen to specify that the point sequence corresponding to the detected circle must contain a large part of the benchmark sequence. Figure 4.13 shows an example of a benchmark sequence that was successfully detected by the algorithm. It also shows how the semi-convex partitioning step split up the input sequence at the point where it starts to drift away from the ground truth circle. A successful detection, for which the fitted circle better matches the ground truth circle, can be seen in figure 4.14.

Table 4.2 shows for roundabout Gaimersheim Large, Casino and Lana Grossa (only gates 1 and 2) the number of benchmark sequences, the total number of detected circles and the number of benchmark sequences that were not detected. We see that the number of benchmarks that were not detected is relatively low. This number is relatively the highest for the Lana Grossa roundabout where 10.1% of the benchmarks were not detected. For the Casino and Gaimersheim Large roundabouts less than 2.5% of benchmarks were not detected.

Let us now have a look at what the causes are for benchmarks not being detected. The basic principle of the presented algorithm is to split up the point sequence in the input into smaller subsequences and to try to fit a circle to each final subsequence satisfying certain criteria. This means that the algorithm can make an error by splitting up a sequence at a wrong position, or by not splitting up a subsequence while it should have split it. The former kind of error is called an *unnecessary split* and the latter is called a *missed split*. If a missed/unnecessary split is the cause for an undetected benchmark sequence, then it is called a fatal missed/unnecessary split. We distinguish three kinds of causes for benchmarks not being detected: fatal unnecessary splits in the semi-convex partitioning step, fatal unnecessary splits in the curvature-based partitioning step and fatal missed splits. The main cause of each benchmark not being detected is considered to be one of these three kinds based on the following criteria:

- Let P' be the benchmark sequence and Q be the set of points in P' where the semi-convex partitioning step made splits. Let \mathcal{S} be the partitioning of P' that is obtained when P' is

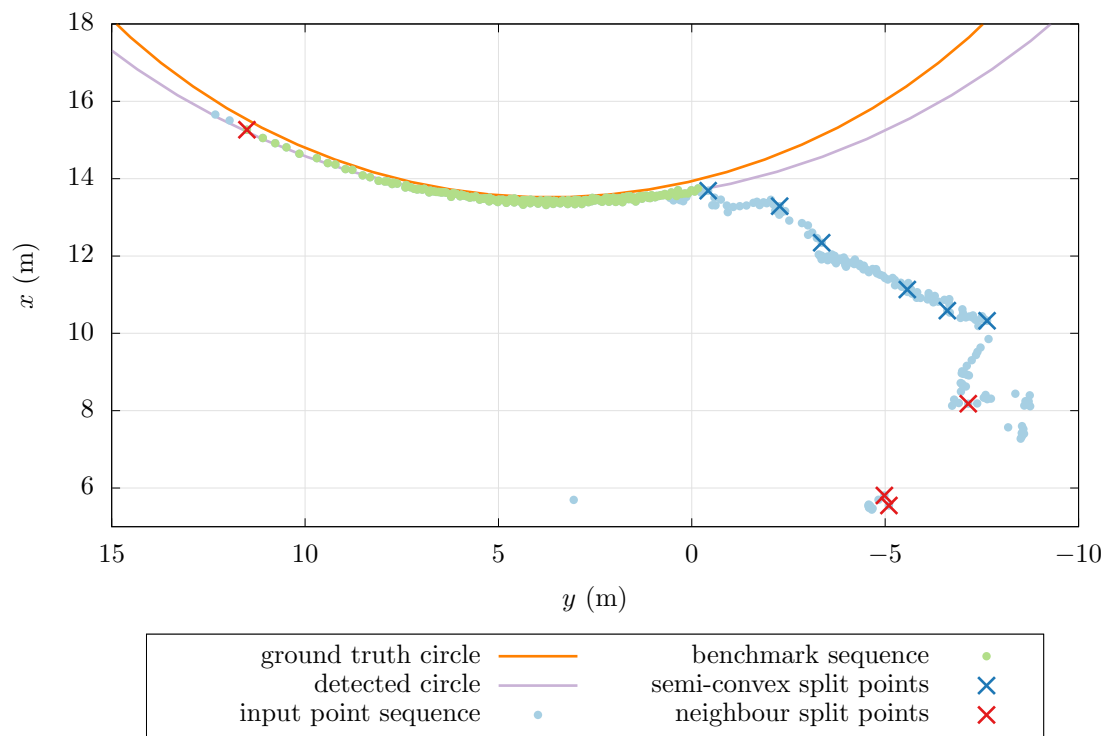


Figure 4.13: An example where the algorithm correctly splits the input point sequence at the point where it starts to move away from the central island.

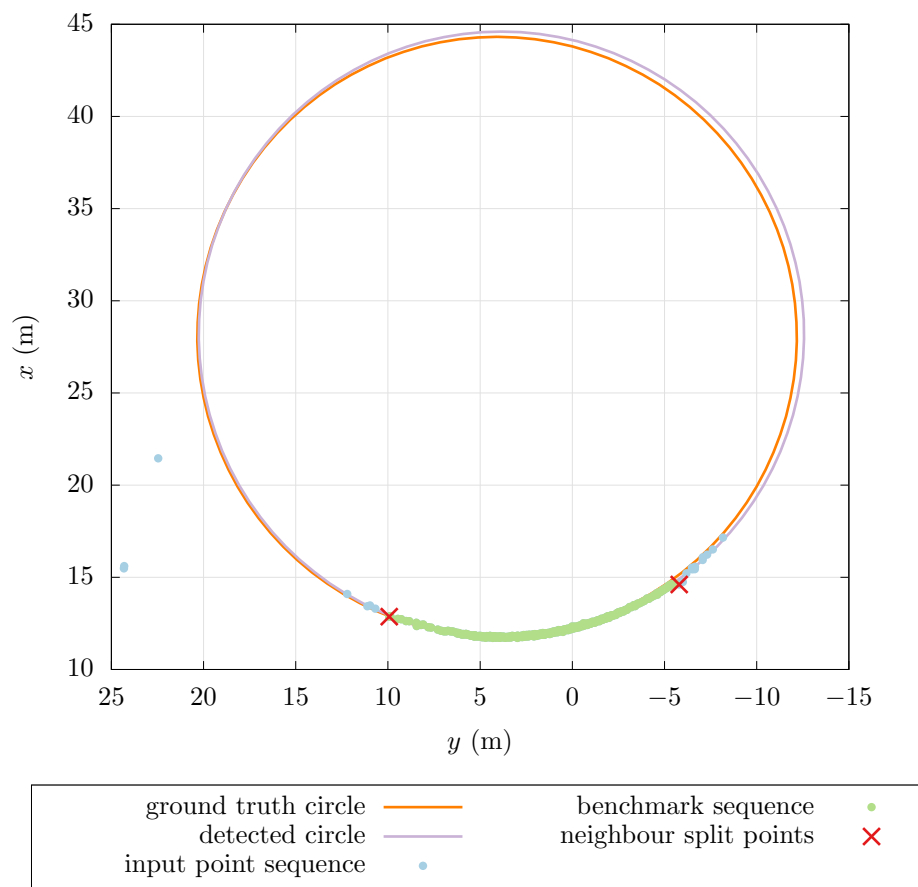


Figure 4.14: An example where the detected circle is very similar to the ground truth circle.

split at the points that are in Q . Assume for partitioning \mathcal{S} we would run the final step of the algorithm which fits a circle and checks the specified constraints. Then if in that final step no circle is detected that has at least 80% of the points of P' (i.e. there is no detected circle that is a detection of the benchmark sequence), the cause of the benchmark not being detected is a fatal unnecessary semi-convex split.

- Let P' be the benchmark sequence and Q be the set of points in P' where both the semi-convex and curvature-based partitioning step made splits. Let \mathcal{S} be the partitioning of P' that is obtained when P' is split at the points that are in Q . Assume for partitioning \mathcal{S} we would run the final step of the algorithm which fits a circles and checks the specified constraints. Then if in that final step no circle is detected that has at least 80% of the points of P' , the cause of the benchmark not being detected is a fatal unnecessary curvature-based split, unless the cause is a fatal unnecessary semi-convex split.
- If the cause of not detecting a benchmark is not a fatal unnecessary split then it must be a fatal missed split.

Theoretically there may be a fourth cause for not detecting a benchmark and that is the possibility that the least-square circle for the benchmark sequence does not satisfy the constraints that are verified by the last step of the algorithm. However this case never occurred in any of the evaluation runs and is therefore not mentioned in the remainder of this section.

Table 4.3 gives an overview of the frequency of each of these causes for the evaluated recordings. The first row of the table shows the data for the configuration in which we want to evaluate the algorithm, that is, the configuration in which the semi-convex partitioning step allows a maximum semi-convexity error of $\varepsilon_c + \varepsilon_m$ and where at least 95% of the benchmark sequence's points must have a distance of at most 0.3 m to the ground truth circle. In this configuration the majority of the causes are the fatal semi-convex splits and fatal missed splits. Since we allow 5% of the points of a benchmark sequence to have a distance between 0.3 m and 0.6 m it may be possible that some benchmark sequences are sequences that cannot correspond to an ε_c -circle approximation in the real world. If we only consider those benchmark sequences that have all points within the 0.3 m margin, then table 4.2 shows that we have 2144 instead of 2360 benchmark sequences and that the number of undetected benchmarks is 16 instead of 50. For this case the second row of table 4.3 shows that the number of fatal semi-convex splits decreases from 34 to only 6.

Let us now turn to the explanation for choosing the maximum semi-convexity error to be $\varepsilon_c + \varepsilon_m$ instead of $2\varepsilon_c + 2\varepsilon_m$. In table 4.2 we can see that by choosing $\varepsilon_c + \varepsilon_m$ instead of $2\varepsilon_c + 2\varepsilon_m$, the total number of detected circles decreases by 1147 while the number of undetected benchmark circles only decreases by 11. So this means that by choosing $\varepsilon_c + \varepsilon_m$ instead of $2\varepsilon_c + 2\varepsilon_m$ we detect fewer circles that do not correspond to the central island while detecting negligibly fewer circles that do correspond to the central island. This justifies our choice since it results in an algorithm of which the behaviour is closer to its ideal behaviour.

The third row of table 4.3 shows that if we would use a maximum semi-convexity error of $2\varepsilon_c + 2\varepsilon_m$, then the number of missed splits increases from 13 to 32. Figure 4.15 shows an example of such a missed split that does not occur in case $\varepsilon = \varepsilon_c + \varepsilon_m$: since the sequence is split too far away from the benchmark sequence, the least-square circle for the remaining sequence was not a good fit and therefore no circle was detected. This missed split is not the result of an error by the semi-convex partitioning algorithm, since for the remaining sequence there actually exists a circle that is a good fit for that sequence. Instead the problem is caused by our specification that not just any circle, but the least-square circle of the sequence should be a good fit. Note that such problems may also occur when $\varepsilon = \varepsilon_c + \varepsilon_m$, but at a lower rate. The first row of table 4.3 shows that for all 2360 benchmark sequences in our experiment, this error could not have occurred more than 13 times. Therefore this ‘‘flaw’’ in the problem specification has an almost negligible impact on the detector's performance in practice.

In section 4.4.3 we can read that the only error the semi-convex partitioning can make is that it can split off a part at the beginning or end of a maximal ε -semi-convex subsequence that overlaps with another maximal ε -semi-convex subsequence. The length of such an overlapping part

ε	Allow Holes	Semi-Convex	Curvature	Missed
$\varepsilon_c + \varepsilon_m$	yes	34	3	13
$\varepsilon_c + \varepsilon_m$	no	6	3	7
$2\varepsilon_c + 2\varepsilon_m$	yes	2	5	32
$2\varepsilon_c + 2\varepsilon_m$	no	0	2	27

Table 4.3: This table shows, for different cases, the number of fatal unnecessary splits in the semi-convex partitioning step, the number of fatal unnecessary splits in the curvature-based partitioning step and the number of fatal missed splits. Column ε shows for each case the maximum allowed semi-convexity error used by the semi-convex partitioning step. Column “Allow Holes” shows for each case whether we allowed benchmark sequences to have points with a distance larger than 0.3 m from the ground truth circle.

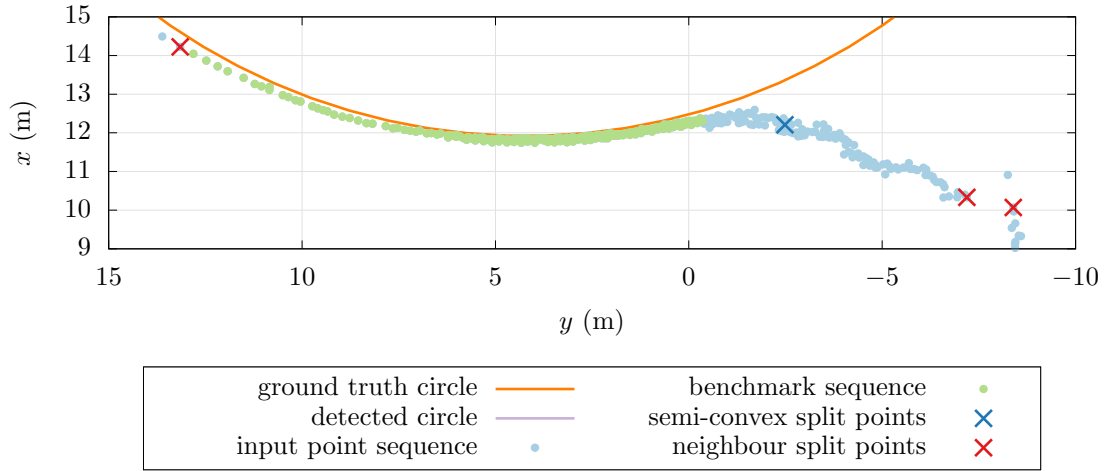


Figure 4.15: An example of a missed split that occurs when the maximum semi-convexity error ε is set to $2\varepsilon_c + 2\varepsilon_m$ instead of $\varepsilon_c + \varepsilon_m$. As a result no circle is detected.

is expected to be small and thus the effect of this error will be limited. This expectation (which assumes $\varepsilon = 2\varepsilon_c + 2\varepsilon_m$) is confirmed by the fourth row of table 4.3 where for each benchmark sequence there exists a real world ε_c -circle approximation and in which none of the failed detections were caused by the semi-convex partitioning step.

4.6.4 Detection Rate Depending on the Distance to the Central Island

The main focus of the previous subsection was to analyse how well the detector detects a central island if it was present in the input data. In this subsection the goal is to evaluate how large the fraction of all detected circles is, that correspond to the central island. We analyse the size of this fraction as a function of the vehicle’s distance to the border of the central island.

We classify each detected circle as either a good detection or a bad detection. A *good detection* is a circle for which at least 95% of its corresponding points have a distance of at most 0.3 m to the ground truth circle. All other detections are classified as a *bad detection*. We define the distance from the vehicle to the central island to be the distance of the vehicle rear axle’s centre (which is 3.84 m behind the laser scanner) and the border of the central island.

Figure 4.16 shows the number of detections (good and bad) as a function of the vehicle’s distance to the central island. It also visualises the contribution of each of the laser scanner’s layers. Figure 4.17 shows a similar plot, but then only for the number of circles that were classified

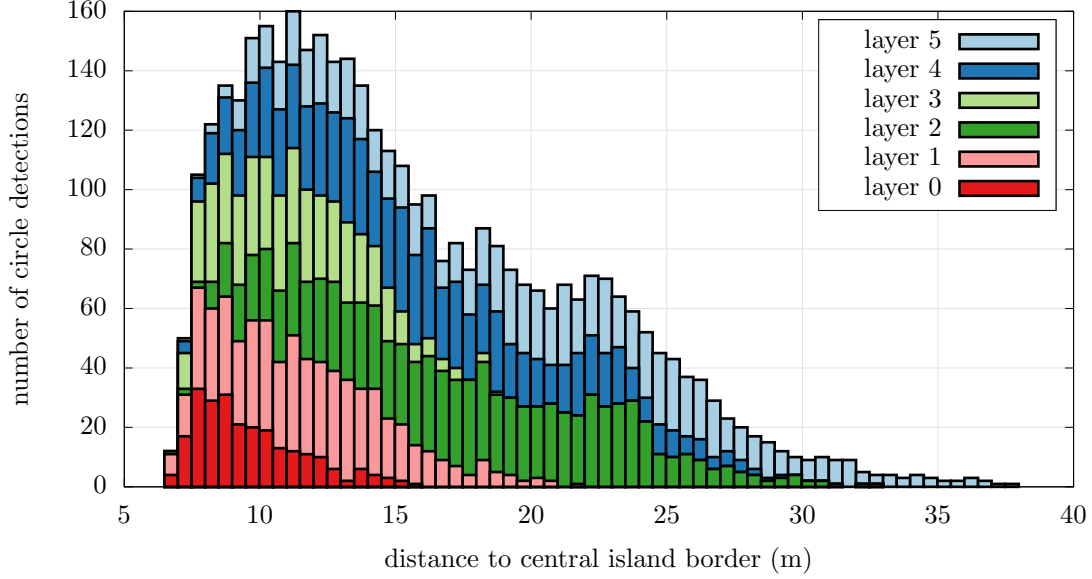


Figure 4.16: This figure shows for each distance bin the number of detected circles, where the bars representing the number of detected circles for the individual layers are stacked on top of each other.

as good detections. Figure 4.18 visualizes the contribution of good and bad detections to the total number of detections.

The data visualised by figures 4.16 and 4.17 allows us to create plots like figure 4.19 that show, for a specific layer, the good-to-total detection ratio as a function of the distance to the central island. Such ratio can be interpreted as the probability that a detected circle is a good detection, given the vehicle's distance to the central island. Plots for each of the six layers can be found in appendix B and are used for the measurement models discussed in chapters 5 and 6.

4.6.5 Error of Fitted Circles

In this subsection we analyse the characteristics of the circles classified as good detections. This is done by deriving an error distribution for the circle's centre and radius and a distribution for the laser scanner points relative to the ground truth circle. For each of the one-dimensional or two-dimensional histograms in this subsection, the distribution is represented as a probability density distribution. This means that for a bin b the value v_b of the bin is computed as:

$$v_b = \frac{n_b}{w \sum_d n_d}$$

where n_i is the number of circle/points corresponding to bin i and w is the width (or area in case of a 2D histogram) of the bins. This means that instead having bin values representing the number of circles/points within that bin, we have a histogram with the same shape but for which the area (or volume in case of a 2D histogram) of all bins sums to one. The reason for representing the histograms as probability density distributions is that some of these histograms can then directly be used as probability density distributions that are required by the measurement models in later chapters.

Figure 4.21 shows the error distribution in the x and y direction of the circle centres and figure 4.22 shows the error distribution of the radius. Striking is that the order of magnitude of the radius error (being metres) is much larger than the order of magnitude of the circle fit error allowed by the algorithm (being decimetres). This can be explained by the fact that the

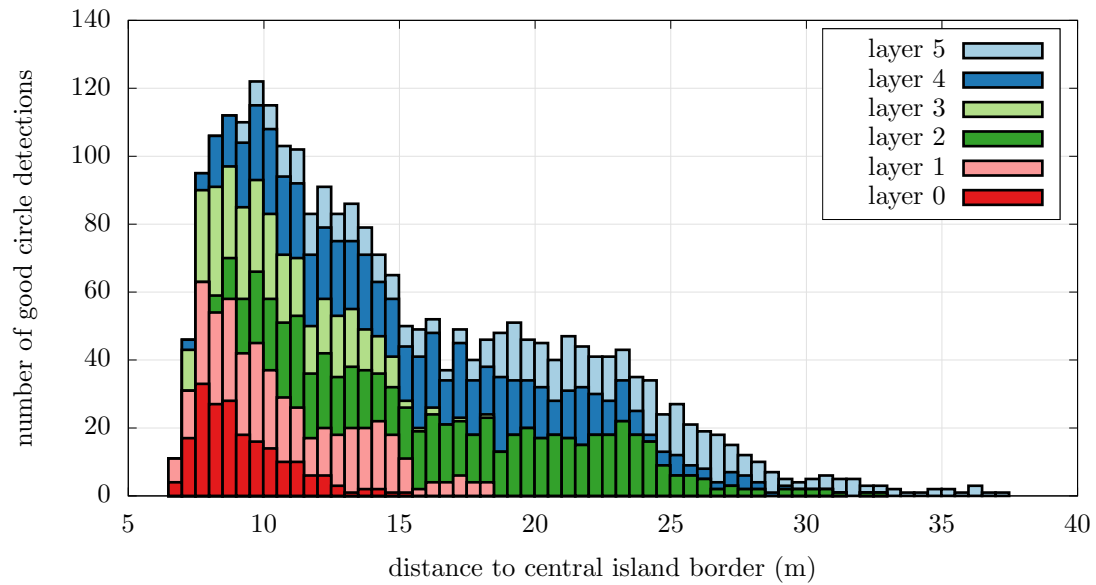


Figure 4.17: This figure shows for each distance bin the number of good circle detections, where the bars representing the number of good detections for the individual layers are stacked on top of each other.

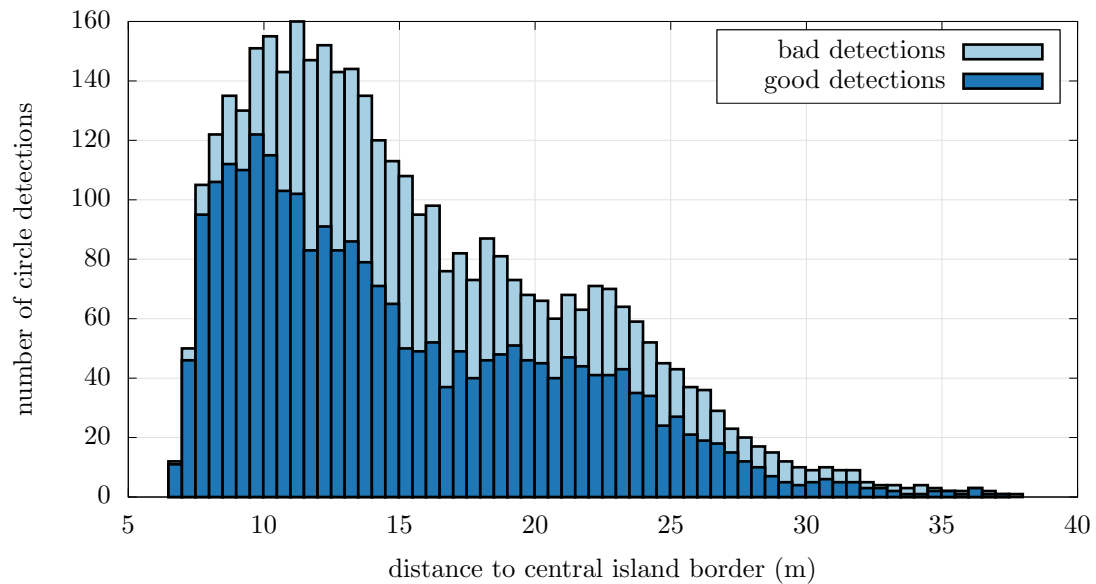


Figure 4.18: This figure shows for each distance bin the number of detected circles, where the bars representing the number of good detections and bad detections are stacked on top of each other.

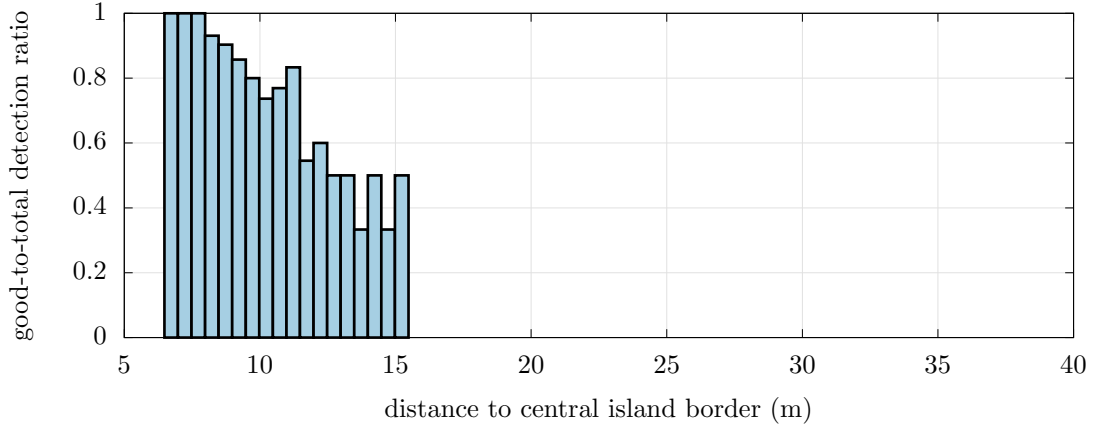


Figure 4.19: This graph shows for each distance bin the good-to-total detection ratio of laser scanner layer 0.

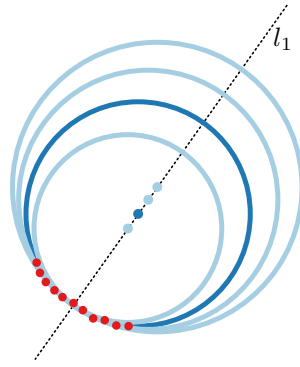


Figure 4.20: The dark blue circle is the circle from which the measured points (red) originate. Let K be the set of circles (the bright blue circles are a few of them) that fit to the set of measured points given a maximum fit error. This figure shows that while the maximum fit error can be relatively small, the radius error of a circle in K can be relatively large. Line l_1 is the line orthogonal to the point in the middle of the arc that is covered by the measured points. The circle centres of K vary mainly along this line.

algorithm fits a circle to a set of points that, instead of being distributed across the entire circle, is only distributed across a certain arc of the circle (which spans less than half of the circle). Consequently, as illustrated in figure 4.20, the set of circles that fit the set of points given a maximum fitting error may show an enormous variation in their radii compared to the maximum fitting error. Let us assume that this is a valid explanation for the large radius errors. Then the fitted circle will be near the ground truth circle near the measured points. Since an error in the estimated radius makes the estimated centre move away from the measured points by the same amount from the ground truth centre (see figure 4.20), one would expect the centre error and radius error to be strongly correlated. This expected correlation between the radius error and centre error is confirmed by figure 4.23.

The question can be asked whether the xy -coordinate system is a good coordinate system for analysing the centre error, since the direction of the centre error largely depends on the orientation of the arc covered by the set of points the circle is fitted to. Let us assume that if the arc is split in two equal size parts, the distributions of the points for these two parts are approximately symmetric. Then, as illustrated in figure 4.20, the major part of the fitted circle's centre error (of which the magnitude is approximately the radius error) will be along the line l_1 orthogonal

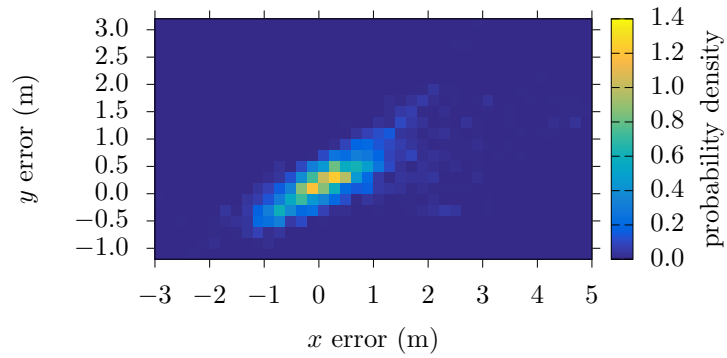


Figure 4.21: The error distribution of the centre of circles classified as good detections.

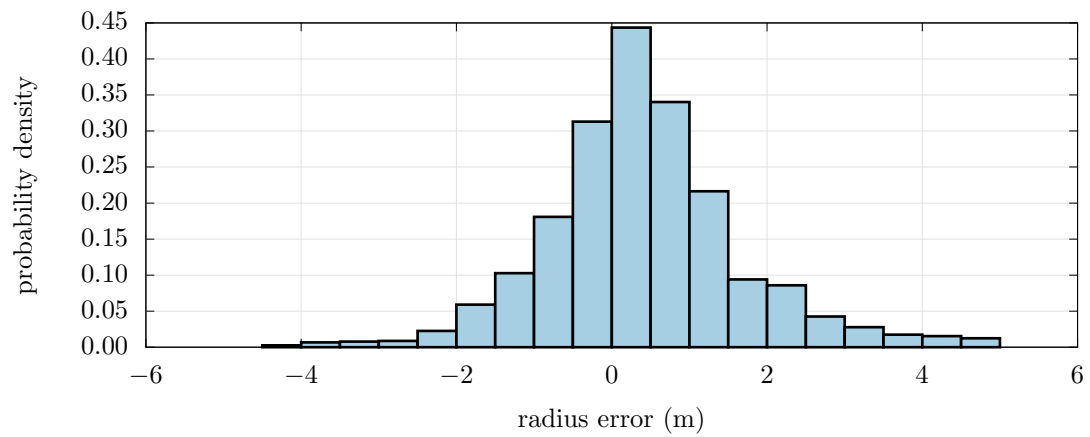


Figure 4.22: The error distribution of the radius of circles classified as good detections.

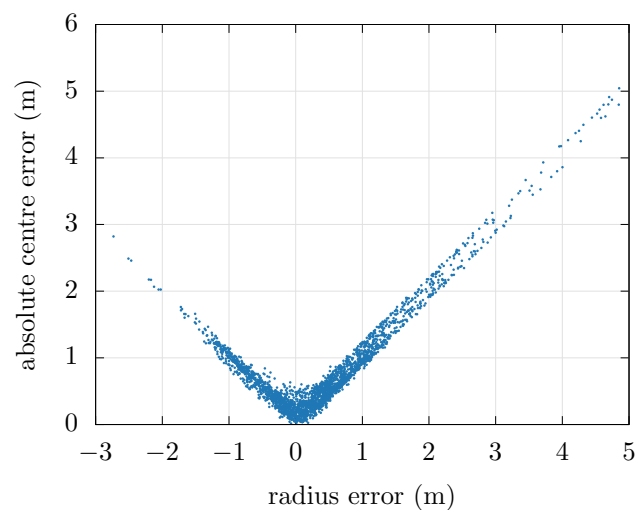


Figure 4.23: This graph shows for each good circle detection the absolute centre error (i.e. the distance between the detected circle centre and ground truth centre) and radius error.

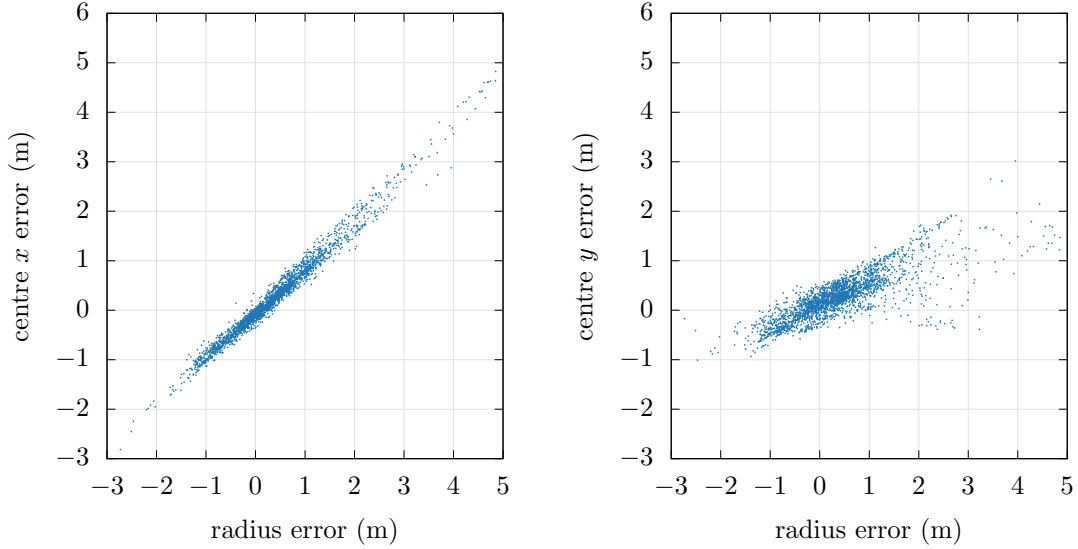


Figure 4.24: For each good circle detection in the left and right graph the radius error is plotted against the centre error in x -direction and the centre error in the y -direction respectively.

to the arc at point in the middle. It therefore seems a good idea to express the centre error in a coordinate system where one of the axes is parallel to l_1 . We expect the point in the middle of the observed arc to be close to the line l_2 from the sensor's origin to the centre of the ground truth circle. Thus l_2 will have approximately the same direction as l_1 . We define the $x'y'$ -coordinate system to be a right handed Cartesian coordinate system with axes x' and y' , where the x' -axis is in the same direction as the vector from the sensor's origin to the centre of the ground truth circle.

By comparing figures 4.24 and 4.25 we see that the centre error in x' -direction is more correlated with the radius error than the centre error in the x -direction is with the radius error. At the same time the centre error in the y' -direction is less correlated with the radius error than the centre error in the y -direction is with the radius error. Figure 4.26 hints that the centre error and radius error may be decorrelated by subtracting the radius error from the centre error's x' component. It also show that if the radius error is subtracted from the centre error's x component, that there is still a correlation with the radius error. From the above it can be concluded that the $x'y'$ -coordinate system is better than the xy -coordinate system in the sense that in the $x'y'$ -coordinate system the dependence on the radius error is isolated to only one of the components of the centre error.

Figure 4.27 shows the centre error distribution in the $x'y'$ -coordinate system and figure 4.29 shows the distributions in the x' and y' direction separately. When we compare the distribution for the centre error in the $x'y'$ -coordinate system with its distribution in the xy -coordinate system (figure 4.21), then we see that the error's x' and y' components are less correlated than its x and y components. Figure 4.28 shows the centre error distribution when the radius error is subtracted from the centre error's x' component.

Since the $x'y'$ -coordinate system seems to be more suitable than the xy -coordinate system to express the circle centre error, this will be the coordinate system used for the circle centre error in the measurement model in the following chapters.

Apart from evaluating the parameters of the detected circle it is also interesting to have a look at the points in the laser scanner point sequence which correspond to the detected circle, that is, those points where the circle was fitted on. We define the error d of a point to be the distance from that point to the ground truth circle, where d is negative if the point is inside the circle and positive otherwise. Figure 4.30 shows the distribution for d for all points of circles that

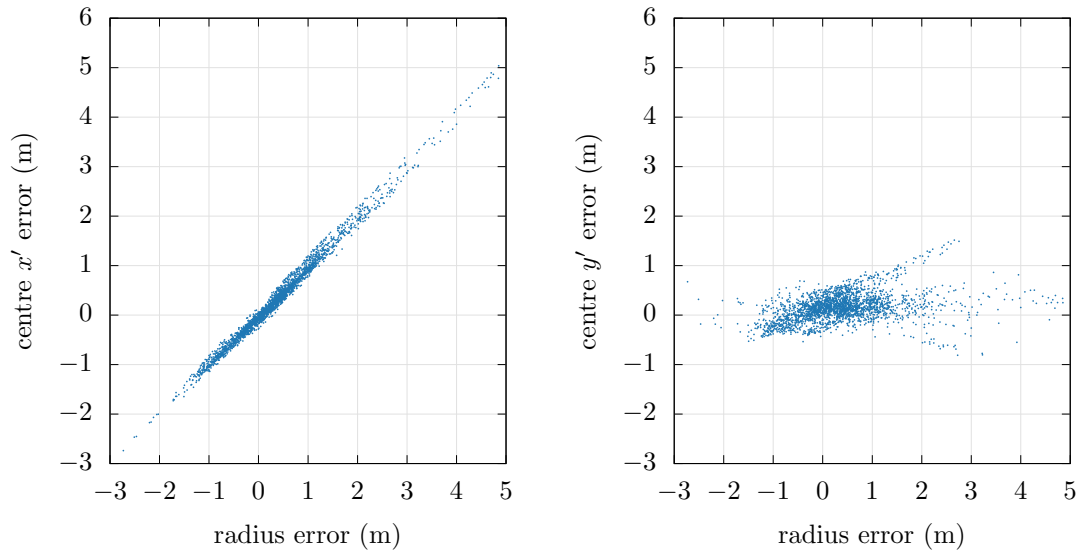


Figure 4.25: For each good circle detection in the left and right graph the radius error is plotted against the centre error in x' -direction and the centre error in the y' -direction respectively.

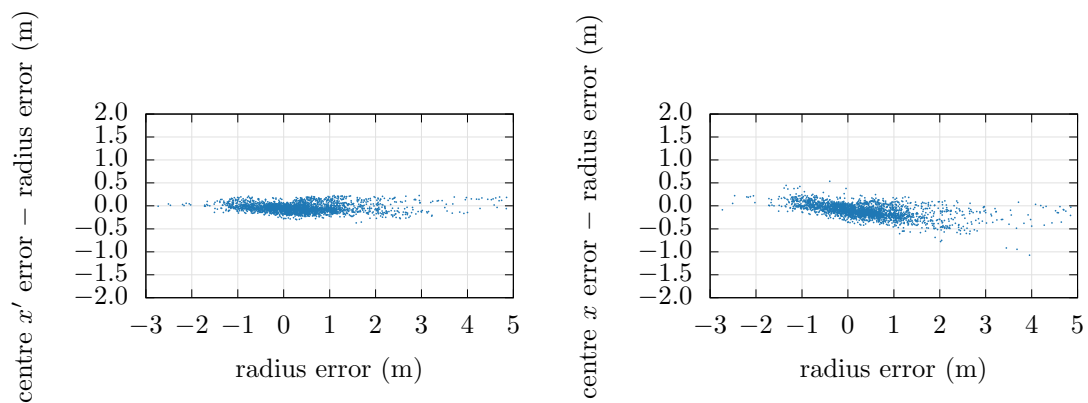


Figure 4.26: The left and right plot show a scatter plot for all the good circle detections where radius error is subtracted from the centre error's x' and x component respectively.

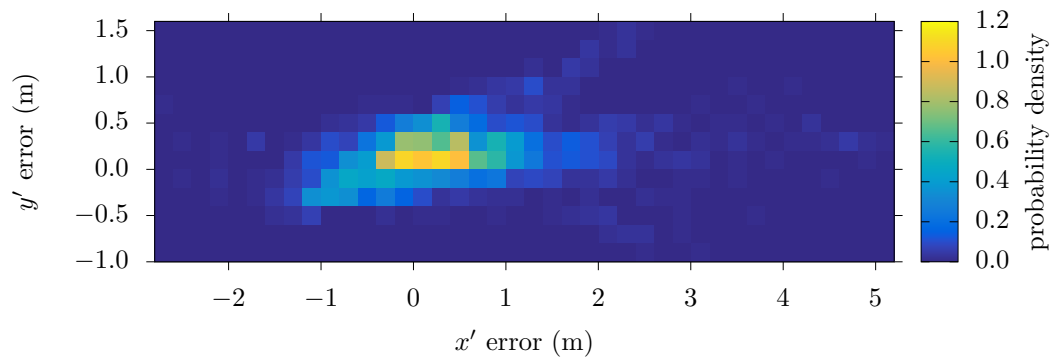


Figure 4.27: The error distribution in the $x'y'$ -coordinate system of the centre of circles classified as good detections.

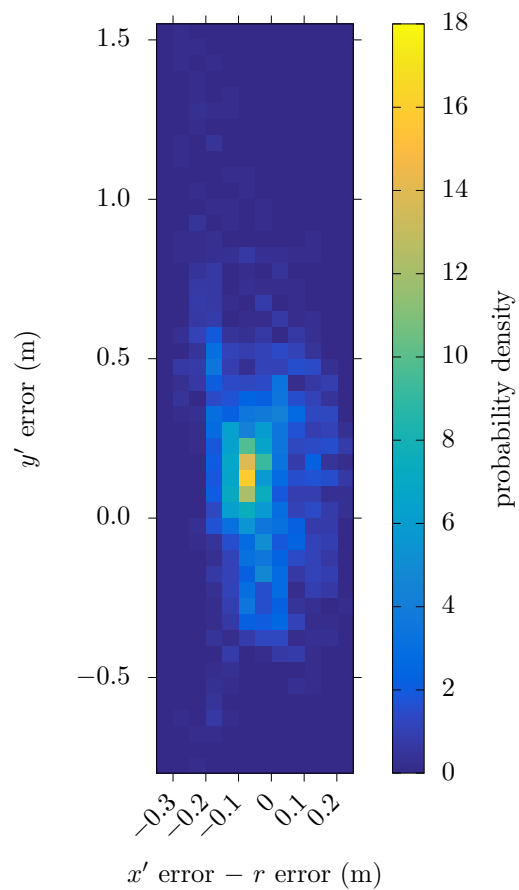


Figure 4.28: The error distribution in the $x'y'$ -coordinate system of the centre of circles classified as good detections, where the radius error is subtracted from the x' component of the centre error.

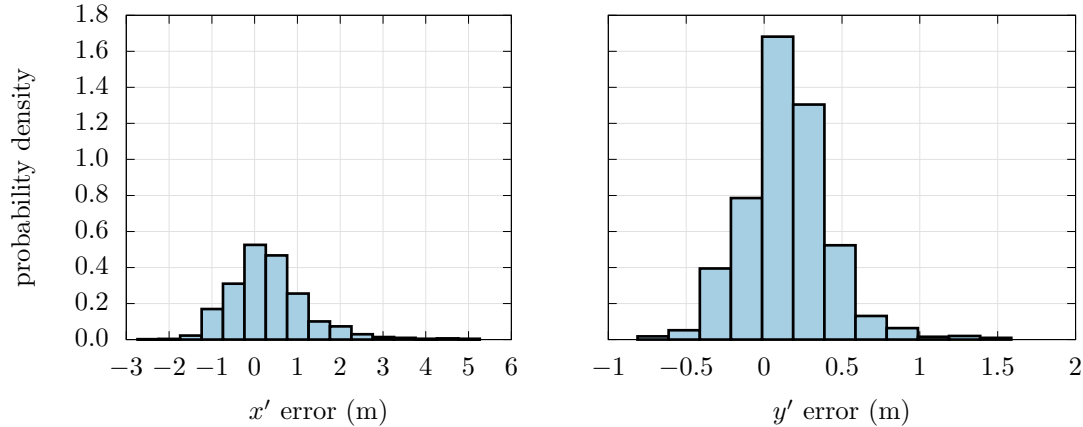


Figure 4.29: The error distribution for the x' component and y' component of the circle's centre error separately (of the circles classified as good detections).

are classified as good detections. It should be noted that this distribution, of course depends on our definition for a good detection, since we require a good detection to have at least 95% of the points with a d -value less than 0.3m. If we compare the centre error of the detected circle (see figure 4.27) to the distance of each point corresponding to the detected circle, then we see that the latter is relatively small. The distribution of d is used in chapter 6 in the measurement model for the roundabout map verification method.

4.7 Conclusions and Recommendations for Future Research

This chapter introduced a circle detection algorithm for the purpose of detecting the roundabout central island in a single layer of a laser scan. The computation times of an implementation of the algorithm on a PC, indicated that the algorithm could be used in a real-time system for online roundabout detection.

Besides evaluating the performance of the algorithm itself, also the quality of the input data was evaluated. This was done by investigating, based on ground truth data about the central island, whether the input point sequence contained a subsequence of points corresponding to the central island. Such a subsequence is called a benchmark sequence. At three out of the six evaluated roundabouts, the input data only contained a small number of benchmark sequences (i.e. at those roundabouts the laser only measured the central island a few times). At the other three roundabouts the number of benchmark sequences was high enough to consider the input data to be good for the detection of central islands. We do not know why exactly the laser scanner hardly measures the central islands of certain roundabouts, and suggest this as a topic for further research. Furthermore the results of older experiments, with a laser scanner that differed in its optical characteristics, showed better performance of the laser scanner measuring central islands. This indicates that the evaluation results presented in this chapter could even be improved if the laser scanner is optimized for measuring curbs of central islands.

An evaluation of the algorithm showed that out of the 2360 benchmark sequences in the input, only 50 of them were not detected by the algorithm, which means that the probability of a false negative detection is very low. This means that in 98% of the cases where the laser scanner measured a central island, the algorithm successfully detected the central island. The causes for those 50 undetected benchmark sequences were further investigated. The fact that the formal specification of benchmark sequences is weaker than the specification of circles that the algorithm must detect, seemed to be the main cause for these undetected benchmark sequences

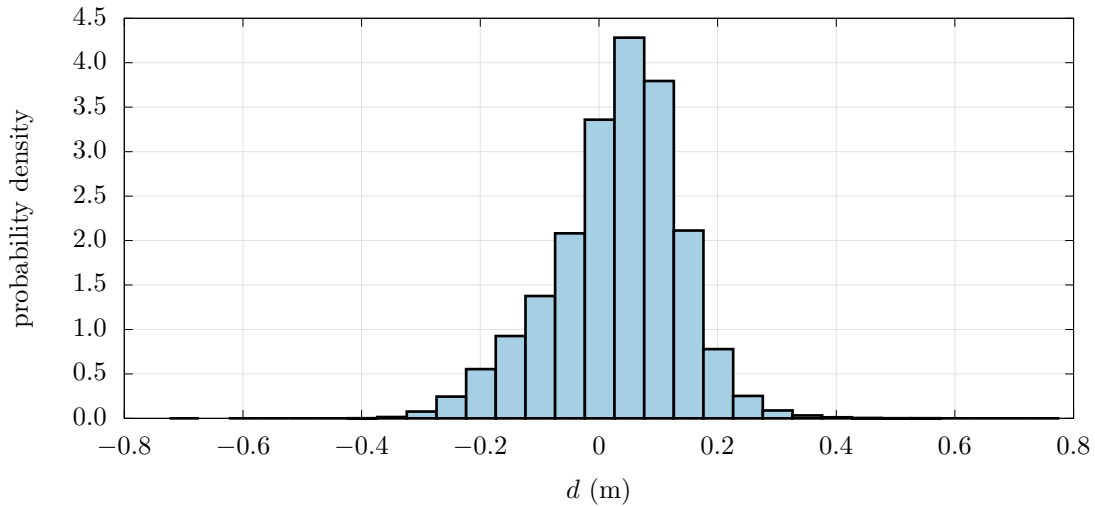


Figure 4.30: The distribution of d , the distance between a point and the ground truth circle, for all point corresponding to good detections.

While the input data contained 2360 benchmark sequences in total, 3039 circles were detected by the algorithm. This means that 78% of the detected circles did correspond to a central island. It should be noted however that, although not every detected circle corresponds to a central island, every detected circle is still a correctly detected circle in the input data. The good-to-total detection rate (a detection corresponding to a central island is called a good detection) was also found to depend on the layer of the laser scanner and the vehicle’s distance to the border of the central island. This was further analysed and appendix B shows the results for each layer.

Future research may focus on the reduction of the number of central island false detections. One suggestion on how this can be done is to verify a detected central island using a camera. Since a detection consists of a sequence of points with a three-dimensional position, the positions of these points can be projected to image coordinates. The challenge is then to develop an image processing algorithm that can verify whether the region around the image coordinates of the detection indeed looks like the curb of a central island.

For detected circles corresponding to the central island, the error of its parameters—the radius and planar coordinates of the centre—were also analysed. Since only a fraction of the central island’s border is measured by the laser scanner, the estimated radius of the detected circle can have an error of several metres. The error of the estimated centre showed to be strongly correlated with the radius error. In order to better model the error distribution of detected circles, a coordinate system was introduced in which the correlation of the centre’s error with the radius error is mainly restricted to one of the axes.

Although the error of a detected circle’s parameters may be several metres, the distance from the measured individual points, corresponding to such a detected circle, to the ground truth of the central island, is at most several decimetres. This means that the point sequence corresponding to a detected circle gives a quite accurate description of the central island’s geometry at those positions where it was measured, compared to the error of the detected circle’s parameters as a description of the complete central island geometry.

From the above it can be concluded that the central-island detector shows very promising results for use in sensor-based roundabout perception. Chapter 5 presents a roundabout perception system that makes use of the central-island detector presented in this chapter.

Chapter 5

Roundabout Perception Supported by A Priori Map Data

5.1 Introduction

In chapter 4 a method was presented to detect roundabout central islands in single-layer laser scans. While the estimation of the central island's geometry by this method can be relatively precise at those parts of the central island that were covered by the laser scan, the other parts of the central island geometry may show a large error because of the uncertainty of the estimated radius. Furthermore, a roundabout perception system for highly automated driving should be able to perceive all relevant elements of the roundabout as described by the roundabout model in section 2.2. Since current state of the art in sensor-based perception does not offer us methods to perceive all parameters of this roundabout model relative to the ego vehicle, this chapter proposes a roundabout perception method that combines sensor-based observations with a detailed a priori roundabout geometry from a digital map. Only some of the model's parameters are estimated using on-board sensor observations. The complete roundabout geometry is computed by aligning a detailed geometrical description from an a priori digital map with the estimated parameters. This a priori geometry describes with high precision the complete roundabout geometry except for its global position and orientation.

Similar to the intersection perception method of Nedevschi et al. [NPM⁺10], the approach presented in this chapter uses only the geometric roundabout description stored in the a priori map to localize itself relative to the corresponding roundabout. The advantages of such an approach, compared to other localization methods, are that no high accuracy global localization (like DGPS) is required and that the a priori map does not have to contain additional landmarks for localization.

A preliminary version of the results in this paper was published in the proceedings of the *2015 IEEE 18th International Conference on Intelligent Transportation Systems* [RB15]. The current chapter presents an improvement over the algorithm from the paper, based on an extensive evaluation of the central-island detector's performance in section 4.6.

Section 5.2 discusses the contents of the a priori digital map used by the roundabout perception algorithm. The algorithm itself is explained in section 5.3. Using ground-truth data, the performance of the algorithm is evaluated and compared with the old version of the algorithm in section 5.4. Final conclusions and recommendations for future research are given in section 5.5.

5.2 Digital Map Data

The presented system makes use of a priori map data which consist of a regular digital map normally used for navigation purposes, called the base map, that is extended with detailed data about roundabouts.

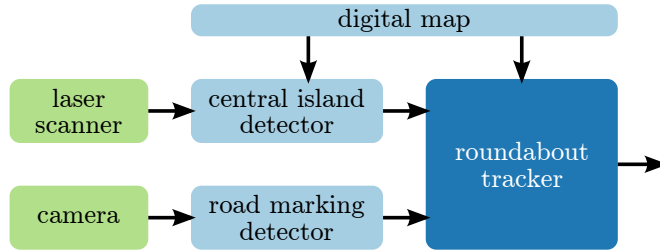


Figure 5.1: A schematic overview of the system’s components where the arrows represent data flow.

The base map can be seen as a graph of which the edges are road segments and the vertices are the points where the road segments are connected. We assume that for each edge in the map, only the centre-line geometry of the corresponding road segment is stored; information about the individual lanes of the road segment is not assumed to be available. A roundabout appears in this map as a cycle of edges. A flag indicates that these edges belong to a roundabout. The system does not require the base map’s geometry to have a high accuracy. In the implementation of the system a regular navigation map with a absolute centre-line accuracy of ± 7 m is used as the base map.

The base map is extended with a detailed description of the roundabouts in the map. This detailed description uses the model presented in section 2.2, but without describing the global position and orientation. That is, all geometric information is described relative to the centre of the roundabout and the position of the centre is not represented. For each of the road connections in the detailed roundabout description, the ID of the corresponding road in the base map is stored. The precision of the extended roundabout data is high enough to consider the data as exact information for our application.

Although the global position of the roundabout’s centre is not stored in the extended map data, a rough estimate of it can be extracted from the base map. So by using the ego-vehicle position and heading obtained via GPS we will be able to compute a rough estimate of the roundabout’s centre relative to the ego vehicle based on the base map data. The maximum error ε_b of this centre estimation, which is derived from the combination of map and GPS uncertainties, is assumed to be known. For the implementation of the presented system ε_b is set to 15 m.

5.3 The Roundabout Perception Algorithm

5.3.1 Algorithm Overview

The central-island detector presented in chapter 4 estimates the position of the roundabout’s centre relative to the ego vehicle. This algorithm was slightly modified such that in its last step it does not try to fit a circle with an unknown radius, but instead it tries to fit a circle with the exact radius r_c that is a priori known from the digital map. If we combine the centre measurements, generated by the central-island detector, with our a priori knowledge from the digital map about the radii of the central island r_c , truck apron r_t and outer border r_o , then we are able to generate a geometric description of the central island, truck apron and circulatory roadway relative to the ego vehicle. However in order to compute the position of the gates relative to the ego vehicle, only knowing the roundabout’s centre is not sufficient. For that we also need to know the orientation of the roundabout relative to the ego vehicle. To compute the orientation, we try to observe one of the endpoints of a gate. A geometric description of the roundabout can then be computed by first translating the detailed geometry from the digital map such that the centre coincides with the measured centre and then rotate the geometry around its centre such that the measured gate endpoint coincides with the corresponding gate endpoint (of the road from which the ego vehicle is approaching the roundabout) in the map.

Since there is no known method to directly measure the position of a gate, an endpoint of a

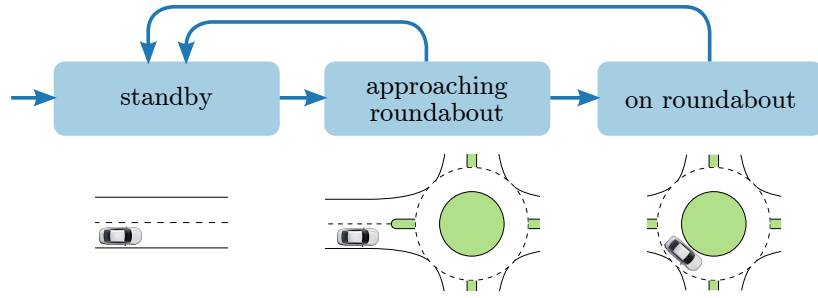


Figure 5.2: The algorithm’s operation stages, the transitions between them and for each stage corresponding situation the ego vehicle is assumed to be in.

gate is not directly observable. The two endpoints of a gate are defined as the points where the boundary of the lane (which is incident to the roundabout) is connected to the outer border of the roundabout. In this chapter we assume right-hand traffic and we assume that the left boundary of each entry lane is marked with a road marking. The left road marking of the lane, via which the ego vehicle approaches the roundabout, is measured in images made by a front-facing camera. If the roundabout’s outer radius (which is known from the a priori map) and the centre are known, then the gate’s endpoint can be derived by computing the intersection of the road marking and the roundabout’s outer border.

In case the left boundary of the entry lane is not a road marking but another type of boundary, the road marking detector can be replaced by a detector for that boundary type. The output of such a detector will then be used in the same way as the output of the road marking detector as described in this chapter. For example, for lanes only bounded by the curb of the splitter island, the laser-scanner-based road curb detector presented by Kellner et al. [KHB⁺14] can be used instead of a road marking detector.

The position and orientation of the roundabout are estimated by a roundabout tracker that is based on recursive state estimation. This tracker is discussed in subsection 5.3.3. How a hypothesis for the roundabout’s geometry is derived from this tracker is explained in subsection 5.3.4.

A schematic overview of the system’s components is provided by figure 5.1.

5.3.2 Stages of Operation

The roundabout perception system has three different stages of operation (as illustrated in figure 5.2): *standby*, *approaching roundabout*, *on roundabout*.

In the *standby* stage the system only performs a map matching on the base map using the location provided by a GPS module with dead reckoning. After the map matching is done the system checks whether the ego vehicle is approaching a roundabout in the base map. In case the ego vehicle is approaching a roundabout, the tracker will be initialized using the roundabout’s centre (relative to the ego vehicle) extracted from the base map. The system then goes into the *approaching-roundabout* stage and remembers the ID of the road via which the ego vehicle is approaching the roundabout.

In the *approaching-roundabout* stage the roundabout tracker is active and updates the roundabout’s state estimate using measurements from the central-island detector and road marking detector. When, given all observations, the probability that the vehicle is on the roundabout exceeds a specified threshold p_{on} , a transition will be made to the *on-roundabout* stage. Section 5.3.5 discusses how this probability can be computed.

If the system is in the *on-roundabout* stage, the left road marking of the lane via which the vehicle approached the roundabout is assumed to be not observable. Because the central island is no longer in front of the vehicle on the roundabout, the laser can no longer detect the central island. Therefore the tracker stops updating the roundabout’s state estimate using measurements during this stage. Instead it will only update the current estimate based on ego motion data.

In both the *approaching-roundabout* and *on-roundabout* stages the system will return to the *standby* stage if the probability that the ego vehicle is on the roundabout is below a specified threshold p_{out} and the vehicle is map matched to a road on which it is travelling away from the roundabout.

5.3.3 The Roundabout Tracker

The roundabout tracker consists of two completely decoupled (Bayesian) trackers: one tracker for the centre of the roundabout and one tracker for the left road marking of the entry lane via which the ego vehicle approaches the roundabout. Both the centre and road marking are tracked relative to the ego vehicle. The final roundabout estimate is computed by combining the information provided by both trackers as described in section 5.3.4. The road marking is tracked by the algorithm presented by Neumaier [Neu13] and Baer [BHG10]. This algorithm uses a multi-clothoid model for the road markings and tracks them using a Kalman filter. The tracker for the centre of the roundabout is updated by measurements of the central-island detector and is described in the rest of this sub-section.

The tracker for the roundabout's centre is implemented as a particle filter. Let M and E be the random variables for the set of central island measurements and the set of ego motion data respectively, received by the system thus far. Ego motion data consists of measurements by wheel speed sensors and accelerometers. Let m and e be the actual received values of the set of central island measurements and the set of ego motion data respectively (i.e. those are realisations of M and E). Furthermore, let \vec{S} be the random variable for the state we wish to estimate, that is the x and y -coordinate of the roundabout's centre. Generally a particle filter uses a set of weighted particles to represent a probability density distribution for the state it wishes to estimate given all information received thus far. In our application, this means that the particle filter will represent the probability density distribution for \vec{S} given $M = m$ and $E = e$. Each particle corresponds to a realisation of \vec{S} and has a weight. The probability density¹ $p(\vec{s} | m, e)$ for a specific realisation \vec{s} of \vec{S} can be derived from the set of particles by computing the weighted particle density around value \vec{s} . From here on, it is assumed the reader is familiar with particle filters and the rest of this sub-section discusses how the concept of a particle filter is applied to our problem of estimating the centre of a roundabout. The book by Thrun et al. [TBF05] provides a good introduction to particle filters.

When the filter is initialized (during the transition from stage *standby* to *approaching roundabout*) the particles will have equal weight and will be uniformly distributed within the circle of which the centre is equal to the rough roundabout centre estimate from the base map and of which the radius is equal to ε_b .

Let \vec{z} be the random variable for a new measurement. After receiving a new measurement \vec{z} , a particle filter updates the weight of each particle by using the measurement probability density $p(\vec{z} | \vec{S} = \vec{s}_p)$, where \vec{s}_p is the state represented by the particle. In our application the measurement is the centre's x and y -coordinate of the circle detected by the central-island detector.

Since it is possible that the central-island detector detects a circle in the laser scanner data that does not originate from the central island, the roundabout tracker takes into account the possibility of such a false positive detection. This is done by introducing random Boolean variable B , that indicates whether the measured circle corresponds to the central island. By applying the law of total probability, the measurement probability density can be written as:

$$p(\vec{z} | \vec{S} = \vec{s}_p) = p(\vec{z} | \vec{S} = \vec{s}_p, B = \text{true}) \cdot p(B = \text{true} | \vec{S} = \vec{s}_p) + p(\vec{z} | \vec{S} = \vec{s}_p, B = \text{false}) \cdot p(B = \text{false} | \vec{S} = \vec{s}_p) \quad (5.1)$$

Recall that for the evaluation of the central-island detector in subsection 4.6.4, we studied what fraction of the detected circles corresponded to the central island. The number of detections corresponding to the central island (i.e. the number of good detections) divided by the total

¹See section 2.4 for the notation adopted in this thesis.

number of detections, called the good-to-total detection ratio, could be used as the probability of correctly detecting the central island, $p(B = \text{true})$. As can be concluded from subsection 4.6.4, this good-to-total ratio varies over different intervals of the vehicle's distance to the central island's border and over the different scan layers of the laser scanner. This resulted in the histograms of appendix B, where for each laser scanner layer the good-to-total ratio is given as a function of the vehicle's distance to the central island's border.

Since in $p(B = \text{true} \mid \vec{S} = \vec{s}_p)$ the parameters \vec{s}_p of the central island are given and its radius is known from the map, the distance of the vehicle to the border of the central island is known. Furthermore the system also knows in which of the laser scan layers the circle was detected. Therefore, in order to obtain a value for the probability $p(B = \text{true} \mid \vec{S} = \vec{s}_p)$, the roundabout tracker looks up the good-to-total detection ratio for the corresponding distance to the central island's border in the histogram of the corresponding layer. Note that, although the histograms in appendix B are based on the output data of the central-island detector where it detects central islands with an unknown radius, data based on the central-island detector detecting central islands with a known radius resulted in similar histograms and therefore only the former histograms are included in this thesis and are used in the experiments.

Let us turn to $p(\vec{z} \mid \vec{S} = \vec{s}_p, B = \text{true})$, which is the probability density of measurement value \vec{z} given the parameters of the central island \vec{s}_p and given the fact that the measurement originates from the central island. In case the measurement originates from the central island with parameters \vec{s}_p , the following measurement model is used:

$$\vec{z} = \vec{s}_p + \vec{E}_z \quad (5.2)$$

where \vec{E}_z is a random variable describing the measurement error for the circle centre. Note that \vec{E}_z is not just a number that specifies the distance between the measured centre and the detected centre, but it is a vector. The reason for this is that the error is not uniform in all directions. In section 4.6.5 we concluded that probability density distribution of the circle centre's measurement error can be best described in the $x'y'$ coordinate system (that is, this coordinate system is preferred over xy -coordinate system). Since the measurement \vec{z} , state \vec{s}_p and error \vec{E}_z in equation 5.2 are described in xy -coordinates, they need to be transformed to $x'y'$ coordinates. Note that this transformation depends on the value of \vec{s}_p , since the $x'y'$ coordinate system is defined to be a right-handed coordinate system of which the direction of the x' -axis is equal to the direction of \vec{s}_p . Let z' , s'_p and \vec{E}'_z be the values in the $x'y'$ coordinates of \vec{z} , \vec{s}_p and \vec{E}_z respectively. Then the measurement probability density is given by:

$$p(\vec{z} \mid \vec{S} = \vec{s}_p, B = \text{true}) = p(\vec{E}'_z = z' - s'_p \mid B = \text{true})$$

The probability density distribution for \vec{E}'_z (given $B = \text{true}$) is described by figure 5.3 which is the result of performing the experiment of section 4.6.5 for the central-island detector configured to detect central islands with a known radius. The distributions for the x' and y' component of \vec{E}'_z separately, are presented by figure 5.4.

Let us now consider $p(\vec{z} \mid \vec{S} = \vec{s}_p, B = \text{false})$ in equation 5.1 which is the probability density measurement value \vec{z} given the parameters of the central island \vec{s}_p and given the fact that the measurement does not originate from the central island. The central-island detector will discard all detected circles for which the centre deviates more than the map uncertainty ε_b from the centre known from the digital map. This means that the space of possible centre measurements is a disk with radius ε_b . If $B = \text{false}$ (i.e. the detector produced a measurement not corresponding to the central island), then we assume that each centre measurement within the measurement space is equally likely. That is:

$$p(\vec{z} \mid \vec{S} = \vec{s}_p, B = \text{false}) = 1/(\pi\varepsilon_b^2)$$

Now equation 5.1 can be rewritten in a form where all required probability (density) values

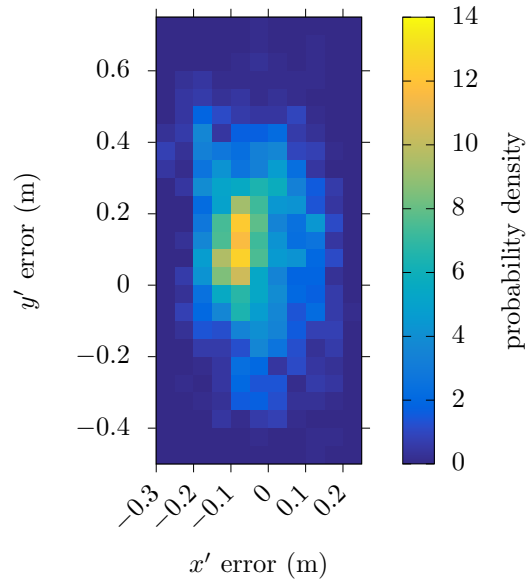


Figure 5.3: The error distribution in the $x'y'$ -coordinate system of the centre of circles classified as good detections, for the detector that detects circles with a known radius.

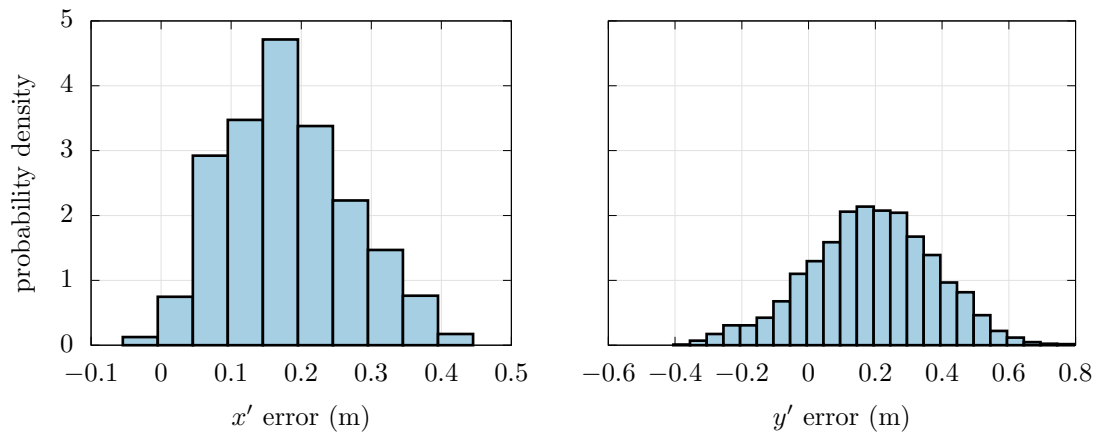


Figure 5.4: The distribution for the x' component and y' component of the circle's centre error separately (of the circles classified as good detections), for the detector that detects circles with a known radius.

are explicitly known:

$$p(\vec{z} \mid \vec{S} = \vec{s}_p) = p(\vec{E}'_z = \vec{z}' - \vec{s}'_p \mid B = \text{true}) \cdot p(B = \text{true} \mid \vec{S} = \vec{s}_p) + \frac{1 - p(B = \text{true} \mid \vec{S} = \vec{s}_p)}{\pi \varepsilon_b^2} \quad (5.3)$$

where \vec{z}' and \vec{s}'_p are the values in the $x'y'$ coordinate system of measurement \vec{z} and state \vec{s}_p respectively.

For state prediction, the particle filter uses information about the ego vehicle's motion based on measurements by wheel speed sensors and accelerometers. The ego motion is modelled as a translation and a difference in yaw angle, having a zero-mean normally distributed error of which the variances are proportional to velocity and yaw rate respectively.

5.3.4 Roundabout State Hypothesis Derivation

After a number of measurement updates the tracker's probability density distribution of the roundabout's centre will converge towards a single position. This convergence has been confirmed by experimental evidence. Consequently a single peak will show up in the distribution that can be interpreted as a single hypothesis for the position of the centre with an uncertainty.

The estimate of the road marking is only an estimate for a certain part of the marking. Only if this estimated part includes the gate endpoint, the road marking estimate provides information about the roundabout's orientation. The system does not know in advance whether this part reaches until the gate endpoint. However, if we have a good estimate of the roundabout centre then we are able to check whether the road marking estimate reaches the border estimate of the roundabout. If it does reach the border then the road marking estimate can be used to estimate the orientation of the roundabout.

Let μ_c be the expectation of the centre distribution (i.e. the distribution's first moment) and ε_μ be a parameter specifying the maximal acceptable deviation of μ_c to the true roundabout centre c . Then we say that μ_c is a good estimate of the roundabout centre if the probability that $|\mu_c - c| \leq \varepsilon_\mu$ is above a specified threshold $p_{c,\text{good}}$. This probability can be computed in a way similar to the computation of the probability that the ego vehicle is on the roundabout as explained in section 5.3.5. In case μ_c is a good estimate for the centre, it will be used as the centre estimate \hat{c} of the output roundabout hypothesis, where the variance is set to the distribution's variance σ_c^2 (i.e. the distribution's second moment).

Given the estimate of the centre, the roundabout's outer radius r_o (known from the digital map) and an estimate of the road marking provided by the road marking tracker, the system computes the point p_i where the road marking intersects the outer border of the roundabout. Recall that when going from the *standby* to the *approaching-roundabout* stage, ID (in the map) of the road via which the ego vehicle is approaching the roundabout was stored. The gate description corresponding to the entry lane of the road with this ID is extracted from the extended map data. In this gate description the position p_m of its left endpoint is described in the local coordinate system of the extended map data.

The position and orientation of the roundabout data extracted from the digital map are aligned in such way that the centre coincides with the estimate of the centre \hat{c} and p_m coincides with the estimate p_i of the left endpoint of the gate. The aligned roundabout model is then the estimate of the roundabout relative the ego vehicle and will be the system's output.

5.3.5 Computing the In-Roundabout Probability

As explained in section 5.3.2, the decision to make a transition to the *on-roundabout* stage or the *standby* stage is based on the probability that the ego vehicle is inside the roundabout. This subsection shows how this probability can be computed from the roundabout's centre probability density distribution maintained by the tracker.

Let random Boolean variable A indicate that the ego vehicle is on the roundabout and let m and e be respectively all measurements and ego motion data until the current time instant. Then $p(A = \text{true} \mid m, e)$, which is the probability that the ego vehicle is on the roundabout given m and e , is used to decide whether to make the transition to the *on-roundabout* stage or *standby* stage.

For simplicity we say that the ego vehicle is on the roundabout if the centre of its rear axle, which is the origin of the ego vehicle's coordinate system, is inside the outer-border circle. This means that if the radius r_o of the outer border and the centre relative to the ego vehicle are known, it can be checked whether the vehicle is on the roundabout. Radius r_o is known a priori and a probability density distribution for the centre is maintained by the centre tracker. Since $A = \text{true}$ is equivalent to $|\vec{S}| \leq r_o$ (meaning that the origin is inside the circle with centre \vec{S} and radius r_o), it follows that:

$$\begin{aligned} p(A = \text{true} \mid m, e) &= \Pr(|\vec{S}| \leq r_o \mid m, e) \\ &= \int_T p(\vec{s} \mid m, e) d\vec{s} \end{aligned} \quad (5.4)$$

where $T = \{\vec{s}' : |\vec{s}'| \leq r_o\}$ is the space of centre states for which the origin is inside the roundabout's outer border. Here $p(\vec{s} \mid m, e)$ is the probability density for the roundabout's centre \vec{s} given all measurements m and ego-motion data e until the current time instant. This conditional probability density distribution for the centre is maintained by the roundabout centre tracker which is a particle filter representing the distribution by a particle set Q . As a result, the integral in equation 5.4 can be approximated by summing the normalized weights of all particles of which the distance to the origin is at most r_o :

$$p(A = \text{true} \mid m, e) = W^{-1} \sum_{p \in \{p' \in Q : |\vec{s}_{p'}| \leq r_o\}} w_p \quad (5.5)$$

where \vec{s}_p is the state (i.e. roundabout centre) corresponding to a particle p , w_p is the weight of a particle p and W is the total weight of all particles in Q .

5.4 Evaluation

This section evaluates the presented roundabout perception method. As indicated in a previous publication [RB15] about this perception method, the error made by the road marking detector directly translates to the error of the estimated roundabout orientation in the output of the complete system. Since the development of a road marking detector is outside the scope of this work we omit the analysis of the roundabout orientation estimation. A detailed analysis of the performance of the central-island detector can be found in section 4.6. Consequently, the focus of this section is on the estimation of the roundabout's centre.

Furthermore the performances of the tracker as presented in the previous publication [RB15] (referred to as the old tracker) is compared with the performance with the tracker as presented in this thesis (referred to as the new tracker). The old tracker uses a zero-mean bivariate Gaussian for the centre error probability density distribution, whereas the new tracker uses a more generic way to represent the probability density distribution in form of a two-dimensional histogram. Additionally the new tracker uses a correct-measurement probability that depends on the distance to the central island's border and scan layer, while in the old tracker this is a constant probability.

In section 4.6, we found out that the central-island detector's input data does not for every roundabout or gate contain sufficient point sequences that are on the central island. Therefore, as in the evaluation of the central-island detector, only the following roundabouts were selected for this evaluation: Casino, Gaimersheim Large and Lana Grossa gates 1 and 2. At each of the gates of these roundabouts three new recordings were made under the same conditions (i.e. no occlusions by other vehicles and a velocity of approximately 20 km/h) as in section 4.6. This means that we have two sets of recordings: those of section 4.6 and those of this section. The recordings of section 4.6 were used to derive the required probability density distributions for the measurement model, and can therefore be called the training set. The additional recordings made

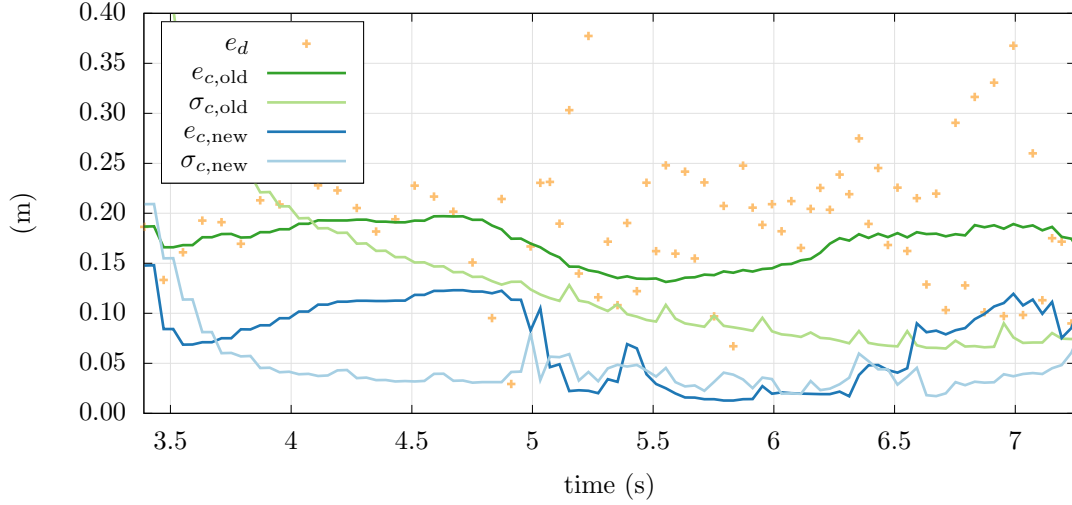


Figure 5.5: This graph shows for one of the recordings where gate 1 of the Gaimersheim Large roundabout is approached, the detector’s circle centre error e_d , the tracker’s centre error $e_{c,\text{old}}$ and $e_{c,\text{new}}$, and the standard deviation of the tracker’s estimation $\sigma_{c,\text{old}}$ and $\sigma_{c,\text{new}}$. Subscript old is used to indicate the results of the old tracker and subscript new to indicate the results of the new tracker.

for this section were used for evaluation of the roundabout perception system, and can therefore be called the test set.

Since the central-island detector used in this chapter detects circles with a known radius, unlike the version of the detector presented in chapter 4 where the radius is unknown, the probability density distribution for the detected centre error is different from the distribution shown in section 4.6. Figure 5.3 shows the probability density distribution for the central-island-with-known-radius detector. Since there is no significant difference in the number of total and successful detections between the detector used in this chapter and the one of chapter 4, the detection histograms for the number of detections of figures 4.16 and 4.18 as well as the probability histograms in appendix B, also hold for the detector used in this chapter. Hence, the probability histograms of appendix B are used for the correct-detection probability $p(B = \text{true} \mid \vec{s})$.

Based on the same method as in section 4.6, a ground truth (for the geometry of the roundabout relative to the vehicle) was computed for each time instant the laser scanner produced a measurement.

We use symbol e_d for the error of a detected circle, which is the distance from its centre to the centre of ground truth roundabout. For a specific instant of time, symbol e_c is used to denote the error of roundabout centre as estimated by the particle filter, which is the distance from the estimated centre to the centre of ground truth roundabout. Figure 5.5 shows for one of the recordings at gate 1 of the Gaimersheim Large roundabout, the performance of the detector and both old and new tracker. Besides the error e_c it also shows the standard deviation σ_c of tracker’s probability density distribution for the central island’s centre, which is computed by the algorithm itself. One can clearly see that for this recording the new tracker outperforms the old tracker.

Instead of having a plot like figure 5.5 for each recording, the tracker error of each recording is summarized by a box plot (which visualises the quartiles of the tracker error). The tracker error is computed each time the tracker performs an update, which is at each time instant where the laser scanner produced a measurement. The box plot of each single recording is based on the updates that were done after the vehicle’s rear axle reached the distance of 20 m to the border of the central island. That is, we evaluate the tracker in the case where the vehicle is at most 20 m away from the border of the central island.

Figure 5.6 shows the box plots for the recordings of the three roundabouts. This figure both

include box plots for the old tracker and the new tracker. In most cases the new tracker shows a better performance than the old tracker. However for gates 0 and 2 of the Casino roundabout, the new tracker performs worse than the old one.

An explanation for the bad performance at Casino gate 0, can be found in figure 5.7, which shows for Casino gate 0 the distribution of the central-island detector's error based on the training data set. When we compare this with the detectors error distribution over all roundabouts in figure 5.4, we see that weight of the error in y' direction is more shifted to the left. Figure 5.8 shows, that indeed the tracker performs better at Casino gate 0 if the tracker uses a centre error distribution specifically for that gate. Similar arguments hold for the bad performance at gate 1 of the Casino roundabout.

From the data from gate 0 of the Casino roundabout, it can be concluded that the error distribution of the detected circle centre based on the data for all roundabouts may not be a representative distribution for the the centre error at individual gates. In other words: the error of the detected circle centre seems to depend on the roundabout and gate the vehicle is approaching. If better results are desired, then the property of the roundabout which causes these distributions to be different should be identified. Once this source is known it can be incorporated into the sensors model which then can produce better probability density distributions.

An alternative to finding the source for the different distributions and constructing a more detailed sensor model, is to learn the detected centre error distribution for each individual roundabout gate. This may be, for example, done during the mapping process of the roundabout, where besides measuring the geometry of the roundabout, the behaviour of the laser scanner and central-island detector is also analysed. However such an approach may be very time consuming during the mapping process and has to be done again every time the sensor changes. It is also possible to construct a distribution for each individual gate using data from a car fleet, that is equipped with a laser scanner, the presented roundabout perception system and another method that localizes itself relative to the roundabout map. Both the localization of the roundabout perception system and second localization method may have a high uncertainty, but by fusing them a more accurate localization can be achieved. This allows us to afterwards estimate the error of each circle that was detected and by that an estimation for the error distribution can be made. Such a system could be a topic for future research.

As indicated in section 4.6, previous experiments with an older version of the laser scanner resulted in more benchmark sequences at the Lana Grossa roundabout. Not only the number of benchmark sequences was higher, but also the centre errors of detected circles were smaller than with the newer version of the laser scanner. The paper about the old algorithm [RB15] presents the evaluation results of the old tracker using this older laser scanner version. This evaluation was done at the Lana Grossa roundabout with exactly the same implementation and parametrisation of the old tracker and detector as in this chapter. Since the circles detected with the central-island detector, in combination with the older laser scanner, have a smaller error compared to the newer laser scanner, the evaluation in [RB15] of the tracker also showed a better performance than the evaluation results in this chapter. In [RB15] the error of the centre estimation at gates 1 and 2 of the Lana Grossa roundabout was most of the times below 0.30 m, whereas with the newer laser scanner the error at gate 1 is well above 0.30 m (even all medians are above 0.40 m).

Unfortunately the older version of the laser scanner was not available any more at the time the experiments for this chapter were done. However the few experiments that were done with the older laser scanner show that a better performance than presented in this chapter can be achieved.

5.5 Conclusions and Recommendations for Future Research

This chapter proposed a perception algorithm that estimates the parameters of a roundabout relative to the ego vehicle. The perception algorithm combines sensor-based observations of the central island and road markings of the entry lane with a geometric roundabout description known from an a priori digital map. The a priori geometry does not describe a global position or orientation and the system works with regular GPS data.

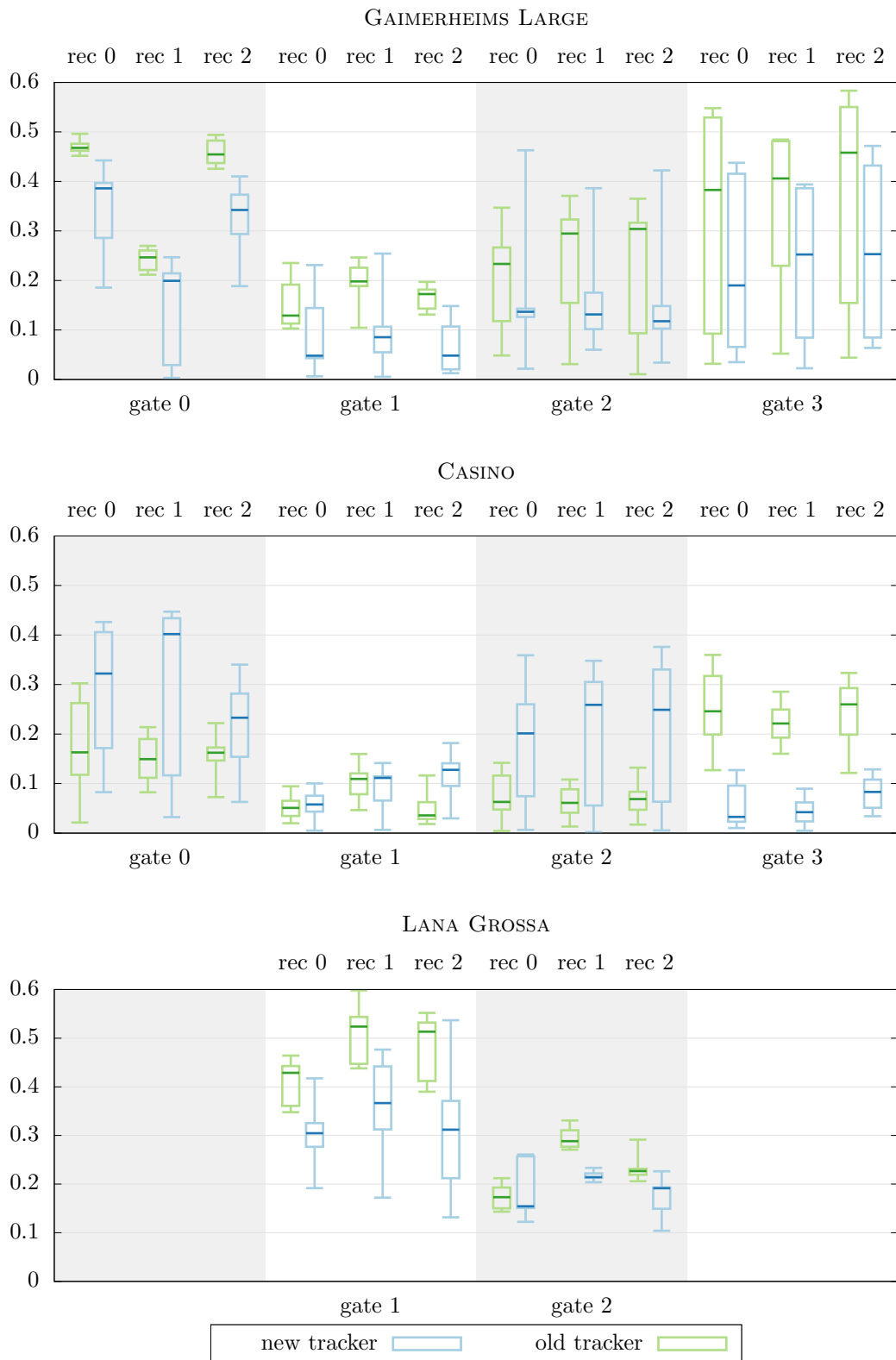


Figure 5.6: This figure shows for each recording a box plot for the error of the estimated roundabout centre, during the time the vehicle was within the 20m distance range of the central island's border.

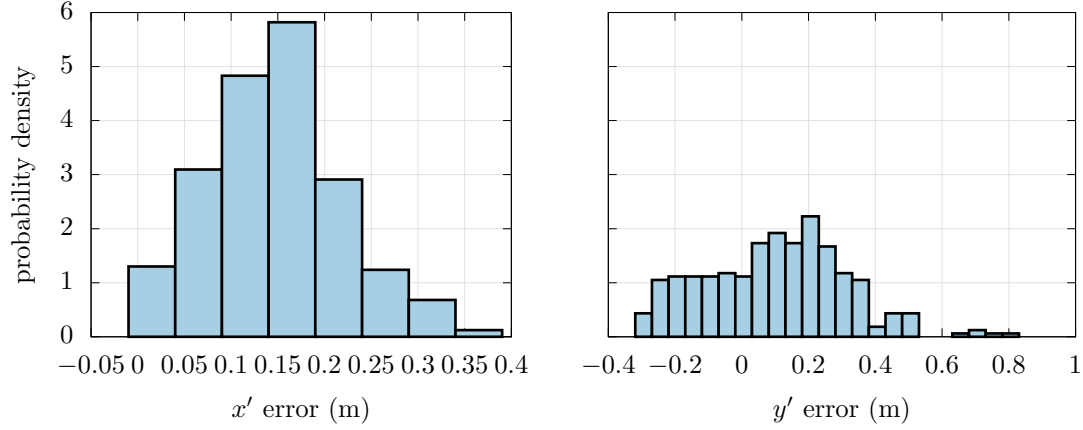


Figure 5.7: For gate 0 of the Casino roundabout, this graph shows the distribution for the x' component and y' component of the circle's centre error separately (of the circles classified as good detections), for the detector that detects circles with a known radius.

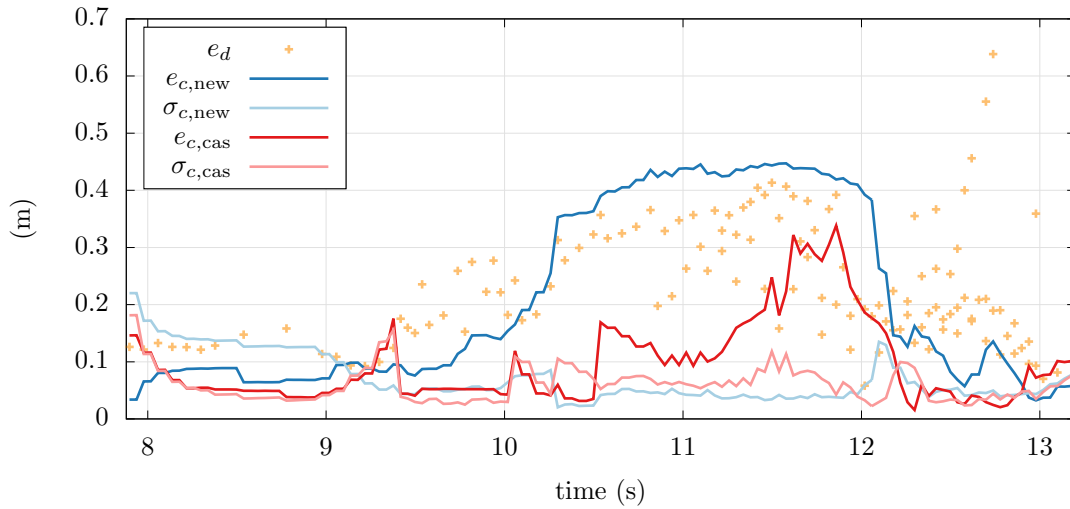


Figure 5.8: This graph shows for one of the recordings where gate 0 of the Casino roundabout is approached, the detector's circle centre error e_d , the tracker's centre error $e_{c,new}$ and $e_{c,cas}$, and the standard deviation of the tracker's estimation $\sigma_{c,new}$ and $\sigma_{c,cas}$. Subscript new is used to indicate the tracker results of the new tracker where the circle centre distribution was used that is based on statistics over all roundabouts. Subscript cas is used to indicate the results of new tracker with the circle centre distribution based on statistics for specifically gate 0 at the Casino roundabout.

The proposed algorithm (referred to as the “new algorithm”) is an evolution of the algorithm (referred to as the “old algorithm”) proposed in our initial paper [RB15]. One of the differences is that in the new algorithm the measurement model uses a generic error distribution for central island detections, instead of a zero-mean bivariate normal distribution. Using roundabout ground truth data, the performance of the old and new algorithm were compared in an evaluation at ten different entry gates of three different roundabouts. Overall the new algorithm showed an improvement over the old algorithm with the exception of two of the gates. At those two gates a worse performance is caused by the fact that the error distribution of the detected central island centre that is based on the statistics over the population of all ten roundabout gates, is not representative for error distributions at those two individual gates. Further experiments showed improved results if a distribution based on statistics for the individual gate was used in the measurement model. Collecting statistics for each individual roundabout gate where we want to drive can be a very time consuming task and can decrease the practicality of the proposed method compared to the case where one error distribution is representative for all roundabout gates. Alternatively, a better understanding of the laser scanner and which aspects at the roundabout have an influence on the measurement errors, might allow us to construct a more generic measurement model that is representative for all gates at all roundabouts.

For all test runs the median error, from the moment where the vehicle was 20 m removed from the central island while approaching it, of the estimated central island centre was at most 0.4 m. The experiments with the old algorithm published in the initial paper [RB15] were done with an older version of the laser scanner of which the optics were optimized to detect points on the road. These previous experiments showed a much smaller error of the estimated centre, compared to the experiment results for the old algorithm in this chapter. It is therefore very likely that the new algorithm will give much better results than the results presented in this chapter, if it is used in combination with a laser scanner that has been optimized for measuring points on the road. Overall it can be concluded that the presented roundabout perception method shows very promising results for use in automated driving.

Suggestions for future research are:

- Improving the design of the laser scanner such that it better measures the central island, like the old version of the laser scanner that was used in previous tests, and performing with this laser scanner new evaluations of the proposed roundabout perception method.
- Developing a complete automated driving system that integrates the proposed roundabout perception system. Closed-loop tests of such a system can be done to investigate the influence of the perception system on the overall system performance.
- Extending the proposed algorithm by integrating sensor observations of more environment elements like, for example, the road curbs of splitter islands or the marking of the outer border.
- Investigating the behaviour of the laser scanner at central islands. A better knowledge about the laser scanner can help us in identifying the major factors of a roundabout that influence the different characteristics of central island measurement errors among different roundabout gates. Such knowledge could then be incorporated into the measurement model to improve the system’s performance.

Chapter 6

Map Verification

6.1 Introduction

Future highly automated vehicles may require a priori digital maps for information about environment elements that cannot be perceived by purely sensor-based methods. In such cases the correctness of the map must be known to the system in order to achieve the high level of robustness that is required for highly automated driving. This chapter focuses on the verification of a priori digital maps for this purpose. This chapter should be seen as an initial exploration of the thus far unstudied field of map verification for automated driving. It does not present a ready-to-use solution, but identifies some important aspects of and requirements for map verification and investigates a possible way to model the verification problem.

Section 6.2 first discusses the major reasons that make map verification necessary, and how the outcome of a map verification could be used by an automated driving system. From this some major requirements on a map verification system are derived.

The rest of this chapter investigates a possible model for map verification. Section 6.3 gives a formal definition of the map verification problem. In section 6.4 a probabilistic map verification method is proposed based on the principles of recursive state estimation, that depends on a certain kind of conditional independence between the measurements. Although the assumption of this kind of conditional measurement independence is commonly made in the field of probabilistic recursive state estimation, it is not directly obvious whether in this case it is a valid assumption. This chapter shows that under this assumption, one can formulate a mathematically elegant method to verify the contents of a digital map, that seems rather intuitive. An implementation of the proposed model for the verification of mapped roundabout central islands, that uses the output of the central island detector of chapter 4, is presented in sections 6.5 and 6.6. Based on this implementation, the map verification model is analysed in section 6.7. This analysis shows that the assumption of conditionally independent measurements is invalid and is one among other reasons that lead to undesirable results.

6.2 Requirements on Map Verification

Let us first have a look at why a priori map data and its verification will be necessary for the perception systems of highly automated vehicles. As explained in chapter 1, a priori map data will be needed to obtain information about environment elements that cannot be fully perceived by sensor-based methods. There can be several reasons why an environment element cannot be perceived using sensor-based methods:

- An algorithm for its perception exists, but it is too computationally expensive for a real-time implementation in a (production) vehicle.
- An algorithm exists but its output is not accurate enough.

- The accuracy of the sensors is not sufficient.
- There exists no known algorithm at all.

It is possible that an environment element can be robustly perceived by a sensor-based method, but only under certain favourable conditions (e.g. the system only works during day time, when there are no hard shadows and it does not rain). If a perception system wants to use a priori map data for those elements under the conditions where they cannot be perceived by sensor-based methods, then also under those conditions the correctness of the map has to be verified.

For the reasons just mentioned, perceiving an environment element may not always be possible. In cases where perception is impossible, not sufficiently accurate, or too uncertain, the driving system needs additional data about the environment element. A digital map can provide this. However, the system should not assume that the map is always correct and fully accurate: the map could be outdated, and there could be temporary changes in the environment that are not reflected in the map. Moreover, localization in the map is also subject to uncertainty. Thus a robust highly automated driving system needs to deal with sensor data and a priori map data in an integrated manner: if possible, a relevant environment element is perceived purely based on sensor data, and otherwise the perception of the environment element will be based on map data of which the correctness is verified by sensor data.

When the system discovers that a part of the map is incorrect, then the area covered by this incorrect part will be a “blind spot” for the system if it can neither be perceived by sensors nor by the map. Appropriate action must be taken in case of such blind spots. It should be noted that the possibility of having unwanted blind spots in perception is not restricted to the use of a priori map data. For example, sensors can be broken and as a result deliver incorrect data about the environment. A highly automated driving system should also be able to cope with this kind of failure and therefore handling blind spots in perception is a problem that needs to be solved regardless of whether a priori maps are used or not. Plausible ways for the system to handle blind spots due to incorrect map data, are:

- Choose a driving behaviour in which the incorrect part of the map becomes less relevant. For example, if on a three-lane road the right most lane is incorrect in the map, but the other lanes are correct, then the system might plan a trajectory over the left most lane.
- If in a certain region in the map only a part of the map data is incorrect, use the data of the map part that is correct together with sensor measurements to perceive the part of the environment that corresponds to the incorrect map part. That is, a priori map data that is verified to be correct supports a sensor-based method to perceive the area of the environment corresponding to the incorrect part of the map.
- Choose a different driving strategy (e.g. drive slower or make a wider turn) that allows the perception system to perceive the element using sensors only. Certainly this option may work in only a few specific scenarios and might cause less favourable (e.g. less comfortable) behaviour for the passengers of the vehicle, but it may prevent the system from entering a fall-back mode.
- If no other options are left to guarantee safe driving behaviour, request the human driver to take over control of the vehicle (in case of a highly automated vehicle) or bring the vehicle into a safe state (in case of a fully automated vehicle), based on the map data that can be verified and based on sensor data.

It may also be possible that the map verification method cannot decide whether a part of the map is correct or not. It will be a safe choice to handle such a part of the map in the same way as if it was incorrect. If the area of which the correctness could not be verified is relatively small, then the system may assume that the map is correct and use it for behaviour planning as long as safety can be guaranteed based on sensor data. For example, consider the case where the correctness of the ego lane can be verified up to 20 m in front of the vehicle and for the range from

30 m to 40 m of the vehicle, while the correctness for the range from 20 m to 30 m is unknown. Then the system may plan its trajectory on the ego lane up to 40 m while assuming the map's description for the part from 20 m to 30 m is correct. However, the system should only do this if it can guarantee, based on sensor data, that it will travel over a smooth surface, the trajectory is free of static obstacles and it will not collide with other vehicles. While doing this, the possibility that the map is incorrect should still be considered for prediction of the behaviour of other traffic participants.

The range of possibilities to handle incorrect map data obviously depends on the level of detail in which the map verification method can make statements about the correctness of the map. Therefore a map verification method should be as specific as possible about the location of a map error. As a trivial example, it would not be desirable if the verification method would report the entire map to be incorrect when it concludes that some specific part of the environment is not correctly described by the map. Instead, it should explicitly report those environment elements that are falsely described by the map. This way the correct part of the map is still usable for the system. If the geometry of a single environment element is spread over a large distance then a verification system must be able to identify those parts of the geometry that are incorrect. A useful resolution at which parts of a geometry are separately verifiable must be chosen, such that it allows the behaviour planning system to generate safe and robust behaviour in case of a map error. Furthermore it can be dangerous to classify an environment element or a part of it as correct if only a part of it could be verified based on sensor data. For example, it may be dangerous to conclude that the road marking up to 50 m in front of the vehicle is correct while the sensors only have a visibility range up to 20 m. In such case it might be useful to only classify the correctness of the part of the road marking up to 20 m as correct and the rest of it as unknown, or to classify the correctness of the complete road marking as unknown in case the verification resolution is chosen to be more than 20 m.

The topic of localization should also be considered within the map verification problem. Not only is localization of the ego vehicle in the map necessary to describe the a priori map data relative to the vehicle, but it will also be crucial for verifying the map using sensor data. If the map-relative localization is wrong, then a verification method may falsely classify map data as incorrect. As explained in section 1.3.3 localization purely based on GNSS systems may not be sufficient for automated vehicles and therefore many localization techniques rely on landmarks in the map. Consequently the problems of map verification and map-relative localization cannot be treated separately, since map verification relies on localization and localization relies on the correctness of the map data.

From the above discussion the following requirements can be formulated for a map verification system:

- The verification method should not be based on the comparison of the map description of an environment element and the description of this environment element as the result of sensor-based perception (i.e. the method should not require sensor-based perception to perceive an environment element).
- The map description of an environment element or a part of it may only be classified as correct if the correctness of that complete element or its part can be verified based on sensor data.
- The verification method should be able to identify the environment elements that are incorrectly described by the map.
- In case the geometry of an environment element is spread over a large geographic distance, then the verification method must be able to identify errors in this geometry up to a certain resolution.
- In case the verification system cannot be certain about the correctness of an environment element description by the map, then it should express this uncertainty in its output.

- The map description of an environment element must be verifiable under all conditions (e.g. weather, velocity, day/night) where the automated driving system uses the map data for perception.
- The interdependence between the problems of map verification and map-relative localization must be taken into account.

The verification model presented in this chapter describes how a certain part of an environment element can be verified. In the description of this model, we assume for simplicity that environment elements are not subdivided in parts. Whether the above requirements are satisfied depends on the system in which the model is integrated. For example, whether correctness can be verified early enough does not depend on the verification model but on the used sensors, operating conditions of the system, behaviour planning algorithms, etcetera. Since the presented verification model is not intended to offer a complete solution, but more of an initial step towards a solution that we want to evaluate, the verification problem is simplified by assuming it is independent from the localization problem.

6.3 Problem Statement

Let us consider the verification of one environment element described by the a priori map, based on measurements on environment elements of the same type. For better readability, in all following text when we talk about the correctness of a map, then we do not mean the correctness of the complete map, but the correctness of the description by the map of a single environment element. Similarly, when we talk about the verification of the map, we mean the verification of the map's description of a single environment element.

In our problem statement we assume that the map describes the geometry of each environment element relative to the ego vehicle. This means that we assume that the ego vehicle has been localized relative to the map and that there is no uncertainty about this localization.

First we specify what we mean by the correctness of a map. Let S be the parameter space used to model the element of concern in the a priori map. Let e be an environment element that is described by the a priori map. Instead of specifying a single element of S in order to describe e , the map specifies a set of possible values $A \subseteq S$ that environment element e may have. Correctness of the map is defined such that the map is correct in relation to e if and only if there exists an environment element with parameters x_r in the real world such that $x_r \in A$.

Typically, the set A is not described explicitly in the map for each environment element individually. Instead, each element is described by the mean value of all elements in A and a maximum deviation from that mean value. This maximum deviation may be the same for all map elements of a certain type and in such cases it only has to be specified once for the whole map. However, in cases where the uncertainty about the parameters of map elements is not constant across the map, the set A has to be described for each map element. For the discussion in this section it is necessary that each element in the map has a set A , but it is irrelevant how these sets are described.

A measurement can be the direct output of a physical sensor or be the result of a process that processes the signal produced by a physical sensor. For example, a laser scanner produces a sequence of range-bearing tuples. This sequence can be used as a measurement for the verification of central islands, but it is also possible to let the central island detector of chapter 4 process this sequence and to use the output of this detector as a measurement. Note that using the output of the central island detector does not contradict with the requirement that map verification should not be based on comparison: a detected circle does not necessarily correspond to a central island, and even if it does, it may not provide a sufficiently accurate description of the central island. Another example is the light-intensity image produced by a camera. This image can be used as a measurement for the verification of lane boundaries, but it is also possible to let an edge detector process this image and to use the set of detected edges as a measurement.

In this chapter we consider time to be discrete, where at each time instant there is exactly one measurement and the first measurement is done at time zero. In case a process produces more than one measurement based on a single sensor measurement (e.g. the central island detector detects two circles in a single point sequence), those measurements are considered to be measurements at different subsequent time instances. The sequence $M_{0:t} = [Z_0, Z_1, \dots, Z_t]$ is used to represent the random variables of all measurements done until the current time t , where Z_i is the random variable for the measurement at time i .

The goal of a map verification process is to compute the probability that the map is correct given all the measurements thus far. More formally, let C be the random Boolean variable indicating whether the map description of environment element e is correct. (Note that C depends on the element environment element e , but we simply write C instead of $C(e)$ since the element e is fixed at this point.) To verify the correctness of element e in the map, we now need to compute the following probability:

$$p(C = \text{true} \mid m_{0:t}) \quad (6.1)$$

where $m_{0:t}$ is the sequence of actually measured values (i.e. it is a realisation of $M_{0:t}$).

Let random variable X_z be the parameters of the real-world environment element that induced the latest measurement (note that this does not have to be the same element as described by the map). We assume that the latest measurement Z_t is conditionally independent of all previous measurements $M_{0:t-1}$ given the element parameters X_z . That is, for each realisation z_t , $m_{0:t-1}$ and x_z of Z_t , $M_{0:t-1}$ and X_z respectively, we have:

$$p(z_t \mid x_z, m_{0:t-1}) = p(z_t \mid x_z) \quad (6.2)$$

Such a measurement independence can be assumed in cases where the measurement Z_t is a function of the element parameters X_z plus an error term E_t (for which all the distribution parameters are known) that is independent of element parameters X_z or previous measurements:

$$Z_t = g(X_z) + E_t$$

where g is some function. This is commonly the case for a hysteresis-free and calibrated measurement device, and therefore the validity of assumption of equation 6.2 is not questioned here.

Although in this chapter the assumption of equation 6.2 is not questioned, this chapter does investigate the validity of another assumption of conditional measurement independence; namely the assumption that, given the correctness c of the map (which can be true or false), the latest measurement Z_t is conditionally independent of all previous measurements $M_{0:t-1}$:

$$p(z_t \mid c, m_{0:t-1}) = p(z_t \mid c) \quad (6.3)$$

Conditional measurement independencies like these are commonly assumed in probabilistic recursive state estimation, as indicated by Thrun et al. [TBF05], where in our case the correctness C of the map is the state to be estimated.

The goal of the presented verification method is to compute the map correctness probability as in equation 6.1 under the assumptions of conditionally independent measurements as in equations 6.2 and 6.3. The goal of the analysis is to find out whether, in practice, the assumption of equation 6.3 is valid, and what the effects on the performance verification model are in case the assumption does not hold.

6.4 A Generic Probabilistic Map Verification Model

The goal of the map verification process is to estimate whether the map is correct ($C = \text{true}$) or not ($C = \text{false}$) based on a series of measurements $m_{0:t} = [z_0, z_1, \dots, z_t]$ until time t . That is, the map verification process must compute the probability

$$p(c \mid m_{0:t})$$

for the two possible realisations of C : $c = \text{true}$ and $c = \text{false}$. This conditional distribution of C can be computed by Bayesian estimation. That is, from the distribution $p(c | m_{0:t-1})$ based on all previous measurements the distribution can be recursively updated when including the latest measurement z_t :

$$p(c | m_{0:t}) = p(c | m_{0:t-1}, z_t) = \frac{p(z_t | c, m_{0:t-1}) \cdot p(c | m_{0:t-1})}{p(z_t | m_{0:t-1})}$$

Now if we assume that the latest measurement Z_t is conditionally independent from all previous measurements $M_{0:t-1}$ given state C , we have:

$$p(c | m_{0:t}) = \eta_t \cdot p(z_t | c) \cdot p(c | m_{0:t-1}) \quad (6.4)$$

where $\eta_t = p(z_t | m_{0:t-1})^{-1}$ is a constant that does not depend on c , which can be computed by normalizing the distribution for C (i.e. solving $p(C = \text{true} | m_{0:t}) + p(C = \text{false} | m_{0:t}) = 1$ for η_t using equation 6.4). Here $p(c | m_{0:t-1})$ is known from the distribution that was computed before knowing the latest measurement and $p(z_t | c)$ is the probability density for measurement value z_t given the value c for the state C . Based on the problem statement in section 6.3, we will now derive an expression for the probability density $p(z_t | c)$. From now on we will omit the subscript of z_t and use z instead to simplify notation.

Let us first consider $p(z_t | c)$ for the case where the map is correct, that is, $c = \text{true}$. In this case there exists an environment element with parameters x in the real world such that $x \in A$. Let x_r be such an environment element. Although there exists a real-world environment element that fits the map description, it may be possible that the measurement originates from a different real-world environment element. We introduce a random Boolean variable O that indicates whether measurement z originates from element x_r . Then, by the law of total probability, we can write:

$$p(z | C = \text{true}) = p(z | C = \text{true}, O = \text{true}) \cdot p(O = \text{true} | C = \text{true}) + p(z | C = \text{true}, O = \text{false}) \cdot p(O = \text{false} | C = \text{true}) \quad (6.5)$$

Let us consider the case where $O = \text{true}$ and let X_z be the random variable for the parameters of the real-world element from which measurement z originates. Then $p(z | C = \text{true}, O = \text{true})$ can be written as:

$$p(z | C = \text{true}, O = \text{true}) = \int_S p(z | x_z, C = \text{true}, O = \text{true}) \cdot p(x_z | C = \text{true}, O = \text{true}) dx_z$$

We assume that given the measurement's origin, measurement Z is independent of C and O , that is we assume $p(z | x_z, c, o) = p(z | x_z)$ for any z, x_z, c and o . Note that the distribution $p(z | x_z)$ is typically given by the measurement model of the sensor/detector. This gives us:

$$p(z | C = \text{true}, O = \text{true}) = \int_S p(z | x_z) \cdot p(x_z | C = \text{true}, O = \text{true}) dx_z \quad (6.6)$$

Because $X_z \in A$ when $C = \text{true}$ and $O = \text{true}$, the probability density $p(x_z | C = \text{true}, O = \text{true}) = 0$ if $x_z \notin A$. So the integral over parameter space S in equation 6.6 reduces to an integral over A :

$$p(z | C = \text{true}, O = \text{true}) = \int_A p(z | x_z) \cdot p(x_z | C = \text{true}, O = \text{true}) dx_z$$

Let V be the set of parameters of all elements that may be observable. Also here $X_z \in V$ when $C = \text{true}$ and $O = \text{true}$, and therefore $p(x_z | C = \text{true}, O = \text{true}) = 0$ if $x_z \notin V$. The integral can thus be further reduced to:

$$p(z | C = \text{true}, O = \text{true}) = \int_{A \cap V} p(z | x_z) p(x_z | C = \text{true}, O = \text{true}) dx_z \quad (6.7)$$

If the map and sensor model provide no further information about a probability density distribution for the parameters x_r of element e , then we assume each possible value for x_r within $A \cap V$ equally

likely with probability density $\gamma_{A \cap V}$ (here $\gamma_{A \cap V}$ is the reciprocal of the Lebesgue measure of $A \cap V$):

$$p(z \mid C = \text{true}, O = \text{true}) = \gamma_{A \cap V} \int_{A \cap V} p(z \mid x_z) dx_z \quad (6.8)$$

Before continuing to present the verification model, a note must be made on the choice for a uniform distribution for the parameters of X_z inside $A \cap V$. This choice is based on the *principle of indifference*. As described by Keynes [Key21]: “*The Principle of Indifference asserts that if there is no known reason for predicating of our subject one rather than another of several alternatives, then relatively to such knowledge the assertions of each of these alternatives have an equal probability. Thus equal probabilities must be assigned to each of several arguments, if there is an absence of positive ground for assigning unequal ones.*”. Care should be taken however when applying this principle and especially in the case of continuous parameters. Namely, the assumed uniform probability density distribution for X_z depends on the choice of parameters used to model the relevant quantities of the element X_z . Keynes illustrates this by the following example. Consider a probability density distribution of the specific volume of a given substance, that is known to lie within a given range, but no further information is known that might indicate what the value of the specific volume can be. Based on the principle of indifference the probability density can be chosen to be uniform within that range. Now consider the substance’s density. If v is the specific volume, then $1/v$ is the density. This means that if the probability density distribution for the specific volume is uniform, that the probability density for the density is not uniform. However according to the principle of indifference, the distribution for the density should also be uniform. We see in this case that two measures (or models if you want) of the same objective quantity, yield different probability density distributions by applying the principle of indifference for those measures. The goal of this thesis is not to provide a solution to this problem, which seems inevitable in the application of probability theory when prior probability (density) distributions have to be chosen. One should be aware of the existence of this problem and, while applying the presented verification model, one should carefully select a parametrization for X_z such that a uniform distribution for it can be justified.

Let us now consider the case where $O = \text{false}$ in order to evaluate the second term of equation 6.5. This term can be written as:

$$p(z \mid C = \text{true}, O = \text{false}) = \int_S p(z \mid x_z, C = \text{true}, O = \text{false}) \cdot p(x_z \mid C = \text{true}, O = \text{false}) dx_z$$

Since $X_z \in V \setminus A$ when $C = \text{true}$ and $O = \text{false}$ and Z is conditionally independent of O and C , the above equation can be rewritten to:

$$p(z \mid C = \text{true}, O = \text{false}) = \int_{V \setminus A} p(z \mid x_z) \cdot p(x_z \mid C = \text{true}, O = \text{false}) dx_z \quad (6.9)$$

Since we have no further knowledge about the possible values for the measurement’s origin, we assume each value within $V \setminus A$ equally likely with probability density $\gamma_{V \setminus A}$:

$$p(z \mid C = \text{true}, O = \text{false}) = \gamma_{V \setminus A} \cdot \int_{V \setminus A} p(z \mid x_z) dx_z \quad (6.10)$$

Now let us consider the case where the map data is incorrect, that is $C = \text{false}$. Similar to above, the measurement probability density for this case can be written as:

$$p(z \mid C = \text{false}) = \gamma_{V \setminus A} \cdot \int_{V \setminus A} p(z \mid x_z) dx_z \quad (6.11)$$

By combining equations 6.5, 6.8, 6.10 and 6.11, we get the following expression for $p(z \mid c)$ in equation 6.4:

$$p(z \mid c) = \begin{cases} p(O = \text{true} \mid c) \cdot \gamma_{A \cap V} \cdot \int_{A \cap V} p(z \mid x_z) dx_z \\ \quad + p(O = \text{false} \mid c) \cdot \gamma_{V \setminus A} \cdot \int_{V \setminus A} p(z \mid x_z) dx_z & \text{if } c = \text{true} \\ \gamma_{V \setminus A} \cdot \int_{V \setminus A} p(z \mid x_z) dx_z & \text{if } c = \text{false} \end{cases} \quad (6.12)$$

This concludes the formulation of the generic probabilistic map verification model. An implementation of this model, for a specific kind of environment element based on a specific kind of measurements, includes:

- A definition of the parameter space S .
- A characterisation of the subspaces A and V .
- A specification of the measurements and how they are represented.
- A definition of the probability density distribution $p(z \mid x_z)$ for measurement Z given the parameters x_z of the environment element from which the measurement originates.
- A definition of the probability $p(O = \text{true} \mid C = \text{true})$ that the measurement originates from an element in A ($O = \text{true}$) given the fact that the map is correct ($C = \text{true}$).
- An algorithm for the computation of the integrals of equations 6.8, 6.10 and 6.11.

6.5 Implementation of the Verification Model for Central Islands

This section describes how the map verification model of section 6.4 can be implemented for verifying the map description of a roundabout central island, based on the output of the central island detector discussed in chapter 4. We do this by specifying each of the aspects required for an implementation of the model, according to the list that concludes section 6.4.

It has to be noted that some simplifications and assumptions are made in this implementation that will be explained later. The main purpose of the implementation presented in this section is the analysis of the verification model. The simplifications and assumptions will not affect the verification method's output in such way that it will lead to different conclusions. If this implementation were to be used in practice, then the effects of the simplifications and assumptions should be investigated.

Parameter space S . The real-world element to be verified is the central island of a roundabout. We model a central island as a circle in the xy -plane that is parametrized by a three-dimensional vector (k_x, k_y, k_r) , where k_x and k_y are the circle's centre x -coordinate and y -coordinate respectively, and k_r is the radius of the circle. We assume the radius is bounded by a maximum radius R_{\max} and a minimum radius R_{\min} . Thus $S = \mathbb{R} \times \mathbb{R} \times [R_{\min}, R_{\max}]$.

Correct-map space A . The map description of a central island defines an annulus in which the boundary of the real-world central island must lie. Hence, the space A of all correct central islands consists of all circles lying within this annulus. Although A itself is not an annulus (it is a set of circles), for the sake of simplicity we will talk about "annulus A " by which we mean "the annulus described by the map that defines the set of circles A ". We will use symbols r_{\min} and r_{\max} for the inner radius and outer radius respectively of annulus A .

Visibility space V . For this implementation, we define the space V of visible circles to be the set of circles intersecting the region that can be observed by a particular layer of the laser scanner. The region that can be observed by the laser scanner is defined by the position of the laser scanner, its field of view, a minimum distance and a maximum distance. Note that his definition of V is a simplification, where we for example ignore the fact that we can only see a circle from its exterior and not from its interior.

Measurement specification. In chapter 4 we concluded that the measured points corresponding to a correctly detected central island, give a quite accurate description of the central island's geometry at those positions where it was measured. We therefore want to use these point measurements for the verification of the central island. Each individual measured point corresponding to a detected circle is used as a separate measurement for the verification of central island.

Note that when a measurement is received, the map verification model only updates the probability of the map being correct. No other information is maintained about the measurement or about the possible state of the environment element that caused the measurement. If we would only use the point measurement for the verification of the central island, then such a measurement does not provide very precise information about the possible position of the centre of the circle from which the measurement originates. That is, we cannot say that one of two circles is more likely to be the point measurement's origin, if they have the same radius and distance to the measured point, but different directions for the vector from the measured point to the circle centre. However, the central island detector provides more specific information about the location of the circle from which the measurement originates via the centre of the fitted circle. Therefore, the measurement used to verify the correctness of a central island consists of a point measurement and the centre of the corresponding fitted circle. The latest measurement is written as z or (z_p, z_c) where z_p is the latest point measurement and z_c is the centre of the corresponding fitted circle.

Note that treating the combination of each individual point measurement with its corresponding circle centre will make measurements dependent, since measurements corresponding to the same circle detection will have the same value for the measured centre. The effect of this dependence on the performance of the presented implementation should be investigated if the implementation were to be used in practice. However, for the purpose of analysing the verification model in general, this dependence will have no influence. For this analysis we could, for example, assume that the central island detector only reports one measured point per detected circle to avoid the dependence between measurements corresponding to the same detection as described above.

Measurement probability density distribution. We use K as the symbol for the random variable representing the circular object that is the origin of the measurement¹. We thus have to specify the probability density $p(z_p, z_c | k)$ of the latest measurement (z_p, z_c) given the value k for the origin K of the measurement. When we assume that, given the value of K , the point measurement Z_p and centre measurement Z_c are independent, then this probability density can be written as:

$$p(z | k) = p(z_p, z_c | k) = p(z_p | k) \cdot p(z_c | k)$$

The value of $p(z_c | k)$ can then be obtained via the distribution illustrated by the circle centre measurement distribution shown in figure 4.27, after transforming z_c to the $x'y'$ -coordinate system (see section 4.6). For the value of $p(z_p | k)$ we make use of the statistics about the distance from a detected point to the border of the central island, as illustrated by figure 4.30. Let D be the random variable for the distance from the latest point measurement to the border of the circle that is the origin of this measurement. Let d be the distance from point z_p to circle k . From the distribution in figure 4.30 we know the probability density $p(D = d)$. Let dd be an infinitesimal distance. Then $p(D = d) \cdot dd$ is equal to the probability that the distance between the measurement and the circle from which it originates, is between the values d and $d + dd$:

$$p(D = d) \cdot dd = \Pr(d \leq D \leq d + dd)$$

Note that $p(D = d)$ is not the same as $p(z_p | k)$, since z_p is not the only point for which the distance to k is d . In fact $p(D = d) \cdot dd$ is the probability that Z_p is one of the points for which the distance to the circle is between the values d and $d + dd$. Given the circle $k = (k_x, k_y, k_r)$, the points with a distance to k between d and $d + dd$, form the annulus with centre (k_x, k_y) , inner radius $k_r + d$ and outer radius $k_r + d + dd$. Based on the statistics collected thus far, we cannot

¹That is, symbol X_z used in the generic verification model in section 6.4, is replaced by K in the implementation for central island verification.

say that one of the points inside this annulus is more likely to be measured than another point. Hence, we assume that the point measurement probability density is constant inside this annulus. This means that the probability density for each such measurement, and thereby the probability density $p(z_p | k)$, is equal to probability $p(d \leq D \leq d + dd)$ divided by the area $2\pi(d + k_r) \cdot dd$ of the annulus:

$$p(z_p | k) = \frac{\Pr(d \leq D \leq d + dd)}{2\pi(d + k_r) \cdot dd} = \frac{p(D = d)}{2\pi(d + k_r)} \quad (6.13)$$

Note that this measurement probability model neglects the fact that some parts of circle k may be outside the visibility region of the laser scanner.

Map-is-measurement-origin probability. In order to keep the analysis of the verification method simple, the probability $p(O = \text{true} | C = \text{true})$ that the measurement originates from a circle in A ($O = \text{true}$) given the fact that the map is correct ($C = \text{true}$), is in this implementation assumed to be constant. If this implementation were to be used in practice, it should be considered to make this probability depend on the distance of the ego vehicle to annulus A based on the good-to-total detection ratios discussed in section 4.6 and listed in appendix B.

Computation of the integrals. The integrals of equations 6.8, 6.10 and 6.11 can, of course, be computed purely numerically for this chapter's analysis of the presented verification model. However, since S is a three dimensional space, the integration over $V \setminus A$ and $V \cap A$ (which are large, since together they contain all possible circles that can be measured), can be very costly in terms of computation time. This will make the implementation infeasible for practical use. In order to show that the implementation has the potential to be practically usable in terms of computational complexity, section 6.6 demonstrates how the three-dimensional integrals can be analytically reduced to one-dimensional integrals which can then be numerically computed. The algorithm for this was also used for the analysis in section 6.7.

6.6 Integral Computation

This section describes how the integrals of equations 6.8, 6.10 and 6.11 can be computed for the verification of central islands as specified by section 6.5. This section is included for the sake of completeness, by showing that the implementation has the potential to be practically usable in terms of computational complexity. Therefore this section is not essential for understanding the remainder of this chapter and readers who are more interested in the evaluation of the method may skip to section 6.7.

Recall that a measurement for the verification of a central island consists of a point measurement z_p and a measurement of the centre z_c , that are assumed to be independent given the parameters k of the circle from which the measurement originates. Therefore, the integral to be computed for equation 6.8 is

$$\gamma_{A \cap V} \int_{A \cap V} p(z_p | k) \cdot p(z_c | k) dk \quad (6.14)$$

and the integral to be computed for equations 6.10 and 6.11 is:

$$\gamma_{V \setminus A} \int_{V \setminus A} p(z_p | k) \cdot p(z_c | k) dk \quad (6.15)$$

The remainder of this section describes only how equation 6.14 can be computed, since equation 6.15 can be computed by a similar approach.

Section 6.6.1 explains how the circle space over which we integrate can be partitioned into a grid in order to simplify the computation. As a result, the problem of computing the integral of equation 6.14 reduces to the problem of computing for each grid cell, the integral of:

$$\int_{A \cap V \cap B} p(z_p | k) dk \quad (6.16)$$

where B is the set of circle parameters corresponding to the grid cell. Section 6.6.2 derives an analytic solution for:

$$\int_A p(z_p | k) dk$$

Based on this, section 6.6.3 derives, for a given a grid cell, an analytic solution for:

$$\int_{A \cap B} p(z_p | k) dk$$

Section 6.6.4 then shows how this solution can be modified, by making a few approximations to restrict the integral to only circle parameters within V in order to have an analytic solution for equation 6.16. The computation of the integral in equation 6.15 and the computation of $\gamma_{A \cap V}$ and $\gamma_{V \setminus A}$ are not discussed in this section, since a similar (simplified) approach can be used for these computations.

6.6.1 Partitioning the Integral's Space into a Grid

Note that the integral over $A \cap V$ in equation 6.14 is an integral over a three-dimensional space, since we describe a circle by a centre x -coordinate, centre y -coordinate and a radius. Now let us partition the circle parameter space into a set of cells G of an orthogonal grid of which one of the axes is parallel to the radius axis. Then the integral in equation 6.14 can be split up into integrals for each of the cells of the parameter space:

$$\int_{A \cap V} p(z | k) dk = \sum_{g \in G} \int_{A \cap V \cap B(g)} p(z_p | k) \cdot p(z_c | k) dk$$

where $B(g)$ is the set of all circle parameter values in grid cell g . Recall that to obtain a value for $p(z_c | k)$, we make use of the error distribution for circle centre measurements as illustrated in figure 4.27. Also recall that to obtain a value for $p(z_p | k)$, we make use of the distribution for the distance of a measured point to the border of the corresponding circle as illustrated in figure 4.30. As concluded in chapter 4, the distance of a measured point to the border of the corresponding circle is much smaller than the uncertainty of the measured circle centre. We will make use of this fact for the computation of the integral. Let us choose grid G such that the cells are as large as possible but small enough such that each circle parameter value k in a cell has approximately the same value for $p(z_c | k)$. That is, for each cell $g \in G$ we have $p(z_c | k) \approx p(z_c | k_g)$, where k_g is an arbitrary element of cell g . Then we can write:

$$\int_{A \cap V} p(z | k) dk \approx \sum_{g \in G} p(z_c | k_g) \cdot \int_{A \cap V \cap B(g)} p(z_p | k) dk \quad (6.17)$$

Note that since for different values of k within a certain grid cell the value $p(z_p | k)$ may vary significantly, $p(z_p | k)$ is not moved outside the integral in the equation above. Because the radius of a circle k has no influence on the probability density for the measured circle centre $p(z_c | k)$, we choose the cells of grid G to only vary in their x and y coordinates.

The error distribution of the measured centre is represented by a two-dimensional histogram in the $x'y'$ -coordinate system (see section 4.6.5 for the definition of the coordinate system). It would be very convenient if we could choose grid G in such way that each of its cells corresponds with a bin of this histogram, since then computation of the sum in equation 6.17 would translate into an iteration over the histogram's bins and $p(z_c | k_g)$ would be equal to the probability density described by the value of the corresponding bin. Unfortunately this is not possible since the orientation of the $x'y'$ -coordinate system depends on the circle k from which the measurement originates: the x' -axis points in the same direction as the vector from the laser scanner's origin to the centre of k . However for a given value of the centre measurement z_c , the $x'y'$ coordinate system can be approximated by redefining the $x'y'$ -coordinate system such that the x' axis has the same direction as the vector from the laser scanner's origin to the measured circle centre z_c (instead of

Algorithm *ComputeIntegralOverAIntersection* $V(z_p, z_c, A, V)$

1. Let H be the centre-measurement-error histogram as illustrated in figure 4.27
2. Let G be the grid that is aligned with the x' -, y' - and radius-axis and for which, given the centre measurement z_c , each cell corresponds to a bin in H
3. $s \leftarrow 0$
4. **for** each cell $g \in G$
5. **do** Let h be the bin of H corresponding to cell g
6. $p \leftarrow$ the probability density of bin h
7. $u \leftarrow \int_{A \cap V \cap B(g)} p(z_p | k) dk$
8. $s \leftarrow s + p \cdot u$
9. **return** s

Algorithm 6.1: Given the point measurement z_p , centre measurement z_c , a description of A and a description of visibility space V , this algorithm computes the integral in equation 6.14.

the centre of k). This has been verified to be a good approximation via the comparison of the centre error histogram visualised by figure 4.27 with the centre error histogram that was computed (based on the same measurements) with the approximated $x'y'$ -coordinate system. Whenever we refer to the $x'y'$ -coordinate system in the remainder of this chapter, we mean the approximation of this coordinate system.

With the approximated $x'y'$ -coordinate system we can define G to be the grid that is aligned with the x' -axis, the y' -axis and the radius axis, for which each cell has the minimum radius value R_{\min} , maximum radius value R_{\max} and the $x'y'$ -coordinates correspond to a bin in the centre-measurement-error histogram illustrated in figure 4.27 (note that such a correspondence can only be made given the centre measurement z_c).

Algorithm 6.1 shows how equation 6.17 can be evaluated. Line 7 involves the computation of the following integral of the point-measurement probability:

$$\int_{A \cap V \cap B} p(z_p | k) dk$$

where B is the set of all circle parameter values within a certain grid cell. The evaluation of this integral is discussed in the following sections.

6.6.2 Integration over A

Before we consider the evaluation of equation 6.16 which integrates over $A \cap V \cap B$, we consider in this section the evaluation of the integral over the set of circles A that are within the annulus:

$$\int_A p(z_p | k) dk \tag{6.18}$$

Note that this is an integral over a three-dimensional space. This section will show that it can be rewritten to a one-dimensional integral that can be evaluated numerically in an efficient manner.

Exploiting symmetry. Recall from equation 6.13 that the probability density for a point measurement z_p , given that the circle from which the measurement originates is k , is:

$$p(z_p | k) = \frac{p(D = d)}{2\pi(d + k_r)}$$

where d is the smallest distance between z_p and the border of k , and k_r is the radius of k . This means that $p(z_p | k)$ has the same value for every circle k with the same radius and the same

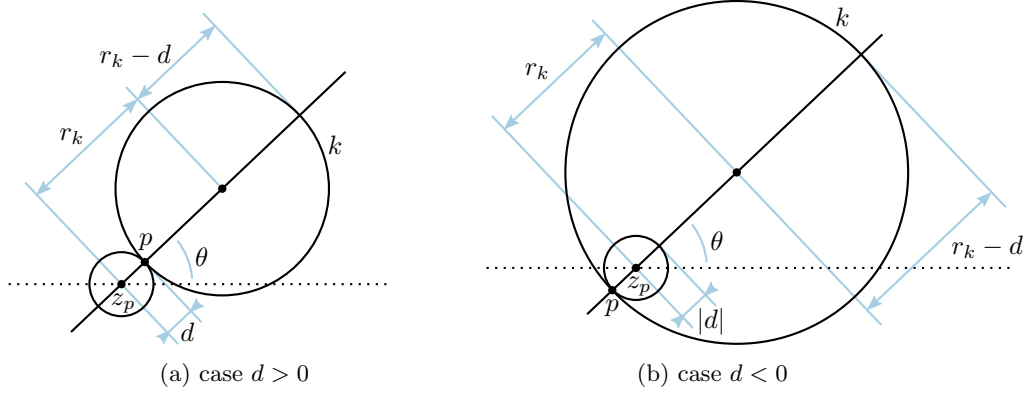


Figure 6.1: These illustrations show how a circle k is described by r_k , θ and d relative to a point measurement z_p . The dotted line is the line through point z_p and the centre of annulus A which is somewhere to the right of z_p . Point p is the point of k with the closest distance to z_p .

distance to point z_p . We want to exploit the symmetry of $p(z_p | k)$ around point z_p to simplify integral 6.18. In order to do so, we express a circle k in a coordinate system different from the x - y -radius-coordinate system we usually use for circles. In this coordinate system a circle k is expressed by an r_k , θ and d parameter, as illustrated by figure 6.1. The r_k parameter value represents the distance from the centre of circle k to the measured point z_p . The value θ represents the angle between the vector from z_p to the centre of k and the vector from z_p to the centre of annulus A . That means that (r_k, θ) are the coordinates of k 's centre in a polar coordinate system with its origin at z_p . We restrict circle parameter θ to values in the interval $[-\pi, \pi]$. Value d is the smallest distance from point z_p to circle k (which conforms to the meaning of symbol d in the rest of this thesis), where a positive value means that z_p is outside circle k (see figure 6.1a) and a negative value means that z_p is enclosed by circle k (see figure 6.1b).

We can now rewrite the integral of equation 6.18 to an integral over A in r_k, θ, d coordinates:

$$\int_A p(z_p | k) dk = \int_{-\infty}^{\infty} \int_{\Theta(d)} \int_{r_{k,\min}(d,\theta)}^{r_{k,\max}(d,\theta)} p(z_p | k) \cdot r_k dr_k d\theta dd$$

where $\Theta(d)$ is the set of values for circle parameter θ for which there exist circles that are inside annulus A , and where the range $[r_{k,\min}(d, \theta), r_{k,\max}(d, \theta)]$ is, given the circle's d and θ parameter value, the range of all r_k parameter values for which the circle is inside annulus A . Now by using the definition in equation 6.13 for $p(z_p | k)$ and noting that $d + r_r = r_k$, the above equation can be rewritten to:

$$\int_A p(z_p | k) dk = \int_{-\infty}^{\infty} \int_{\Theta(d)} \int_{r_{k,\min}(d,\theta)}^{r_{k,\max}(d,\theta)} \frac{p(D = d)}{2\pi r_k} \cdot r_k dr_k d\theta dd,$$

which, in turn, can be rewritten to:

$$\int_A p(z_p | k) dk = \frac{1}{2\pi} \int_{-\infty}^{\infty} p(D = d) \int_{\Theta(d)} (r_{k,\max}(d, \theta) - r_{k,\min}(d, \theta)) d\theta dd. \quad (6.19)$$

What now remains is a derivation of the functions $r_{k,\min}$, $r_{k,\max}$ and Θ and solving the inner integral of equation 6.19.

Derivation of $r_{k,\min}(d, \theta)$ and $r_{k,\max}(d, \theta)$. Let us assume that d and θ are given. Note that the radius of the circle is $r_k - d$. Then the maximum value $r_{k,\max}(d, \theta)$ for r_k is defined by the outer border of annulus A , since the radius of the circle has to be small enough to be enclosed by the outer border of the annulus. Observe that that point p also has to be enclosed by the

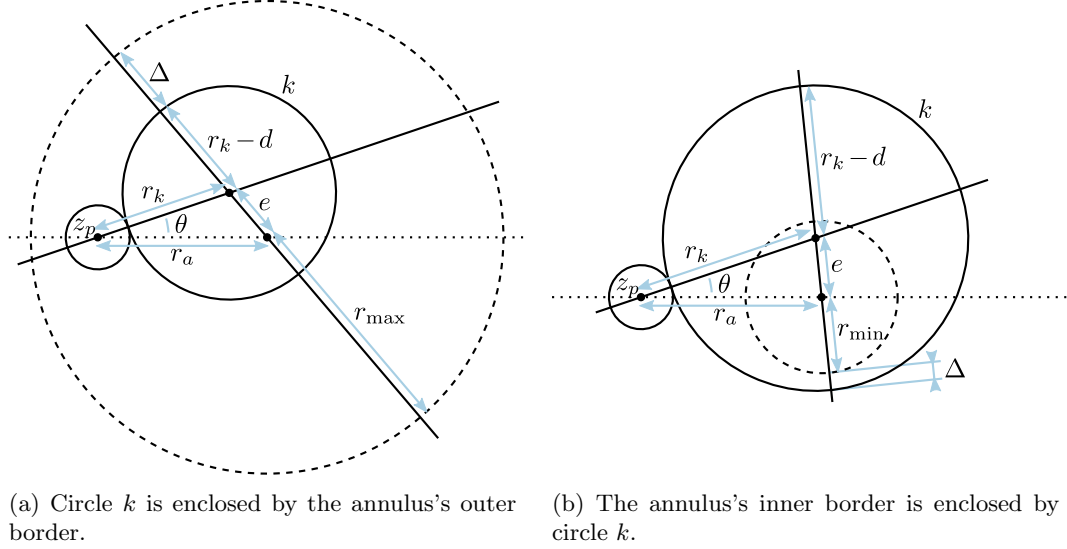


Figure 6.2: This figure illustrates how the smallest distance Δ between a circle k and the annulus's outer border or inner border can be described in terms of r_k , d , r_a and r_{\max} or r_{\min} . Here e is the distance between the circle's centre and the annulus's centre: $e = \sqrt{r_a^2 + r_k^2 - 2r_a r_k \cos(\theta)}$. The dashed circles are the borders of the annulus.

outer border of the annulus, since p is by definition a point of the circle. Let us now assume that p is enclosed by the outer border of the annulus. The smallest distance between a circle with radius r_1 and a circle enclosed by it with radius r_2 , given the distance e between its two centres, is $r_1 - r_2 - e$. In our case this means that the smallest distance between the circle and the outside border of the annulus is $r_{\max} - (r_k - d) - e$, where e is the distance between the centre of the circle and the centre of the annulus (see figure 6.2a). Let r_a be the distance between measurement z_p and the centre of the annulus. Then, by the law of cosines, $e = \sqrt{r_a^2 + r_k^2 - 2r_a r_k \cos(\theta)}$. Now r_k has reached its maximum value when the circle hits the border of the outside annulus, that is the distance between the circle and outside border is zero:

$$r_{\max} - (r_k - d) - \sqrt{r_a^2 + r_k^2 - 2r_a r_k \cos(\theta)} = 0$$

Moving the last term to the right we get:

$$r_{\max} - (r_k - d) = \sqrt{r_a^2 + r_k^2 - 2r_a r_k \cos(\theta)} \quad (6.20)$$

Note that the right side of this equation is non-negative. This means that there is a only a solution for this equation if $r_{\max} - (r_k - d)$ is non-negative. In that case equation 6.20 gives us the maximum value for circle parameter r_k :

$$r_{k,\max}(d, \theta) = \frac{r_a^2 - (d + r_{\max})^2}{2(r_a \cos(\theta) - (d + r_{\max}))} \quad (6.21)$$

The minimum value $r_{k,\min}(d, \theta)$ for circle parameter r_k is defined by the inner border of annulus A , since the radius of the circle has to be large enough to enclose the inner border of the annulus. Note that point p has to be outside the inner border of the annulus, since p is by definition a point on the circle. We therefore now assume that point p is outside the inner border of the annulus. First observe that if d is negative (meaning measurement z_p is enclosed by the circle) and if the circle centred at z_p entirely encapsulates the annulus's inner border (i.e. $-d \geq r_a + r_{\min}$), then for every value of circle parameter r_k the annulus's inner border is enclosed by the circle. Now let us consider the complementary case where $-d < r_a + r_{\min}$. Then the smallest value of circle

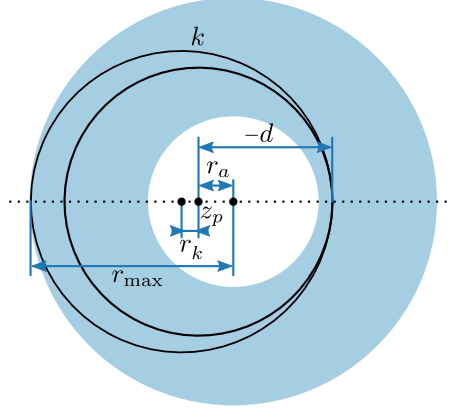


Figure 6.3: This figure illustrates that in case $r_a - r_{\max} \leq d \leq r_a - r_{\min}$ and $-d \geq r_a + r_{\min}$ (which implies $d \leq 0$), there exists a circle k with $\theta = \pi$ at distance d from point z_p that is enclosed by the annulus's outer border. The particular k illustrated here hits the annulus's outer border and therefore its diameter equals $r_{\max} - d - r_a$, which means that $r_k = d + (r_{\max} - d - r_a)/2 = (r_{\max} + d - r_a)/2$. Observe that for any other value of θ , circle k will also be enclosed by the annulus's outer border.

parameter r_k for which the radius of the circle is large enough to enclose the inner border of the annulus, is the value for which the distance between the circle and the inner border of the annulus is zero (see figure 6.2b):

$$(r_k - d) - r_{\min} - \sqrt{r_a^2 + r_k^2 - 2r_a r_k \cos(\theta)} = 0 \quad (6.22)$$

This equation only has a solution if $r_k - d - r_{\min}$ is non-negative. In that case equation 6.22 gives us the minimum value for circle parameter r_k :

$$r_{k,\min}(d, \theta) = \begin{cases} \frac{r_a^2 - (d + r_{\min})^2}{2(r_a \cos(\theta) - (d + r_{\min}))} & \text{if } -d < r_a + r_{\min} \\ 0 & \text{if } -d \geq r_a + r_{\min} \end{cases} \quad (6.23)$$

Derivation of $\Theta(d)$. Let us now derive the set $\Theta(d)$, the set of values for circle parameter θ in the range $[-\pi, \pi]$ for which there exist a circle of which the distance to point z_p is d and that is inside annulus A . First, observe that if $r_a - r_{\max} > d$ (also for $d < 0$) then there will be no circle enclosed by the outer border of annulus A , and thus in that case $\Theta(d)$ will be empty. Similarly, there will be no circle enclosing the inner border of annulus A if $d > r_a - r_{\min}$, and thus also in that case $\Theta(d)$ will be empty. Second, observe that if $r_a - r_{\max} \leq d \leq r_a - r_{\min}$ then there always exists a circle such that the distance between this circle and point z_p equals d , and the circle is inside the annulus. Third, observe in case $r_a - r_{\max} \leq d \leq r_a - r_{\min}$ that if $-d \geq r_a + r_{\min}$ (which implies $d \leq 0$) then every circle with distance d to point z_p encloses the annulus's inner border. Furthermore, for each θ the circle with $r_k = (r_{\max} + d - r_a)/2$ (see figure 6.3) is then enclosed by the annulus's outer border, which means that in that case $\Theta(d) = [-\pi, \pi]$.

Finally, let us consider, given a value for d , the circle $k^*(d)$ for which the distance to point z_p is d , for which parameter θ is non-negative and that touches both the inner border and outer border of the annulus (see figure 6.4). Let the r_k and θ parameter of $k^*(d)$ be $r_k^*(d)$ and $\theta^*(d)$ respectively. Observe that a circle inside the annulus that touches both the annulus's inner border and outer border, has a radius of $(r_{\min} + r_{\max})/2$ and the distance of its centre to the annulus's centre is $(r_{\max} - r_{\min})/2$. This means that $r_k^*(d) = d + (r_{\min} + r_{\max})/2$ and that (by applying the law of cosines):

$$\begin{aligned} \theta^*(d) &= \arccos \left(\frac{r_a^2 + r_k^*(d)^2 - \left(\frac{r_{\max} - r_{\min}}{2}\right)^2}{2r_a r_k^*(d)} \right) \\ &= \arccos \left(\frac{d^2 + dr_{\max} + dr_{\min} + r_{\max} r_{\min} + r_a^2}{r_a(2d + r_{\max} + r_{\min})} \right) \end{aligned}$$

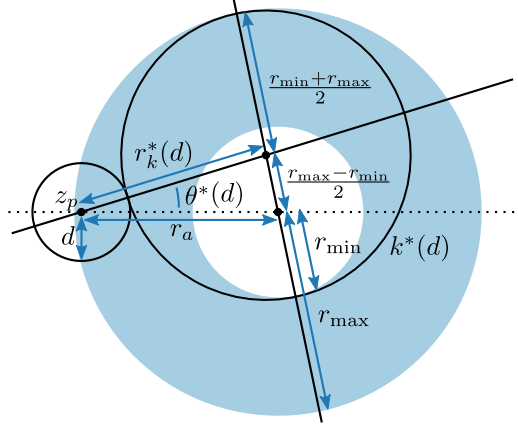


Figure 6.4: Given a value for d , the circle $k^*(d)$ is the circle for which the distance to point z_p is d , for which parameter θ is non-negative and that touches both the inner border and outer border of the annulus.

Now observe that $k^*(d)$ is the only circle for which the distance to point z_p is d and for which the θ parameter is $\theta^*(d)$ that is inside the annulus (except the case where $\theta^*(d) = 0$): increasing its r_k parameter would make the circle intersect with the annulus's outer border and decreasing it would make the circle intersect the annulus's inner border. This means that $r_{k,\min}(d, \theta^*(d)) = r_{k,\max}(d, \theta^*(d))$. Note that we consider the case where $r_a - r_{\max} \leq d \leq r_a - r_{\min}$ and $-d < r_a + r_{\min}$, which means that we have $r_a \geq r_{\min} + d$ and $r_a > -(r_{\min} + d)$ and therefore $r_a \geq |r_{\min} + d| \geq 0$. Furthermore we have $r_a \leq r_{\max} + d \leq |r_{\max} + d|$. Consequently, $r_{k,\min}(d, \theta)$ is a monotonically increasing function of θ and $r_{k,\max}(d, \theta)$ is a monotonically decreasing function of θ in the interval $[0, \pi]$. This means that for $\theta > \theta^*(d)$ we have $r_{k,\min}(d, \theta) > r_{k,\max}(d, \theta)$ which implies that there exists no circle with such θ parameter that both is enclosed by the annulus's outer border and encloses the annulus's inner border. From this we can conclude $\theta^*(d)$ is the largest non-negative value (of at most π) for circle parameter θ , such that the circle is inside the annulus. Note that any circle with non-negative $\theta < \theta^*(d)$, distance d to point z_p and radius $(r_{\min} + r_{\max})/2$, is inside the annulus.

Concluding the discussion above $\Theta(d)$ can be written as:

$$\Theta(d) = \begin{cases} \emptyset & \text{if } r_a > d + r_{\max} \vee r_a < d + r_{\min} \\ [-\theta^*(d), \theta^*(d)] & \text{if } r_a - r_{\max} \leq d \leq r_a - r_{\min} \wedge -d < r_a + r_{\min} \\ [-\pi, \pi] & \text{if } r_a - r_{\max} \leq d \leq r_a - r_{\min} \wedge -d \geq r_a + r_{\min} \end{cases} \quad (6.24)$$

Solving the inner integral. Considering the values of circle parameter d for which $\Theta(d)$ is empty (see equation 6.24), the integral in equation 6.19 can be reduced to:

$$\int_A p(z_p | k) dk = \frac{1}{2\pi} \int_{r_a - r_{\max}}^{r_a - r_{\min}} p(d) \int_{\Theta(d)} (r_{k,\max}(d, \theta) - r_{k,\min}(d, \theta)) d\theta dd \quad (6.25)$$

Let us define function f_b for $b \in \{\max, \min\}$ as follows:

$$f_b(d, \theta) = \sqrt{r_a^2 - (d + r_b)^2} \operatorname{arctanh} \left(\frac{\tan(\theta/2)(r_a + d + r_b)}{\sqrt{r_a^2 - (d + r_b)^2}} \right) \quad (6.26)$$

Note that if $r_a^2 - (d + r_b)^2$ is negative, $f_b(d, \theta)$ is still a real number and can also be written as:

$$f_b(d, \theta) = \sqrt{(d + r_b)^2 - r_a^2} \operatorname{arctan} \left(\frac{\tan(\theta/2)(r_a + d + r_b)}{\sqrt{(d + r_b)^2 - r_a^2}} \right)$$

Now observe that for $b = \max$ or $b = \min \wedge -d < r_a + r_{\min}$:

$$\int r_{k,b}(d, \theta) d\theta = \int \frac{r_a^2 - (d + r_b)^2}{2(r_a \cos(\theta) - (d + r_b))} d\theta = f_b(d, \theta) + \text{constant} \quad (6.27)$$

Note that $f_b(d, \theta)$ is undefined for $\theta = \pi$. However the integral of $r_{k,b}$ from 0 to π can be computed as follows:

$$\int_0^\pi r_{k,b}(d, \theta) d\theta = \lim_{a \rightarrow \pi^+} \int_0^a r_{k,b}(d, \theta) d\theta = \lim_{a \rightarrow \pi^+} f_b(d, a) = \frac{\pi}{2} \sqrt{(d + r_b)^2 - r_a^2} \quad (6.28)$$

Using equations 6.27 and 6.28, we can rewrite equation 6.25 for the case $r_a - r_{\max} < -r_a - r_{\min}$ to:

$$\begin{aligned} \int_A p(z_p | k) dk &= \frac{1}{2} \int_{r_a - r_{\max}}^{-r_a - r_{\min}} p(d) \sqrt{(d + r_{\max})^2 - r_a^2} dd \\ &\quad + \frac{1}{\pi} \int_{-r_a - r_{\min}}^{r_a - r_{\min}} p(d) \cdot (f_{\max}(d, \theta^*(d)) - f_{\min}(d, \theta^*(d))) dd \end{aligned} \quad (6.29)$$

where the first term is the integral over the values of circle parameter d corresponding to the third case of the definition of $\Theta(d)$ in equation 6.24 (in this case we also have $r_{k,\min}(d, \theta) = 0$) and where the second term is the integral over the values of circle parameter d corresponding to the second case. In case $r_a - r_{\max} \geq -r_a - r_{\min}$, equation 6.25 can be rewritten to:

$$\int_A p(z_p | k) dk = \frac{1}{\pi} \int_{r_a - r_{\max}}^{r_a - r_{\min}} p(d) \cdot (f_{\max}(d, \theta^*(d)) - f_{\min}(d, \theta^*(d))) dd \quad (6.30)$$

Numerical evaluation. We have now shown that the three-dimensional integral of equation 6.18 can be rewritten to a one-dimensional integral over d (i.e. equations 6.29 and 6.30), the distance between the measured point and the distance of the circle from which the measurement originates. This integral can be numerically evaluated by iterating over the bins of the histogram illustrated in figure 4.30 and computing the integrand's value for each bin where the bin value is used as the value of $p(d)$.

6.6.3 Integration over $A \cap B$

In subsection 6.6.2 we showed how the integral in equation 6.18 over set of circles in A can be evaluated. Based on this result, we will in this section show how we can evaluate an integral of the form

$$\int_{A \cap B} p(z_p | k) dk \quad (6.31)$$

where B is the set of circle parameters corresponding to a cell of the grid as defined in section 6.6.1.

Adding the constraints of B . Now let us describe B by means of the variables x_{\min} , x_{\max} , y_{\min} , y_{\max} , R_{\min} and R_{\max} , such that B is the set of all circles for which the centre's x' -coordinate is within the interval $[x_{\min}, x_{\max}]$, the centre's y' -coordinate is within the interval $[y_{\min}, y_{\max}]$ and the radius is within the interval $[R_{\min}, R_{\max}]$.

Recall equation 6.25 for the integral over A :

$$\int_A p(z_p | k) dk = \frac{1}{2\pi} \int_{r_a - r_{\max}}^{r_a - r_{\min}} p(d) \int_{\Theta(d)} (r_{k,\max}(d, \theta) - r_{k,\min}(d, \theta)) d\theta dd$$

Here $\Theta(d)$ is the set of all values for θ for which there exists a circle within annulus A . Interval $[r_{k,\max}(d, \theta), r_{k,\min}(d, \theta)]$ is the interval of values for circle parameter r_k (the distance between z_p and the circle centre) of all circles with the specified θ and d that are within annulus A . That

is, functions Θ , $r_{k,\min}$ and $r_{k,\max}$ define the constraints on a circle's r_k and θ parameter in order to be a circle that is in A . Now for the integral of equation 6.31 over $A \cap B$ we will introduce similar functions Θ' , $r'_{k,\min}$ and $r'_{k,\max}$ that describe the constraints for circles that are in $A \cap B$. Equation 6.31 can then be written in a form similar to equation 6.31:

$$\int_{A \cap B} p(z_p | k) dk = \frac{1}{2\pi} \int_{r_a - r_{\max}}^{r_a - r_{\min}} p(d) \int_{\Theta'(d)} (r'_{k,\max}(d, \theta) - r'_{k,\min}(d, \theta)) d\theta dd \quad (6.32)$$

Here $\Theta'(d) = \Theta(d) \cap \Theta_B$, where Θ_B is the set of values for circle parameter θ for which there exists a circle with centre x' -coordinate in the interval $[x_{\min}, x_{\max}]$ and centre y' -coordinate in the interval $[y_{\min}, y_{\max}]$. Values $r'_{k,\min}(d, \theta)$ and $r'_{k,\max}(d, \theta)$ are defined such that:

$$\begin{aligned} [r'_{k,\min}(d, \theta), r'_{k,\max}(d, \theta)] &= [r_{k,\min}(d, \theta), r_{k,\max}(d, \theta)] \\ &\quad \cap [r_{k,\text{low}}(\theta), r_{k,\text{up}}(\theta)] \\ &\quad \cap [d + R_{\min}, d + R_{\max}] \end{aligned}$$

where for a given θ , $[r_{k,\text{low}}(\theta), r_{k,\text{up}}(\theta)]$ is the interval of all values for r_k for which the circle centre is within the axis-parallel rectangle defined by B . Interval $[d + R_{\min}, d + R_{\max}]$ comes from the constraint from B that the circle's radius (which is equal to $r_k - d$) must be in the interval $[R_{\min}, R_{\max}]$. The following lemma will help us to obtain a closed-form expression for $r'_{k,\min}(d, \theta)$ and $r'_{k,\max}(d, \theta)$.

Lemma 5. *Let $[a_1, a_2]$ and $[b_1, b_2]$ be two intervals. Then if $c_1 = \min(\max(a_1, b_1), a_2)$ and $c_2 = \max(\min(a_2, b_2), a_1)$, for any function $f : \mathbb{R} \rightarrow \mathbb{R}$ the following holds:*

$$\int_{[a_1, a_2] \cap [b_1, b_2]} f(x) dx = \int_{[c_1, c_2]} f(x) dx$$

Proof. Let us first consider the case where intervals $[a_1, a_2]$ and $[b_1, b_2]$ do not overlap. That means that either $a_2 < b_1$ or $b_2 < a_1$. Let us consider the case that $a_2 < b_1$ (the case $b_2 < a_1$ is similar). Note that $a_1 \leq a_2$ and therefore $a_1 < b_1$. Thus:

$$c_1 = \min(\max(a_1, b_1), a_2) = \min(b_1, a_2) = a_2$$

Since $b_1 \leq b_2$ we have $a_2 < b_2$ and therefore:

$$c_2 = \max(\min(a_2, b_2), a_1) = \max(a_2, a_1) = a_2$$

The following equation concludes the proof in case of non-overlapping intervals:

$$\int_{[a_1, a_2] \cap [b_1, b_2]} f(x) dx = \int_{\emptyset} f(x) dx = 0 = \int_{a_2}^{a_2} f(x) dx = \int_{[c_1, c_2]} f(x) dx$$

Now let us consider the case where the intervals are overlapping. This means that $a_2 \geq b_1$ and $b_2 \geq a_1$. In case of overlapping intervals we have:

$$[a_1, a_2] \cap [b_1, b_2] = [\max(a_1, b_1), \min(a_2, b_2)]$$

Since both a_1 and b_1 are at most a_2 :

$$c_1 = \min(\max(a_1, b_1), a_2) = \max(a_1, b_1)$$

Since both b_2 and a_2 are at least a_1 :

$$c_2 = \max(\min(a_2, b_2), a_1) = \min(a_2, b_2)$$

So we can conclude that:

$$[a_1, a_2] \cap [b_1, b_2] = [\max(a_1, b_1), \min(a_2, b_2)] = [c_1, c_2] \quad \square$$

By applying lemma 5 twice we obtain the following definitions for $r'_{k,\min}(d, \theta)$ and $r'_{k,\max}(d, \theta)$:

$$r'_{k,\min}(d, \theta) = \min(\max(\min(\max(r_{k,\text{low}}(\theta), d + R_{\min}), d + R_{\max}), r_{k,\min}(d, \theta)), r_{k,\max}(d, \theta)) \quad (6.33)$$

$$r'_{k,\max}(d, \theta) = \max(\min(\max(\min(r_{k,\text{up}}(\theta), d + R_{\max}), d + R_{\min}), r_{k,\max}(d, \theta)), r_{k,\min}(d, \theta)) \quad (6.34)$$

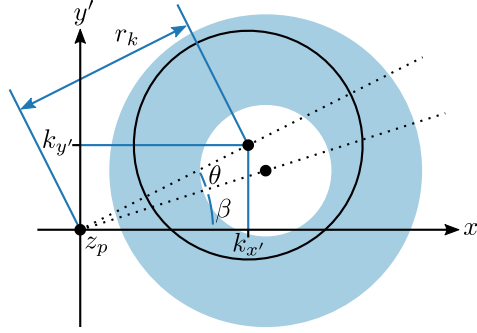


Figure 6.5: The Cartesian coordinate system in which the centre $(k_{x'}, k_{y'})$ of a circle is described is defined as the coordinate system that has point z_p as its origin and the centre of annulus A (the annulus is blue in this illustration) on the positive x' -axis.

Deriving $r_{k,\text{low}}(\theta)$, $r_{k,\text{up}}(\theta)$ and Θ_B . Recall that the set of possible circle centre coordinates in B is an axis-parallel rectangle in the $x'y'$ -coordinate system. Without loss of generality, we assume that the $x'y'$ -coordinate system has point z_p as its origin. Let β be the angle between the vector from z_p to the annulus's centre and the positive x' -axis (see figure 6.5). For an angle $\phi \in [-\pi, \pi]$ relative to the positive x' -axis, we define $\psi(\phi)$ to be the corresponding angle in $[-\pi, \pi]$ relative to the vector from z_p to the annulus's centre (i.e. θ value):

$$\psi(\phi) = \begin{cases} \phi - \beta & \text{if } \phi - \beta \geq -\pi \\ \phi - \beta + 2\pi & \text{otherwise} \end{cases}$$

We introduce function Ψ , in order to map angular intervals relative to the positive x' -axis to the corresponding angular intervals relative to the vector from z_p to the annulus's centre. For two angles $\phi_a, \phi_b \in [-\pi, \pi]$ relative to the positive x' axis, $\Psi(\phi_a, \phi_b)$ is defined to be the corresponding set of angles in $[-\pi, \pi]$ relative to the vector from z_p to the annulus's centre (i.e. θ values), that are covered when we start with an angle of ϕ_a and increase it (in counter-clockwise direction) until we reach the first angle coterminal to ϕ_b :

$$\Psi(\phi_a, \phi_b) = \begin{cases} [\psi(\phi_a), \psi(\phi_b)] & \text{if } \psi(\phi_a) \leq \psi(\phi_b) \\ [\psi(\phi_a), \pi] \cup [-\pi, \psi(\phi_b)] & \text{otherwise} \end{cases}$$

Now let $(k_{x'}, k_{y'})$ be the $x'y'$ -coordinates of a circle's centre. Then $k_{x'}$ and $k_{y'}$ can be described in terms of the circle's r_k and θ as follows (see figure 6.5):

$$\begin{aligned} k_{x'} &= r_k \cos(\beta + \theta) \\ k_{y'} &= r_k \sin(\beta + \theta) \end{aligned}$$

Using this we can formulate the constraints on r_k that the centre of the circle must be inside the axis-parallel rectangle defined by B :

$$\begin{aligned} x_{\min} &\leq r_k \cos(\beta + \theta) \leq x_{\max} \\ y_{\min} &\leq r_k \sin(\beta + \theta) \leq y_{\max} \end{aligned}$$

Depending on the position of the rectangle of valid centre coordinates relative to the x' -axis and y' -axis, these constraints can be used to obtain a definition of $r_{k,\text{low}}(\theta)$ and $r_{k,\text{up}}(\theta)$. We consider the nine different cases listed in table 6.1 of positions of the axis-parallel rectangle, which are also illustrated in figure 6.7. Let ϕ_0, ϕ_1, ϕ_2 and ϕ_3 be the angle between the positive x' -axis and the vector from the origin to the rectangle's upper-right, upper-left, lower-left and lower-right corner respectively (see figure 6.6). Table 6.1 also shows for each of the cases the set Θ_B in terms of these corner angles. Table 6.2 shows for each of the cases and intervals of θ what the values of $r_{k,\text{low}}(\theta)$ and $r_{k,\text{up}}(\theta)$ are.

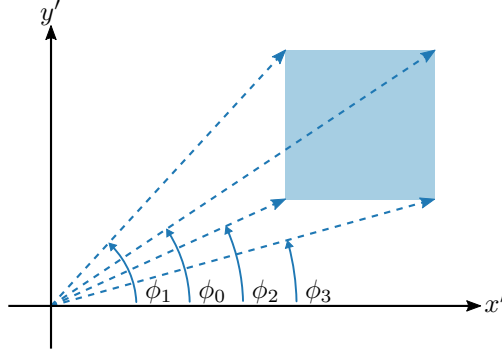


Figure 6.6: ϕ_0, ϕ_1, ϕ_2 and ϕ_3 are defined as the angles between the positive x' -axis and the vector between the origin and rectangle's upper-right corner, upper-left corner, lower-left corner and lower-right corner respectively.

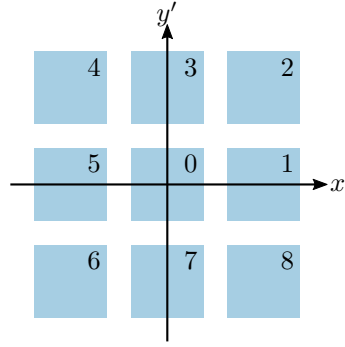


Figure 6.7: This figure shows the 9 different cases of the rectangle's position relative to the x' and y' -axis as described in table 6.1.

case	condition	Θ_B
0	$x_{\min} \leq 0 \leq x_{\max} \wedge y_{\min} \leq 0 \leq y_{\max}$	$[-\pi, \pi]$
1	$0 \leq x_{\min} \wedge y_{\min} \leq 0 \leq y_{\max}$	$\Psi(\phi_2, \phi_1)$
2	$0 \leq x_{\min} \wedge 0 \leq y_{\min}$	$\Psi(\phi_3, \phi_1)$
3	$x_{\min} \leq 0 \leq x_{\max} \wedge 0 \leq y_{\min}$	$\Psi(\phi_3, \phi_2)$
4	$x_{\max} \leq 0 \wedge 0 \leq y_{\min}$	$\Psi(\phi_0, \phi_2)$
5	$x_{\max} \leq 0 \wedge y_{\min} \leq 0 \leq y_{\max}$	$\Psi(\phi_0, \phi_3)$
6	$x_{\max} \leq 0 \wedge y_{\max} \leq 0$	$\Psi(\phi_1, \phi_3)$
7	$x_{\min} \leq 0 \leq x_{\max} \wedge y_{\max} \leq 0$	$\Psi(\phi_1, \phi_0)$
8	$0 \leq x_{\min} \wedge y_{\max} \leq 0$	$\Psi(\phi_2, \phi_0)$

Table 6.1: The set of centre coordinates of all circles in B is an axis-parallel rectangle. This table lists the 9 different cases, considered in this section, for the rectangle's position relative to the x' and y' -axis. These cases are illustrated in figure 6.7. This table also lists for each case the value of set Θ_B , which is the set of all values of circle parameter θ for which there exists a circle with the centre inside the rectangle.

case	$r_{k,\text{low}}(\theta)$	$r_{k,\text{up}}(\theta)$	for θ interval(s)
0	0	$y_{\text{max}}/\sin(\beta + \theta)$	$\Psi(\phi_0, \phi_1)$
0	0	$x_{\text{min}}/\cos(\beta + \theta)$	$\Psi(\phi_1, \phi_2)$
0	0	$y_{\text{min}}/\sin(\beta + \theta)$	$\Psi(\phi_2, \phi_3)$
0	0	$x_{\text{max}}/\cos(\beta + \theta)$	$\Psi(\phi_3, \phi_0)$
.....			
1	$x_{\text{min}}/\cos(\beta + \theta)$	$y_{\text{max}}/\sin(\beta + \theta)$	$\Psi(\phi_0, \phi_1)$
1	$x_{\text{min}}/\cos(\beta + \theta)$	$y_{\text{min}}/\sin(\beta + \theta)$	$\Psi(\phi_2, \phi_3)$
1	$x_{\text{min}}/\cos(\beta + \theta)$	$x_{\text{max}}/\cos(\beta + \theta)$	$\Psi(\phi_3, \phi_0)$
.....			
2	$x_{\text{min}}/\cos(\beta + \theta)$	$y_{\text{max}}/\sin(\beta + \theta)$	$\begin{cases} \Psi(\phi_0, \phi_1) & \text{if } \phi_0 > \phi_2 \\ \Psi(\phi_2, \phi_1) & \text{if } \phi_0 \leq \phi_2 \end{cases}$
2	$x_{\text{min}}/\cos(\beta + \theta)$	$x_{\text{max}}/\cos(\beta + \theta)$	$\Psi(\phi_2, \phi_0)$ if $\phi_0 > \phi_2$
2	$y_{\text{min}}/\sin(\beta + \theta)$	$y_{\text{max}}/\sin(\beta + \theta)$	$\Psi(\phi_0, \phi_2)$ if $\phi_0 \leq \phi_2$
2	$y_{\text{min}}/\sin(\beta + \theta)$	$x_{\text{max}}/\cos(\beta + \theta)$	$\begin{cases} \Psi(\phi_3, \phi_0) & \text{if } \phi_0 \leq \phi_2 \\ \Psi(\phi_3, \phi_2) & \text{if } \phi_0 > \phi_2 \end{cases}$
.....			
3	$y_{\text{min}}/\sin(\beta + \theta)$	$y_{\text{max}}/\sin(\beta + \theta)$	$\Psi(\phi_0, \phi_1)$
3	$y_{\text{min}}/\sin(\beta + \theta)$	$x_{\text{min}}/\cos(\beta + \theta)$	$\Psi(\phi_1, \phi_2)$
3	$y_{\text{min}}/\sin(\beta + \theta)$	$x_{\text{max}}/\cos(\beta + \theta)$	$\Psi(\phi_3, \phi_0)$
.....			
4	$x_{\text{max}}/\cos(\beta + \theta)$	$y_{\text{max}}/\sin(\beta + \theta)$	$\begin{cases} \Psi(\phi_0, \phi_1) & \text{if } \phi_1 \leq \phi_3 \\ \Psi(\phi_0, \phi_3) & \text{if } \phi_1 > \phi_3 \end{cases}$
4	$x_{\text{max}}/\cos(\beta + \theta)$	$x_{\text{min}}/\cos(\beta + \theta)$	$\Psi(\phi_1, \phi_3)$ if $\phi_1 \leq \phi_3$
4	$y_{\text{min}}/\sin(\beta + \theta)$	$y_{\text{max}}/\sin(\beta + \theta)$	$\Psi(\phi_3, \phi_1)$ if $\phi_1 > \phi_3$
4	$y_{\text{min}}/\sin(\beta + \theta)$	$x_{\text{min}}/\cos(\beta + \theta)$	$\begin{cases} \Psi(\phi_3, \phi_2) & \text{if } \phi_1 \leq \phi_3 \\ \Psi(\phi_1, \phi_2) & \text{if } \phi_1 > \phi_3 \end{cases}$
.....			
5	$x_{\text{max}}/\cos(\beta + \theta)$	$y_{\text{max}}/\sin(\beta + \theta)$	$\Psi(\phi_0, \phi_1)$
5	$x_{\text{max}}/\cos(\beta + \theta)$	$x_{\text{min}}/\cos(\beta + \theta)$	$\Psi(\phi_1, \phi_2)$
5	$x_{\text{max}}/\cos(\beta + \theta)$	$y_{\text{min}}/\sin(\beta + \theta)$	$\Psi(\phi_2, \phi_3)$
.....			
6	$y_{\text{max}}/\sin(\beta + \theta)$	$x_{\text{min}}/\cos(\beta + \theta)$	$\begin{cases} \Psi(\phi_1, \phi_2) & \text{if } \phi_2 \leq \phi_0 \\ \Psi(\phi_1, \phi_0) & \text{if } \phi_2 > \phi_0 \end{cases}$
6	$y_{\text{max}}/\sin(\beta + \theta)$	$y_{\text{min}}/\sin(\beta + \theta)$	$\Psi(\phi_2, \phi_0)$ if $\phi_2 \leq \phi_0$
6	$x_{\text{max}}/\cos(\beta + \theta)$	$x_{\text{min}}/\cos(\beta + \theta)$	$\Psi(\phi_0, \phi_2)$ if $\phi_2 > \phi_0$
6	$x_{\text{max}}/\cos(\beta + \theta)$	$y_{\text{min}}/\sin(\beta + \theta)$	$\begin{cases} \Psi(\phi_2, \phi_3) & \text{if } \phi_2 > \phi_0 \\ \Psi(\phi_0, \phi_3) & \text{if } \phi_2 \leq \phi_0 \end{cases}$
.....			
7	$y_{\text{max}}/\sin(\beta + \theta)$	$x_{\text{min}}/\cos(\beta + \theta)$	$\Psi(\phi_1, \phi_2)$
7	$y_{\text{max}}/\sin(\beta + \theta)$	$y_{\text{min}}/\sin(\beta + \theta)$	$\Psi(\phi_2, \phi_3)$
7	$y_{\text{max}}/\sin(\beta + \theta)$	$x_{\text{max}}/\cos(\beta + \theta)$	$\Psi(\phi_3, \phi_0)$
.....			
8	$x_{\text{min}}/\cos(\beta + \theta)$	$y_{\text{min}}/\sin(\beta + \theta)$	$\begin{cases} \Psi(\phi_2, \phi_3) & \text{if } \phi_3 \leq \phi_1 \\ \Psi(\phi_2, \phi_1) & \text{if } \phi_3 > \phi_1 \end{cases}$
8	$x_{\text{min}}/\cos(\beta + \theta)$	$x_{\text{max}}/\cos(\beta + \theta)$	$\Psi(\phi_3, \phi_1)$ if $\phi_3 \leq \phi_1$
8	$y_{\text{max}}/\sin(\beta + \theta)$	$y_{\text{min}}/\sin(\beta + \theta)$	$\Psi(\phi_1, \phi_3)$ if $\phi_3 > \phi_1$
8	$y_{\text{max}}/\sin(\beta + \theta)$	$x_{\text{max}}/\cos(\beta + \theta)$	$\begin{cases} \Psi(\phi_3, \phi_0) & \text{if } \phi_3 > \phi_1 \\ \Psi(\phi_1, \phi_0) & \text{if } \phi_3 \leq \phi_1 \end{cases}$

Table 6.2: This table lists for each of the cases in table 6.1, the values of r_k 's lower bound $r_{k,\text{low}}(\theta)$ and r_k 's upper bound $r_{k,\text{up}}(\theta)$ for specific intervals for circle parameter θ .

Splitting up the integral. Using table 6.2, the integral of equation 6.32 can be rewritten to a sum in which each term has one of two forms, either

$$\frac{1}{2\pi} \int_{r_a - r_{\max}}^{r_a - r_{\min}} p(d) \int_{\Theta(d) \cap [\theta_{\min}, \theta_{\max}]} r'_{k, \max}(d, \theta) d\theta dd \quad (6.35)$$

for some interval $[\theta_{\min}, \theta_{\max}]$ or

$$- \frac{1}{2\pi} \int_{r_a - r_{\max}}^{r_a - r_{\min}} p(d) \int_{\Theta(d) \cap [\theta_{\min}, \theta_{\max}]} r'_{k, \min}(d, \theta) d\theta dd \quad (6.36)$$

for some interval $[\theta_{\min}, \theta_{\max}]$. In each term of the form of equation 6.35 we can rewrite $r'_{k, \max}(d, \theta)$ to the expression of equation 6.34, which contains $r_{k, \text{up}}(\theta)$. The $[\theta_{\min}, \theta_{\max}]$ of such a term can be chosen such that there exists a b such that either for each $\theta \in [\theta_{\min}, \theta_{\max}]$ we have $r_{k, \text{up}}(\theta) = b/\sin(\beta + \theta)$, or for each $\theta \in [\theta_{\min}, \theta_{\max}]$ we have $r_{k, \text{up}}(\theta) = b/\cos(\beta + \theta)$ (see table 6.2). Similarly in each term of the form of equation 6.36 we can rewrite $r'_{k, \min}(d, \theta)$ to the expression of equation 6.33, which contains $r_{k, \text{low}}(\theta)$. The $[\theta_{\min}, \theta_{\max}]$ of such a term can be chosen such that there exists a b such that either for each $\theta \in [\theta_{\min}, \theta_{\max}]$ we have $r_{k, \text{low}}(\theta) = b/\sin(\beta + \theta)$, or for each $\theta \in [\theta_{\min}, \theta_{\max}]$ we have $r_{k, \text{low}}(\theta) = b/\cos(\beta + \theta)$. As an example of how equation 6.32 can be split up into terms as described above, consider case 2 (see figure 6.7 and table 6.2) with $\phi_0 > \phi_2$ and $\beta = 0$, for which equation 6.32 can be rewritten to:

$$\begin{aligned} \int_{A \cap B} p(z_p | k) dk &= \frac{1}{2\pi} \int_{r_a - r_{\max}}^{r_a - r_{\min}} p(d) \int_{\Theta(d) \cap [\phi_0, \phi_1]} r'_{k, \max}(d, \theta) d\theta dd \\ &+ \frac{1}{2\pi} \int_{r_a - r_{\max}}^{r_a - r_{\min}} p(d) \int_{\Theta(d) \cap [\phi_2, \phi_0]} r'_{k, \max}(d, \theta) d\theta dd \\ &+ \frac{1}{2\pi} \int_{r_a - r_{\max}}^{r_a - r_{\min}} p(d) \int_{\Theta(d) \cap [\phi_3, \phi_2]} r'_{k, \max}(d, \theta) d\theta dd \\ &- \frac{1}{2\pi} \int_{r_a - r_{\max}}^{r_a - r_{\min}} p(d) \int_{\Theta(d) \cap [\phi_0, \phi_1]} r'_{k, \min}(d, \theta) d\theta dd \\ &- \frac{1}{2\pi} \int_{r_a - r_{\max}}^{r_a - r_{\min}} p(d) \int_{\Theta(d) \cap [\phi_2, \phi_0]} r'_{k, \min}(d, \theta) d\theta dd \\ &- \frac{1}{2\pi} \int_{r_a - r_{\max}}^{r_a - r_{\min}} p(d) \int_{\Theta(d) \cap [\phi_3, \phi_2]} r'_{k, \min}(d, \theta) d\theta dd, \end{aligned}$$

where for $\theta \in [\phi_0, \phi_1]$ we have $r_{k, \text{low}}(\theta) = x_{\min}/\cos(\beta + \theta)$ and $r_{k, \text{up}}(\theta) = y_{\max}/\sin(\beta + \theta)$, for $\theta \in [\phi_2, \phi_0]$ we have $r_{k, \text{low}}(\theta) = x_{\min}/\cos(\beta + \theta)$ and $r_{k, \text{up}}(\theta) = x_{\max}/\cos(\beta + \theta)$, for $\theta \in [\phi_3, \phi_2]$ we have $r_{k, \text{low}}(\theta) = y_{\min}/\sin(\beta + \theta)$ and $r_{k, \text{up}}(\theta) = x_{\max}/\cos(\beta + \theta)$.

Let us now consider finding a solution for integrals of the form of equation 6.35 (integrals of the form of equation 6.36 can be solved in a way similar to the following approach). Similar to our approach in section 6.6.2, we want to solve the inner integral by finding the antiderivative of its integrand. Depending on the value of θ and d this integrand $r'_{k, \max}(d, \theta)$ is either $r_{k, \text{up}}(\theta)$, $d + R_{\max}$, $d + R_{\min}$, $r_{k, \max}(d, \theta)$ or $r_{k, \min}(d, \theta)$ (see equation 6.34). These different functions obviously have different antiderivatives. We therefore split up the integral of equation 6.35 into integrals over intervals of d and θ for which $r'_{k, \max}(d, \theta)$ is either $r_{k, \text{up}}(\theta)$, $d + R_{\max}$, $d + R_{\min}$, $r_{k, \max}(d, \theta)$ or $r_{k, \min}(d, \theta)$ for each d, θ value in such interval. In order to do that, we introduce $d\theta$ -interval sets.

We will use so-called $d\theta$ -interval sets as a representation for subspaces of the two-dimensional real number space. The name $d\theta$ -interval set is chosen because we apply this to the two dimensional space of the d and θ parameters discussed in this section.

Definition 10 ($d\theta$ -interval set). A $d\theta$ -interval set is a subset of $((\overline{\mathbb{R}} \times \overline{\mathbb{R}}) \times \mathbb{P}([\mathbb{R} \rightarrow \overline{\mathbb{R}}] \times [\mathbb{R} \rightarrow \overline{\mathbb{R}}]))$, where $\overline{\mathbb{R}}$ are the affinely extended real numbers (i.e. $\overline{\mathbb{R}} = \mathbb{R} \cup \{-\infty, \infty\}$). Let

$Y = \{(\delta_1, \Omega_1), (\delta_2, \Omega_2), \dots, (\delta_n, \Omega_n)\}$ be such a $d\theta$ -interval set with for each i , $1 \leq i \leq n$, $\delta_i = (\delta_i^{(1)}, \delta_i^{(2)})$ and $\Omega_i = \{(\omega_{i,1}^{(1)}, \omega_{i,1}^{(2)}), \dots, (\omega_{i,m(i)}^{(1)}, \omega_{i,m(i)}^{(2)})\}$ ($m(i)$ is used to indicate the number of elements of Ω_i). In its closed-interval interpretation, an element (δ_i, Ω_i) of this $d\theta$ -interval set represents the set of all elements $(d, \theta) \in \mathbb{R} \times \mathbb{R}$ for which $d \in [\delta_i^{(1)}, \delta_i^{(2)}]$ and $\theta \in [\omega_{i,j}^{(1)}(d), \omega_{i,j}^{(2)}(d)]$ for some j (note that the interval for θ is a function of d). The complete $d\theta$ -interval set represents the union of all the subspaces of $\mathbb{R} \times \mathbb{R}$ represented by its elements. Function C maps a $d\theta$ -interval set to the subset of $\mathbb{R} \times \mathbb{R}$ corresponding to its closed-interval interpretation:

$$C(Y) = \{(d, \theta) \in \mathbb{R} \times \mathbb{R} : (\exists i, j : 1 \leq i \leq n \wedge 1 \leq j \leq m(i) : d \in [\delta_i^{(1)}, \delta_i^{(2)}] \wedge \theta \in [\omega_{i,j}^{(1)}(d), \omega_{i,j}^{(2)}(d)])\}$$

We require all elements in a $d\theta$ -interval set to be disjoint. Two elements (δ_i, Ω_i) and (δ_j, Ω_j) are disjoint if and only if the Lebesgue measure of $C(\{(\delta_i, \Omega_i)\}) \cap C(\{(\delta_j, \Omega_j)\})$ equals 0. Furthermore, we require for each element $((\delta_i^{(1)}, \delta_i^{(2)}), \Omega_i)$ of a $d\theta$ -interval set and for each pair of different elements $(\omega_{i,a}^{(1)}, \omega_{i,a}^{(2)})$ and $(\omega_{i,b}^{(1)}, \omega_{i,b}^{(2)})$ of such Ω_i , that for each value $d \in [\delta_i^{(1)}, \delta_i^{(2)}]$ the Lebesgue measure of $[\omega_{i,a}^{(1)}(d), \omega_{i,a}^{(2)}(d)] \cap [\omega_{i,b}^{(1)}(d), \omega_{i,b}^{(2)}(d)]$ equals 0.

In this section's application of $d\theta$ -interval sets we only consider values in range $[-\pi, \pi]$ for the second dimension (i.e. circle parameter θ) of the two-dimensional space.

Let $\{f_1, \dots, f_n\}$ be a set of n functions such that for each i , $1 \leq i \leq n$, $f_i: \mathbb{R} \times \mathbb{R} \rightarrow \mathbb{R}$ and there exists a function $F_i: \mathbb{R} \times \mathbb{R} \rightarrow \mathbb{R}$ such that for each d, θ we have $f_i(d, \theta) = \partial F_i(d, \theta) / \partial \theta$. Let $g: \mathbb{R} \times \mathbb{R} \rightarrow \mathbb{R}$ be a function such that for each d and θ , there exists an i , $1 \leq i \leq n$, such that:

$$g(d, \theta) = f_i(d, \theta)$$

Let for each i , $1 \leq i \leq n$, Y_i be a $d\theta$ -interval set such that: (i) for each $(d, \theta) \in C(Y_i)$ we have $g(d, \theta) = f_i(d, \theta)$, (ii) we have $C(Y_1) \cup C(Y_2) \cup \dots \cup C(Y_n) = \mathbb{R} \times \mathbb{R}$, (iii) for each j , with $1 \leq j \leq n$ and $j \neq i$, the Lebesgue measure of $C(Y_i) \cap C(Y_j)$ equals 0. Then for some lower bound d_a and upper bound d_b for d , and some lower bound $\theta_a(d)$ and upper bound $\theta_b(d)$ for θ as functions of d , we have:

$$\int_{d_a}^{d_b} p(d) \int_{\theta_a(d)}^{\theta_b(d)} g(d, \theta) d\theta dd = \sum_{i=1}^n \sum_{(\delta, \Omega) \in Y_i} \int_{[\delta^{(1)}, \delta^{(2)}] \cap [d_a, d_b]} \left(p(d) \sum_{(\omega_a, \omega_b) \in \Omega} \int_{[\omega_a(d), \omega_b(d)] \cap [\theta_a(d), \theta_b(d)]} f_i(d, \theta) d\theta \right) dd \quad (6.37)$$

and by applying lemma 5:

$$\int_{d_a}^{d_b} p(d) \int_{\theta_a(d)}^{\theta_b(d)} g(d, \theta) d\theta dd = \sum_{i=1}^n \sum_{(\delta, \Omega) \in Y_i} \int_{[\delta^{(1)}, \delta^{(2)}] \cap [d_a, d_b]} p(d) \sum_{(\omega_a, \omega_b) \in \Omega} (F_i(d, v) - F_i(d, u)) dd \quad (6.38)$$

where $u = \min(\max(\omega_a(d), \theta_a(d)), \omega_b(d))$ and $v = \max(\min(\omega_b(d), \theta_b(d)), \omega_a(d))$.

Equation 6.38 is a solution for integrals as in equation 6.35, by choosing $d_a = r_a - r_{\max}$, $d_b = r_a - r_{\min}$, $[\theta_a(d), \theta_b(d)] = \Theta(d) \cap [\theta_{\min}, \theta_{\max}]$, $g(d, \theta) = r'_{k, \max}(d, \theta)$ and:

$$\begin{aligned} f_1(d, \theta) &= r_{k, \text{up}}(\theta) \\ f_2(d, \theta) &= d + R_{\max} \\ f_3(d, \theta) &= d + R_{\min} \\ f_4(d, \theta) &= r_{k, \text{max}}(d, \theta) \\ f_5(d, \theta) &= r_{k, \text{min}}(d, \theta) \end{aligned}$$

Observe that for each case and θ -interval in table 6.2, there exists a positive value a and some value δ with $-\pi \leq \delta \leq \pi$ such that $r_{k, \text{up}}(\theta) = a / \sin(\theta + \delta)$. Given this a and δ , the antiderivatives

of f_1, \dots, f_n are:

$$\begin{aligned} F_1(d, \theta) &= a \ln \left(\tan \left(\frac{\theta + \delta}{2} \right) \right) \\ F_2(d, \theta) &= (d + R_{\max})\theta \\ F_3(d, \theta) &= (d + R_{\min})\theta \\ F_4(d, \theta) &= f_{\max}(d, \theta) \quad (\text{see equation 6.26}) \\ F_5(d, \theta) &= \begin{cases} 0 & \text{if } d \leq -r_a - r_{\min} \\ f_{\min}(d, \theta) & \text{otherwise (see equation 6.26)} \end{cases} \end{aligned}$$

Now in order to evaluate equation 6.38, the $d\theta$ -interval sets Y_1, \dots, Y_n have to be known. For the derivation of Y_1, \dots, Y_n we will make use of the following definition and lemma.

Definition 11. We define \otimes as the binary operator on $d\theta$ -interval sets, of which the result is another $d\theta$ -interval set, with the following property for each pair Y_a, Y_b of $d\theta$ -interval sets:

$$\forall (d, \theta) \in \mathbb{R} \times \mathbb{R} : (d, \theta) \in C(Y_a \otimes Y_b) \iff (d, \theta) \in C(Y_a) \cap C(Y_b)$$

Lemma 6. Let f_1, \dots, f_n be a set of n functions such that for each i , with $1 \leq i \leq n$, $f_i: \mathbb{R} \times \mathbb{R} \rightarrow \mathbb{R}$. Let g_1, \dots, g_n a set of functions such that for each i , with $1 \leq i \leq n$, function $g_i: \mathbb{R} \times \mathbb{R} \rightarrow \mathbb{R}$ is defined as follows:

$$g_i(d, \theta) = \begin{cases} h_{i-1}(g_{i-1}(d, \theta), f_i(d, \theta)) & \text{if } i > 1 \\ f_1(d, \theta) & \text{if } i = 1 \end{cases}$$

where for each $j < i$:

$$h_{i-1}(f_j(d, \theta), f_i(d, \theta)) = f_k(d, \theta) \quad \text{where } k = i \text{ or } k = j$$

Furthermore we define for each i and j , with $1 \leq i < n$ and $1 \leq j \leq i$, the sets $\beta_{i,j}$ and $\alpha_{i,j}$ to be some $d\theta$ -interval sets satisfying: (i) for each $(d, \theta) \in \mathbb{R} \times \mathbb{R}$, either $(d, \theta) \in C(\beta_{i,j})$ and $h_i(f_j(d, \theta), f_{i+1}(d, \theta)) = f_j(d, \theta)$ or $(d, \theta) \in C(\alpha_{i,j})$ and $h_i(f_j(d, \theta), f_{i+1}(d, \theta)) = f_{i+1}(d, \theta)$, (ii) the Lebesgue measure of $C(\alpha_{i,j}) \cap C(\beta_{i,j})$ equals 0. We use symbol $Y_{i,j}$ for i and j , with $1 \leq i \leq n$ and $1 \leq j \leq i$, for a $d\theta$ -interval set for which $g_i(d, \theta) = f_j(d, \theta)$ for each $(d, \theta) \in C(Y_{i,j})$. We furthermore require that for each i , with $1 \leq i \leq n$, we have $C(Y_{i,1}) \cup C(Y_{i,2}) \cup \dots \cup C(Y_{i,i}) = \mathbb{R} \times \mathbb{R}$ and that for each j_1 and j_2 , with $1 \leq j_1, j_2 \leq i$ and $j_1 \neq j_2$, the Lebesgue measure of $C(Y_{i,j_1}) \cap C(Y_{i,j_2})$ equals 0. Then $Y_{i,j}$ can be computed by induction as follows:

$$Y_{i,j} = \begin{cases} Y_{i-1,j} \otimes \beta_{i-1,j} & \text{if } j < i \\ \bigcup_{k=1}^{i-1} Y_{i-1,k} \otimes \alpha_{i-1,k} & \text{if } j = i \\ \{((-\infty, \infty), \{(d \mapsto -\infty, d \mapsto \infty)\})\} & \text{if } i = 1 \end{cases} \quad (6.39)$$

Now if we choose

$$h_1 = \min, \quad h_2 = \max, \quad h_3 = \min, \quad h_4 = \max$$

then $g = g_5 = r'_{k,\max}$ and $d\theta$ -intervals sets Y_1, Y_2, \dots, Y_5 are equal to $Y_{5,1}, Y_{5,2}, \dots, Y_{5,5}$. For our choice of f_1, \dots, f_n and h_1, \dots, h_4 , appendix C lists the $\beta_{i,j}$ sets as defined in lemma 6. An $\alpha_{i,j}$ set as defined in lemma 6 can be seen as a kind of complement of $\beta_{i,j}$. The $\alpha_{i,j}$ sets can easily be derived from the corresponding $\beta_{i,j}$ sets listed in appendix C and are therefore not listed in this thesis. Using the $\beta_{i,j}$ and $\alpha_{i,j}$ sets, the $d\theta$ -interval sets Y_1, Y_2, \dots, Y_5 can be computed by following equation 6.39.

Summary of the approach. Let us summarize the approach presented in this section. We adapt the formulation of the integral of equation 6.25 over the circles in A such that it integrates over $A \cap B$ (see equation 6.32). This adaptation includes the introduction of $r_{k,\text{low}}(\theta)$ and $r_{k,\text{up}}(\theta)$,

which is the lower bound and upper bound respectively as a function of θ for the r_k parameter of all circles in B . Table 6.2 lists for each of the 9 considered cases of B 's position relative to the x' and y' -axis, intervals for circle parameter θ and for each of these intervals a corresponding closed-form expression for $r_{k,\text{low}}(\theta)$ and $r_{k,\text{up}}(\theta)$. The integral of equation 6.32 is split into integrals of which each corresponds to a θ interval of table 6.2 and is of the same form as equation 6.35 or 6.36. As result, we only have integrals in which $r_{k,\text{low}}(\theta)$ and $r_{k,\text{up}}(\theta)$ can be written as a single closed-form expression. Depending on the value of θ and d , the integrand $r'_{k,\text{max}}(d, \theta)$ of equation 6.35 is either $r_{k,\text{up}}(\theta)$, $d + R_{\text{max}}$, $d + R_{\text{min}}$, $r_{k,\text{max}}(d, \theta)$ or $r_{k,\text{min}}(d, \theta)$ (see equation 6.34). These different functions obviously have different antiderivatives. We therefore split up the integral of equation 6.35 into integrals over intervals of d and θ for which $r'_{k,\text{max}}(d, \theta)$ is either $r_{k,\text{up}}(\theta)$, $d + R_{\text{max}}$, $d + R_{\text{min}}$, $r_{k,\text{max}}(d, \theta)$ or $r_{k,\text{min}}(d, \theta)$ for each d, θ value in such interval. These intervals are represented by the $d\theta$ -interval sets Y_1, \dots, Y_5 , which can be computed by following lemma 6. Given Y_1, \dots, Y_5 and the known antiderivatives of $r_{k,\text{up}}(\theta)$, $d + R_{\text{max}}$, $d + R_{\text{min}}$, $r_{k,\text{max}}(d, \theta)$ and $r_{k,\text{min}}(d, \theta)$, equation 6.38 provides a solution for equation 6.35. This solution only contains a one-dimensional integral over circle parameter d , which (as the solution found in section 6.6.2) can be numerically evaluated by iterating over the d bins of the histogram illustrated in figure 4.30. In a similar way we can solve integrals of the form of equation 6.36, and therefore the presented approach gives us a solution that can be used for the evaluation the integral over $A \cap B$ as in equations 6.31 and 6.32.

6.6.4 Integration over $A \cap B \cap V$

We will now show how an approximation of the visibility space V only requires a few simple modifications to the approach presented in section 6.6.3 in order to evaluate equation 6.16:

$$\int_{A \cap B \cap V} p(z_p | k) dk$$

Recall the definition of the visibility space V from section 6.5: space V is the set of circles that intersect the region that can be observed by a particular layer of the laser scanner. The region that can be observed by the laser scanner is defined by the position of the laser scanner, its field of view, a minimum distance and a maximum distance.

We approximate V by the set V' . Let k be a circle and p be the point on k that is closest to the point measurement z_p . Then circle k is in V' if and only if p is in the region that can be observed by the laser scanner.

Now let us first have a look at the differences between V and V' . Observe that $V' \subseteq V$: if a circle k is in V' then there exists a point on k (namely the point on k closest to z_p) that is in the region that can be observed by the laser scanner, and therefore circle k intersects with the region that can be observed by the laser scanner, which means that k is also in V . Circles that are in V but not in V' are those circles that intersect the region that can be observed by the laser scanner, but of which the closest point to z_p is not within this observable region. Note from equation 6.13 and figure 4.30, that $p(z_p | k) = 0$ if the distance from z_p to its k closest point p is more than 0.4 m. This means that if point measurement z_p is more than 0.4 m from the border of the region that can be observed by the laser scanner, then using the approximation V' gives the same results as V :

$$\int_{A \cap B \cap V'} p(z_p | k) dk = \int_{A \cap B \cap V} p(z_p | k) dk$$

How well V is approximated by V' has not been investigated further since for the verification method's analysis in section 6.7 we can focus on point measurements that are more than 0.4 m away from the border of the observable region.

Let us now turn to the modifications that are necessary to the method of section 6.6.3 for the computation of equation 6.16 where V is approximated by V' . Note that for a circle k , point p is defined relative to z_p by the circle's d and θ parameters. Observe that for a given d we need at most five intervals to describe those values for θ for which p is inside the observable region (five

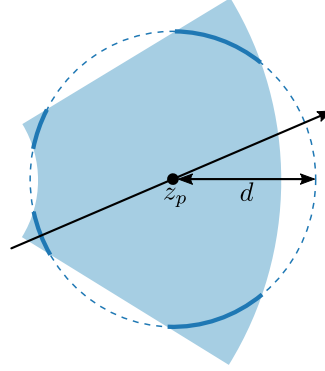


Figure 6.8: The bright blue area is the region that can be observed by the laser scanner and the black line is the line passing through z_p and the centre of the annulus that is to the right of point z_p . The illustrated circle consists of those points for which the distance to z_p is d . The dashed part are those points that are outside the observable region. The solid part are those points that are inside the observable region. Let $\theta \in [-\pi, \pi]$ be the angle between the vector from z_p to a point p on the circle and the vector between z_p and the centre of the annulus. Then the values of θ for which point p is inside the observable region, can be described by five intervals as illustrated in this figure. Note that the solid part of the circle in the lower left corner of the observable region is described by two intervals: one ending at π and one starting at $-\pi$.

instead of four since we only allow θ values of the interval $[-\pi, \pi]$, as illustrated in figure 6.8. Let $\Theta_{V,1}(d), \dots, \Theta_{V,5}(d)$ be these five intervals. Now equation 6.32 for the integral over $A \cap B$ can be modified to the integral over $A \cap B \cap V'$:

$$\int_{A \cap B \cap V'} p(z_p | k) dk = \sum_{j=1}^5 \frac{1}{2\pi} \int_{r_a - r_{\max}}^{r_a - r_{\min}} p(d) \int_{\Theta'(d) \cap \Theta_{V,j}(d)} (r'_{k,\max}(d, \theta) - r'_{k,\min}(d, \theta)) d\theta dd \quad (6.40)$$

Similar to equation 6.32, each of the terms in this equation can be rewritten to the form of equation 6.38 which represents a solution for the integral.

6.7 Analysis of the Method

This section presents an analysis of the presented map verification method for roundabout central islands. The analysis shows the results are not satisfactory for the intended goal of the method. It is also shown that the causes of these unsatisfactory results are not specific to the method's application for verifying central islands, but to the verification method in general. One of the causes of this unsatisfactory result can be traced back to the assumption of conditionally independent measurements given the correctness of the map, but, as we will see, this is not the only cause.

6.7.1 Analysis of the Central Island Implementation

We will now have a look at what the output of the central island map verification method is for a certain scenario and different measurement values. Instead of only evaluating the final output, we also look at the intermediate measurement probability densities that were computed by the method in order to get a better understanding of the final output.

In the scenario used for this evaluation, the digital map describes a central island by an annulus with centre x -coordinate 40 m, centre y -coordinate 0 m, inner radius $r_{\min} = 11.8$ m and outer radius $r_{\max} = 12.2$ m. The minimum radius R_{\min} and maximum radius R_{\max} of a central island that can be detected by the central island detector are set to 7 m and 18 m respectively. The sensor is located at the origin, it points in the x -direction, has a field of view of 84° , a minimum range of 20 m and a maximum range of 50 m.

The probability density distributions for the circle-centre measurement and for the point measurement's distance to the circle border were generated from the histograms in figure 4.27 and 4.30, which were obtained by the central island detector's evaluation.

Recall equation 6.12 for the probability density of a measurement z given the correctness c of a map:

$$p(z | c) = \begin{cases} p(O = \text{true} | c) \cdot \gamma_{A \cap V} \cdot \int_{A \cap V} p(z | x_z) dx_z \\ \quad + p(O = \text{false} | c) \cdot \gamma_{V \setminus A} \cdot \int_{V \setminus A} p(z | x_z) dx_z & \text{if } c = \text{true} \\ \gamma_{V \setminus A} \cdot \int_{V \setminus A} p(z | x_z) dx_z & \text{if } c = \text{false} \end{cases}$$

where the integral

$$\gamma_{A \cap V} \cdot \int_{A \cap V} p(z | x_z) dx_z \quad (6.41)$$

represents the probability density of measurement z given the fact that the measured element (a circle in this case) is inside visibility space V and inside annulus A , and integral

$$\gamma_{V \setminus A} \cdot \int_{V \setminus A} p(z | x_z) dx_z \quad (6.42)$$

represents the probability density of measurement z given the fact that the measured circle is inside visibility space V and outside annulus A .

For the verification of the map description of a central island, the measurement z was chosen to consist of two measurements: a point measurement z_p and a circle-centre measurement z_c . Before we evaluate the probability density distributions for the combination of measurements z_p and z_c , we first have a look at the distribution for the point measurement z_p (i.e. equations 6.41 and 6.42 where z is substituted by z_p).

Figure 6.9 shows the distribution for z_p given that the measured circle is not in annulus A . One can clearly recognize the shape of the sensor's visibility region, which makes sense since outside this region it is impossible to measure a point and therefore the probability density is zero and inside this region there will always be a set of circles for which the measured point has a non-zero probability density.

Figure 6.10 shows the point measurement's probability density for points near the annulus. Note that compared to figure 6.9 the range of the colour map has been adjusted in order to visualize the small drop in the density for points near the annulus. The reason for this density drop can be explained as follows: the closer the measured point is to the annulus, the more circles for which the point has a non-zero probability density are inside the annulus and those circles will not contribute to the integral of equation 6.41. The drop in the probability density near the annulus is relatively small, since the number of circles inside the annulus is a lot smaller than the number of circles that are outside the annulus and inside the visibility region.

Figure 6.11 shows the point measurement's probability density distribution given that the measured circle is inside annulus A . As one would expect, this is an almost rotational symmetric distribution with the annulus' centre as the symmetry axis. It is not entirely rotational symmetric, since the point measurements outside the visibility region have a zero probability density. The shape of this distribution is better visualized by figure 6.12 which shows the probability density as a function of z_p 's x -coordinate where the y -coordinate is zero.

Let us now focus on the probability density distribution of a complete measurement z which consists of a point measurement z_p and a circle centre measurement z_c . Since it is hard to visualize a four-dimensional probability space, we plot the probability density as a function of the point measurement's coordinates for a fixed value of the measured circle centre.

Figure 6.13 shows the probability density of a measurement given that the origin of the measurement is outside annulus A , for a measured centre with value (40.0 m, 0.0 m). The illustration shows a disc-shape that is a bit more stretched in the x -direction than in the y -direction, because a centre measurement (with y -coordinate equal to 0.0 m) has a greater uncertainty in the x -direction than in the y -direction (see figure 4.27). The position of this disc shape depends on

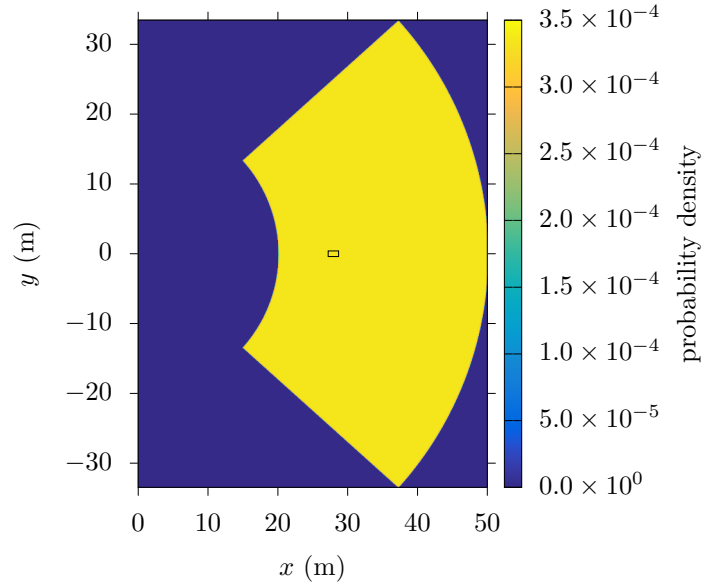


Figure 6.9: The probability density of the point measurement z_p given that the measurement's origin is not in A : $\gamma_{V \setminus A} \cdot \int_{V \setminus A} p(z_p | x_z) dx_z$. In this image, the x and y -coordinates represent the x and y -coordinates of the measured point z_p and the colour represents the probability density for those measured coordinates. The small rectangle marks the region that is illustrated in figure 6.10 with an adjusted colour scale.

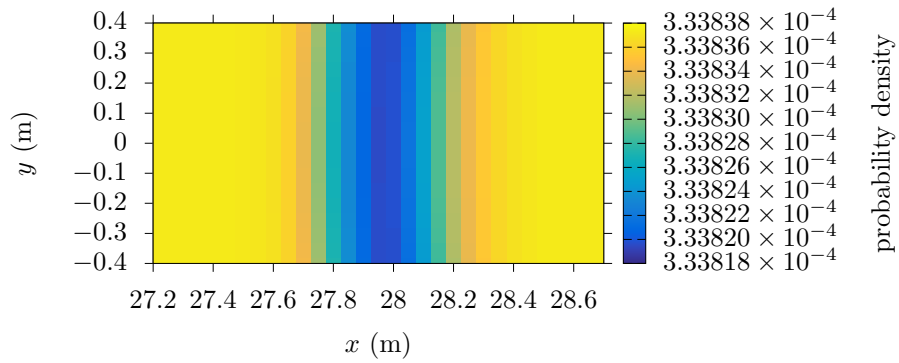


Figure 6.10: The probability density of the point measurement z_p given that the measurement's origin is not in A : $\gamma_{V \setminus A} \cdot \int_{V \setminus A} p(z_p | x_z) dx_z$. In this image, the x and y -coordinates represent the x and y -coordinates of the measured point z_p and the colour represents the probability density for those measured coordinates. Shown here is the area near the annulus, which in this image is between x -coordinates 27.8 m and 28.2 m.

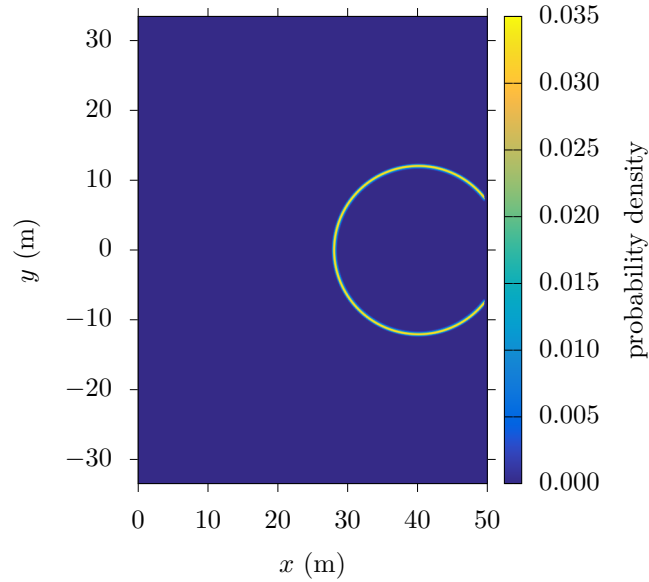


Figure 6.11: The probability density of the point measurement z_p given that the measurement's origin is in A : $\gamma_{A \cap V} \cdot \int_{A \cap V} p(z_p | x_z) dx_z$. In this image, the x and y -coordinates represent the x and y -coordinates of the measured point z_p and the colour represents the probability density for those measured coordinates.

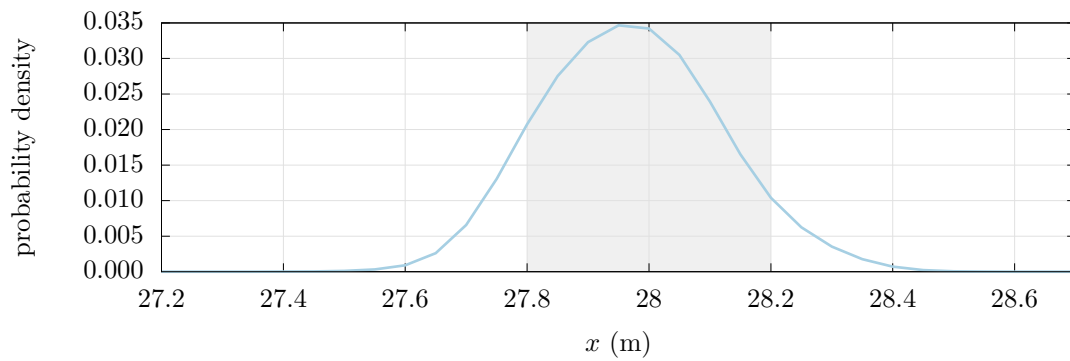


Figure 6.12: The probability density of the point measurement z_p given that the measurement's origin is in A , as a function of z_p 's x -coordinate: $\gamma_{V \cap A} \cdot \int_{V \cap A} p(z_p | x_z) dx_z$, where z_p 's y -coordinate is zero. The grey shaded area corresponds to the x -coordinates that are within the annulus.

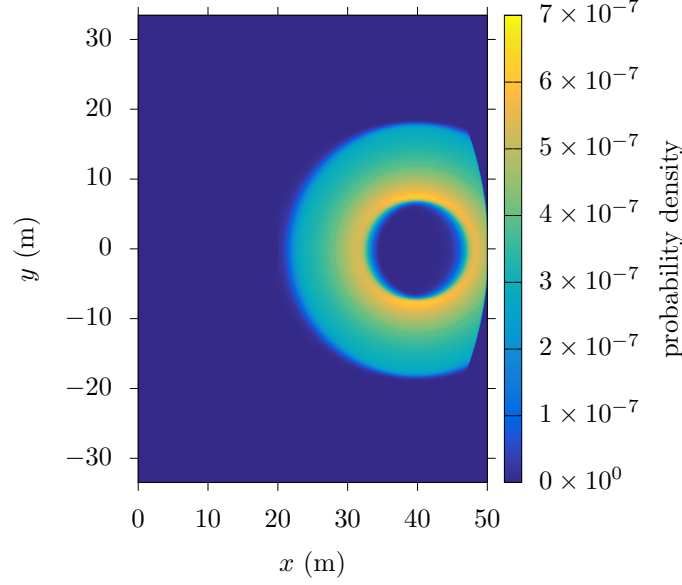


Figure 6.13: The probability density of the measurement $z = (z_p, z_c)$ given that the measurement's origin is not in A , as a function of z_p : $\gamma_{V \setminus A} \cdot \int_{V \setminus A} p(z_p, z_c | x_z) dx_z$, where $z_c = (40.0 \text{ m}, 0.0 \text{ m})$. In this image, the x and y -coordinates represent the x and y -coordinates of the measured point z_p and the colour represents the probability density for those measured coordinates.

the actual measured centre z_c , which can also be seen in figure 6.14 where for different values of z_c 's x -coordinate the probability density is plotted as function of z_p 's x -coordinate where z_p 's y -coordinate is zero. Around $x = 28 \text{ m}$, which is the area around the annulus, there is a small dip in three of the distributions. These dips can be explained by the fact that for those (z_p, z_c) values there exist circles inside A for which the probability density of measurement values z_p and z_c is non-zero. For the centre measurements with 37 m or 38 m as its x -coordinate there is no such dip, because for such centre measurements there is no circle inside annulus A for which the centre z_c measurement has a non-zero probability density.

Figure 6.15 shows the probability density of a measurement given that the origin of the measurement is inside annulus A , for a measured centre with value $(40.0 \text{ m}, 0.0 \text{ m})$. As one would expect, the probability density is high for points inside and near the annulus and lower for points further removed. Figure 6.16 shows for different values of z_c 's x -coordinate, the probability density as a function of z_p 's x -coordinate, where z_p 's y -coordinate is zero. It shows that the probability density is lower if the measured centre z_c is further away from the annulus' centre, which makes sense since in such case the probability density of the centre measurement will be lower for each origin circle inside the annulus.

Above we analysed the probability density distributions given by equations 6.41 and 6.42. Once these probability densities have been computed for a measurement, they can be used to compute the probability density $p(z | c)$ of measurement z given the correctness c of the map by applying equation 6.12. In order to compute $p(z | c)$, we also need to know the probability to detect the central island described in the map given that the map is correct, $p(O = \text{true} | C = \text{true})$. In the following example we chose this probability to be 0.3; choosing a different value will have no significant influence on the outcomes in the sense that it would change the conclusions drawn in this chapter. The computed value of $p(z | c)$ together with the a priori map correctness probability $p(c | m_{0:t-1})$ can now be used to compute the a posteriori map correctness probability $p(c | m_{0:t})$ by applying equation 6.4. For the following example we assume an a priori probability of 0.5 for the map being correct.

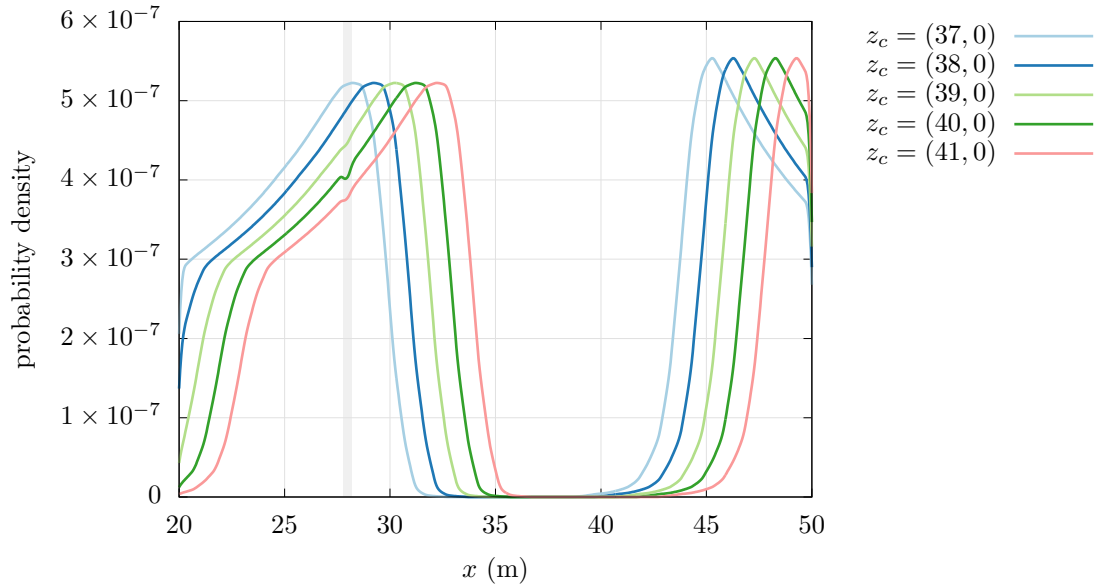


Figure 6.14: For different values of the measured centre z_c , the probability density of the measurement $z = (z_p, z_c)$ given that the measurement's origin is not in A , as a function of z_p 's x -coordinate: $\gamma_{V \setminus A} \cdot \int_{V \setminus A} p(z_p, z_c | x_z) dx_z$, where z_p 's y -coordinate is zero. The grey shaded area corresponds to the x -coordinates that are within the annulus.

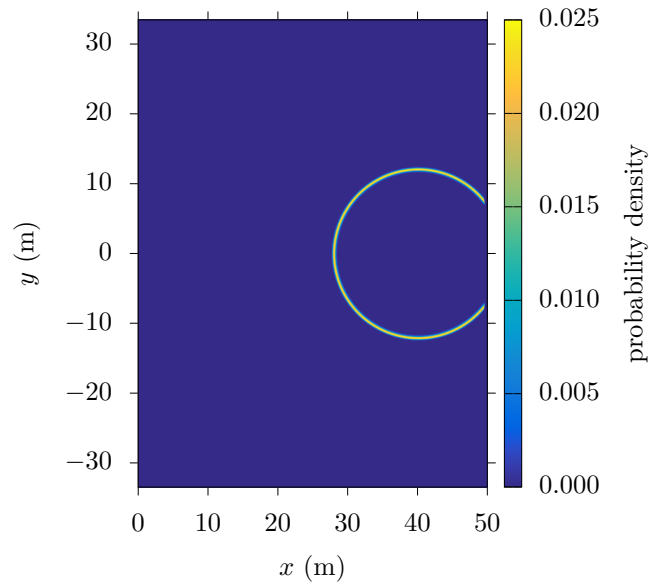


Figure 6.15: The probability density of the measurement $z = (z_p, z_c)$ given that the measurement's origin is in A , as a function of z_p : $\gamma_{V \cap A} \cdot \int_{V \cap A} p(z_p, z_c | x_z) dx_z$, where $z_c = (40.0 \text{ m}, 0.0 \text{ m})$. In this image, the x and y -coordinates represent the x and y -coordinates of the measured point z_p and the colour represents the probability density for those measured coordinates.

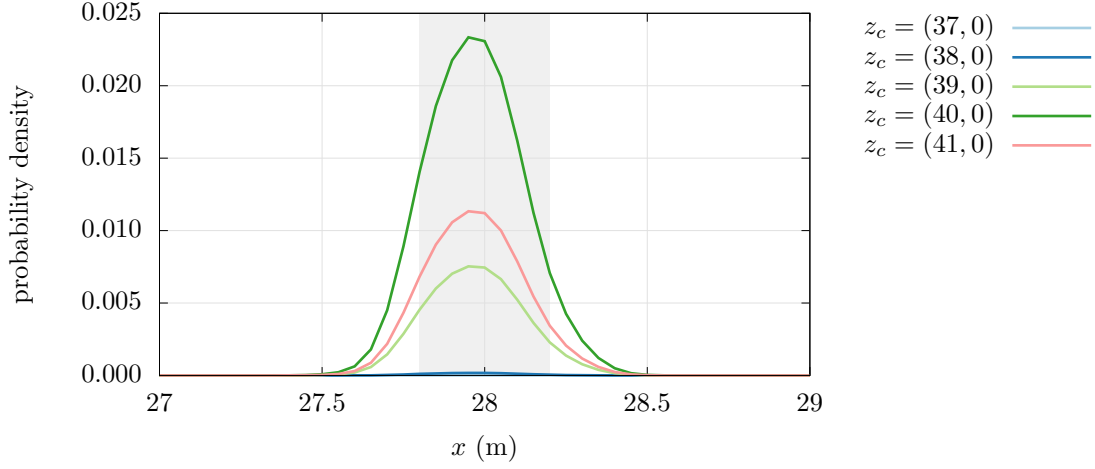


Figure 6.16: For different values of the measured centre z_c , the probability density of the measurement $z = (z_p, z_c)$ given that the measurement's origin is in A , as a function of z_p 's x -coordinate: $\gamma_{V \cap A} \cdot \int_{V \cap A} p(z_p, z_c | x_z) dx_z$, where z_p 's y -coordinate is zero. The grey shaded area corresponds to the x -coordinates that are within the annulus.

Figure 6.17 shows the a posteriori probability of the map being correct $p(C = \text{true} | m_{0:t})$, as a function of the coordinates of the measured point z_p , where the circle centre measurement z_c has coordinates (40.0 m, 0.0 m) (i.e. the same z_c value as used for figures 6.13 and 6.15). Note that figure 6.17 does not visualise a probability density function, but instead it shows the conditional probability of C being true (i.e. the map being correct) given all measurements done thus far $m_{0:t}$, as a function of its last point measurement z_p . One may notice that for point measurements in a large part of the visibility region, the a posteriori probability is 0.5. This is because within this region both the measurement probability density given that the origin is outside the annulus (equation 6.42) and the measurement probability density given that the origin is inside the annulus (equation 6.41), is exactly zero. In such case the measurement's probability density is the same for the case where the map is correct and the case where it is incorrect, and therefore such a measurement will not increase or decrease the a priori probability 0.5 of the map being correct. We furthermore see that another large region of point measurements causes the a posteriori probability to be approximately 0.4. This is because in that region the measurement's probability density, given that its origin is inside the annulus, is exactly zero (i.e. the first term for the case $c = \text{true}$ in equation 6.12). Consequently the probability density $p(z | C = \text{true})$ of the measurement, given that the map is correct, is exactly $p(O = \text{false} | C = \text{true}) = 0.7$ times the probability density $p(z | C = \text{false})$ of the measurement given that the map is incorrect. Plugging this into equation 6.4 in combination with an a priori probability of 0.5, gives us indeed a posteriori map correctness probability of approximately 0.4.

As can be expected, figure 6.17 shows that the a posteriori probability $p(C = \text{true} | m_{0:t})$ is higher the closer the point measurement is to the annulus. Figure 6.18 shows for z_p with y -coordinate equal to zero and the same z_c values as used in figures 6.14 and 6.16, the a posteriori probability as a function of the point measurement's x -coordinate. We see that for different values of z_c , the shape of the a posteriori probability as a function of z_p is roughly the same, except when z_c is too far from the annulus' centre in which case the a posteriori probability is a constant function of z_p . One can see that if a point measurement is made within the annulus, that the a priori probability will be a value very close to 1. This is not only the case for points inside the annulus, but also for points that are 20 cm or less away from the annulus' border. As can be seen in figure 6.16, the probability density of a point measurement at $x < 27.5$ m, given that the measurement's origin circle is in the annulus, is very low. However for the case

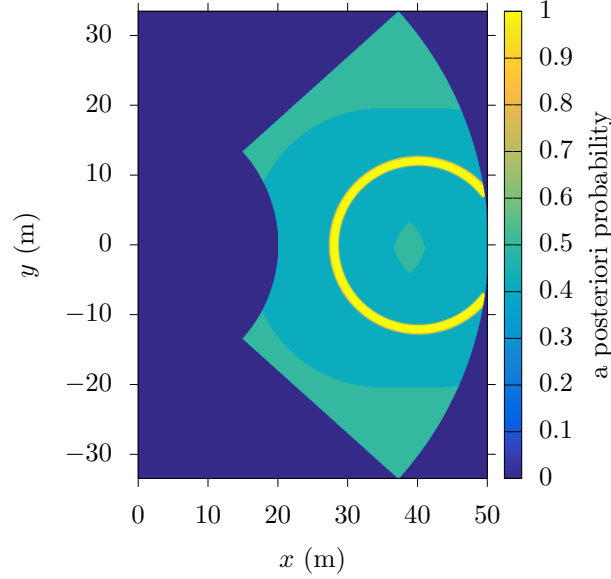


Figure 6.17: For a priori probability for the map being correct $p(C = \text{true} \mid m_{0:t-1}) = 0.5$ and latest centre measurement $z_c = (40.0 \text{ m}, 0.0 \text{ m})$, this figure shows the a posteriori probability $p(C = \text{true} \mid m_{0:t})$ as a function of the latest point measurement's coordinates z_p .

$z_c = (40.0 \text{ m}, 0.0 \text{ m})$, the a posteriori probability for the map being correct has increased compared to its a priori probability for point measurements with an x value between 27.2 m and 27.5 m, which is also the case for other a priori probabilities as can be observed in figure 6.19. Although this may be surprising, it still makes sense because the probability density of a measurement given that the map is incorrect is always several orders of magnitude lower than the highest probability density for a measurement given that the map is correct (as can be concluded from figures 6.13 and 6.15), and the a posteriori probability of the map being correct is only lower than its a priori value if $p(z \mid C = \text{true}) < p(z \mid C = \text{false})$.

6.7.2 Evaluation

As discussed in the previous subsection, the a posteriori probability of the map being correct quickly approaches 1 after only one measurement near the annulus. Probabilistically this makes sense and can be explained as follows. Let x_1 be a circle inside the annulus with a certain probability density $p(z \mid X_z = x_1)$ for the measurement z , given that x_1 was the origin of this measurement. Now let x_2 be a circle outside the annulus such that $p(z \mid X_z = x_1) = p(z \mid X_z = x_2)$. So that means that if the map would be known to be been incorrect and the measurement origin known to be x_2 , then the probability density for measurement value z is equal to its probability density in the case the map is known to be correct and the measurement origin is known to be x_1 . However the probability density $p(X_z = x_2 \mid C = \text{false})$ for x_2 being the measurement origin in case of an incorrect map, is much smaller than the probability density $p(X_z = x_1 \mid C = \text{true})$ for x_1 being the origin in case of a correct map. This is because we modelled each circle outside the annulus equally likely to be measured, and each circle inside the annulus equally likely to be measured, and the Lebesgue measure of the set of circles outside the annulus is larger compared to the set of circles inside the annulus (in simple language: there are “more” circles outside the annulus than inside the annulus). Consequently the contribution of $p(X_z = x_2 \mid C = \text{false}) \cdot p(z \mid X_z = x_2)$ to the integral of equation 6.42 will be much smaller than the contribution of $p(X_z = x_1 \mid C = \text{true}) \cdot p(z \mid X_z = x_1)$ to the integral of equation 6.41. As a

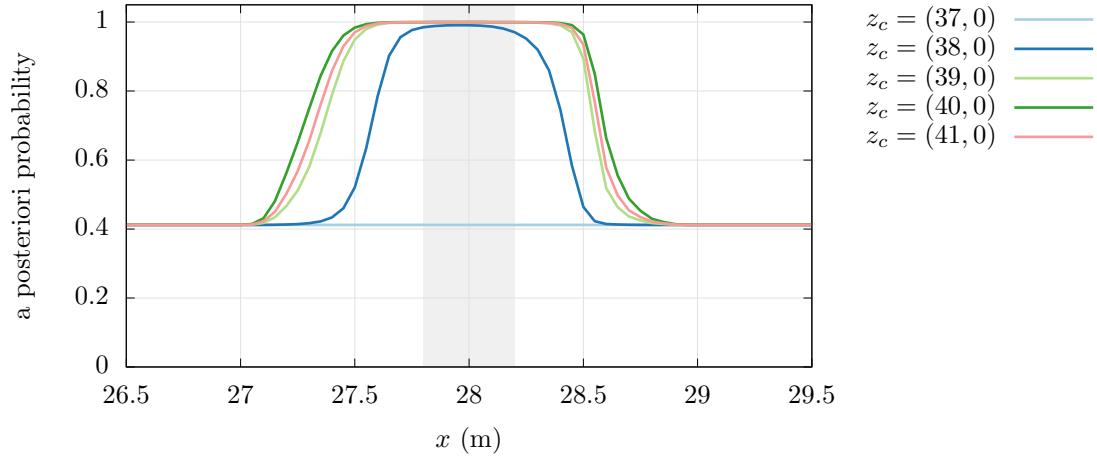


Figure 6.18: For different values of latest centre measurement z_c , the a priori probability for the map being correct $p(C = \text{true} \mid m_{0:t-1}) = 0.5$ and latest point measurement z_p having y -coordinate equal to zero, this figure shows the a posteriori probability $p(C = \text{true} \mid m_{0:t})$ as a function of z_p 's x -coordinate. The grey shaded area corresponds to the x -coordinates that are within the annulus.

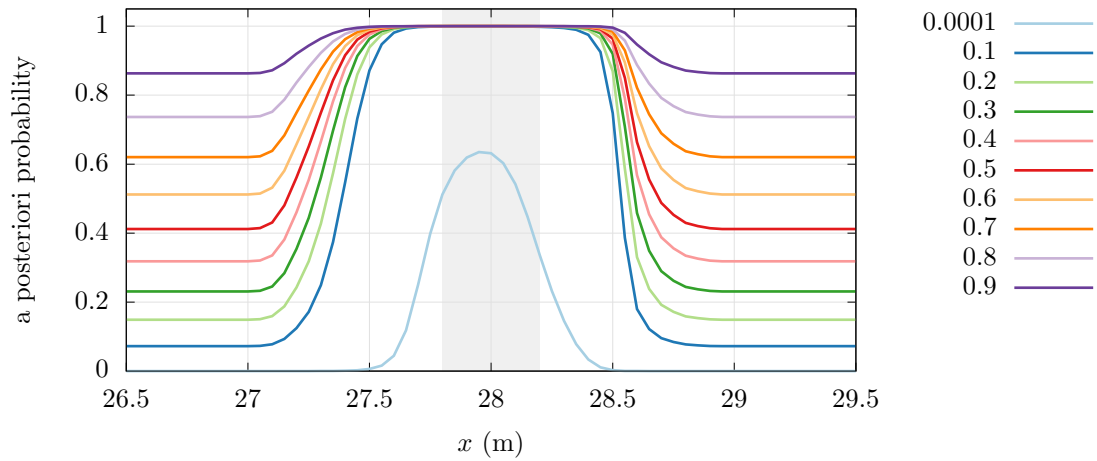


Figure 6.19: For different values of the a priori probability for the map being correct $p(C = \text{true} \mid m_{0:t-1})$, latest centre measurement $z_c = (40.0 \text{ m}, 0.0 \text{ m})$ and latest point measurement z_p having y -coordinate equal to zero, this figure shows the a posteriori probability $p(C = \text{true} \mid m_{0:t})$ as a function of z_p 's x -coordinate. The grey shaded area corresponds to the x -coordinates that are within the annulus.

result, measurements that may not seem very probable in case of a correct map, may still cause the a posteriori probability for the map being correct to be higher than its a priori probability. Although probabilistically justifiable at first sight, we should question whether it is desirable to have such a drastic increase in the probability of the map being correct.

What if a measurement is false/noise? Then also in such case the a posteriori probability will be close to 1. However one should realize that probability theory will never give guarantees about a hypothesis being true. Even if you are very certain about a hypothesis being true because you found the probability for that to be nearly 1, there is still a chance that the hypothesis is false. This is also the case with false/noise measurements: their probability to occur is very low. The verification model can also handle noise cases: despite the fact that the probability jumps to nearly 1 after the noise measurement near the annulus, all following measurements being far away from the annulus will make the probability decrease again (although this decrease is at a lower rate than the increase to 1, as can be concluded from figure 6.19).

Let us consider the case where the map is incorrect and where the real-world central island has the same centre as the annulus but a radius of 12.7 m (recall that in our running example the outer radius of the annulus is 12.2 m). In that case most point measurements will be around 50 cm away from the annulus' border, which will be around $x = 27.3$ m in figure 6.19. As concluded in the previous subsection, after such a measurement the probability of the map being correct will have been increased. Unlike the short increase of the probability after a noise measurement, which eventually will decrease, in this case the probability will not decrease afterwards because most following measurements will have a similar distance to the annulus' border. So after a number of measurements the probability of the map being correct will be nearly 1 and will remain nearly 1. This result is undesirable for two reasons. First, it is not very unlikely that an infrastructure element is modified and, in the case of a roundabout, that the central island gets a bigger radius. We therefore want to be able to detect such an infrastructure modification that makes the map incorrect. Second, if we keep getting measurements that are about 50 cm away from the annulus' border, then it is likely that we are indeed measuring a real-world circle that has its border 50 cm away from the annulus' border and that therefore the map is probably incorrect. As we saw before, quite the opposite is true for the proposed verification method as such measurements are always in favour of hypothesis that the map is correct. It can therefore be concluded that the results of the implemented verification method are unsatisfactory.

6.7.3 Causes of Unsatisfactory Results

Section 6.7.2 identified two problems: the rapid increase of the map-correctness probability after just a single measurement and the problem illustrated by the incorrect-map example, where a repeated number of measurements that seem more likely to support an incorrect-map hypothesis still make the map-correctness probability increase. The former problem can be explained by the lack of a priori knowledge we have about the measured element. The latter problem can be attributed to the assumption of conditional independence between the measurements. Both causes are discussed in more detail below.

Lack of a priori knowledge about the measured element As already indicated in the previous subsection, the main cause of a dramatic increase of the map-correctness probability after just a single measurement, is the fact that we consider each element outside A to be equally likely to be measured and each element inside A equally likely to be measured. As mentioned in section 6.4 the choice for a uniform distribution (for X_r) motivated by the principle of indifference, may be debatable. However, even if one would choose a non-uniform probability density distribution, then this still has to be a wide spread distribution, because we have no further knowledge about the parameter values of the element that was measured. Furthermore for the verification of roundabout central islands, the set of visible elements not in A is much larger than the set of visible elements in A and thus the probability densities for the former will still be a lot less than for the latter. This means that choosing a different probability density distribution for the parameters of X_r , which is “fair” considering the lack of knowledge we have about the parameters of the measured element,

will not solve the problem of the rapid increase of the map-correctness probability after just a single measurement. The root cause seems to be the lack of knowledge about the value for the parameters of the measured elements and the inability to express this lack of knowledge in a probabilistic way. This cause is in fact independent of the assumption of conditionally independent measurements of equation 6.3, since it also causes a rapid increase of the map-correctness probability after the first measurement (since in that case there is only measurement, the assumption cannot be violated). Nevertheless, as we will see, the assumption has an effect for all measurements following the first one.

Conditional measurement independence assumption Let us have a look at the effect of the conditionally independent measurement assumption of equation 6.3, by considering the case where this conditional independence does not hold. Then equation 6.4 for the a posteriori map correctness probability becomes

$$p(c \mid m_{0:t}) = \eta_t \cdot p(z_t \mid c, m_{0:t-1}) \cdot p(c \mid m_{0:t-1})$$

Here $p(z_t \mid c, m_{0:t-1})$ can be rewritten in a similar way as in section 6.4 while considering the dependence of the measurements. Equation 6.7 then becomes:

$$p(z \mid C = \text{true}, O = \text{true}, m_{0:t-1}) = \int_{A \cap V} p(z \mid x_z, C = \text{true}, O = \text{true}, m_{0:t-1}) \cdot p(x_z \mid C = \text{true}, O = \text{true}, m_{0:t-1}) dx_z$$

Note we still assume that the measurement z is conditionally independent of the previous measurements $m_{0:t-1}$ given the measurement's origin x_z (see equation 6.2). Therefore the above equation can be rewritten to:

$$p(z \mid C = \text{true}, O = \text{true}, m_{0:t-1}) = \int_{A \cap V} p(z \mid x_z) \cdot p(x_z \mid C = \text{true}, O = \text{true}, m_{0:t-1}) dx_z \quad (6.43)$$

The difference between this equation and equation 6.7 for $p(z \mid C = \text{true}, O = \text{true})$, is that in the latter the second factor of the integrand is $p(x_z \mid C = \text{true}, O = \text{true})$ instead of $p(x_z \mid C = \text{true}, O = \text{true}, m_{0:t-1})$. This factor is factored out of the integral as a constant ($\gamma_{A \cap V}$) which results in equation 6.8, because the probability density for each element inside $A \cap V$ is assumed equal. However in the equation above, $p(x_z \mid C = \text{true}, O = \text{true}, m_{0:t-1})$ cannot be factored out of the integral, because the distribution density distribution of the measured element X_z is not uniform within $A \cap V$. This is because the measurements $M_{0:t-1}$ and measured element X_z are conditionally dependent and therefore $p(x_z \mid C = \text{true}, O = \text{true}, m_{0:t-1})$ differs from $p(x_z \mid C = \text{true}, O = \text{true})$.

Similar to above, when we do not assume the conditional independence of measurements as in equation 6.3, equation 6.9 becomes

$$p(z \mid C = \text{true}, O = \text{false}, m_{0:t-1}) = \int_{V \setminus A} p(z \mid x_z) \cdot p(x_z \mid C = \text{true}, O = \text{false}, m_{0:t-1}) dx_z \quad (6.44)$$

and equation 6.11 becomes

$$p(z \mid C = \text{false}, m_{0:t-1}) = \int_{V \setminus A} p(z \mid x_z) \cdot p(x_z \mid C = \text{false}, m_{0:t-1}) dx_z \quad (6.45)$$

Also here, when comparing equations 6.44 and 6.45 with equations 6.10 and 6.11 respectively, we see that in the latter two the second factor was factored out of the integral as a constant.

Let us return to our invalid-map example of subsection 6.7.2, where after a whole series of measurements 50 cm away from the annulus' border, the a posteriori correctness probability counter-intuitively becomes nearly one. We considered such result counter-intuitive, since our intuition says that if we keep getting measurements 50 cm away from the annulus' border, then most likely

we are measuring a circle outside the annulus and therefore the map is most likely incorrect. This intuition is confirmed by the equations above. For example in equation 6.45, after the first measurement $p(x_z | C = \text{false}, m_{0:t-1})$ will be higher for values of x_z that are “near” the first measurement than for values of x_z that are “further away” from the first measurement. As a consequence, in our example $p(z | C = \text{false}, m_{0:t-1})$ will be larger than $p(z | C = \text{false})$ for a measurement that is approximately 50 cm away from the annulus’ border. This means that a second measurement 50 cm from the annulus’ border in case of an incorrect map is much more likely because the first measurement was similar.

From the discussion above it can be concluded that the conditional measurement independence assumption of equation 6.3, is the main cause of the problem presented by the incorrect-map example in subsection 6.7.2.

6.7.4 Why the problems are hard to solve

We will now discuss some of the difficulties that may arise if we would like to adapt the presented verification model such that it does not assume the conditional independence of measurements as in equation 6.3.

Suppose we want to evaluate the measurement probability density value as in equation 6.43. Unlike equation 6.8 where the integrand is a single probability density function, equation 6.43 is an integral of which the integrand is a product of two probability density functions. Since it is much harder to come up with an analytical solution (or a partial analytical solution) for such an integral, in most cases the evaluation of integral 6.43 requires a numerical computation. Since generally the parameter space of the elements to be verified is multi-dimensional, such a numerical computation will be very costly in terms of computation time. Similar arguments hold for the evaluation of equations 6.44 and 6.45. This makes a verification method based on these equations less usable in practice.

Let us turn to another problem that arises when we want to consider the conditional dependence between measurements. Observe that in each of the equations 6.43, 6.44 and 6.45, the second factor is a function of the sequence of all previous measurements $m_{0:t-1}$. This does not necessarily mean that the computation of these second factors has to have a computational complexity proportional to the number of previous measurements. For example, let us consider the evaluation of $p(x_z | C = \text{true}, O = \text{true}, m_{0:t-1})$, which is the probability density for value x_z for the parameters of the measured real-world element that is described by the correct map given all previous measurements $m_{0:t-1}$. It may be possible to formulate this probability density recursively as a function of the second last measurement z_{t-1} and the probability density $p(x_z | C = \text{true}, O = \text{true}, m_{0:t-2})$ given all previous measurements except the second last one. Instead of an algorithm that computes the probability density with a computational complexity proportional to the number of previous measurements, such recursive formulation can be used to construct an algorithm that incrementally computes the probability density with a constant complexity in terms of the number of previous measurements. The downside of such an algorithm, however, is that it must not only maintain the map correctness probability $p(C = \text{true} | m_{0:t})$ given all measurements, but also the probability density distribution for the real-world element inside A given that the map is correct. In that case the algorithm is in fact explicitly estimating the state of the environment element (only for the case of a correct map though), something which we wanted to avoid (see the motivation for map verification in section 6.1).

Apart from the reasons for avoiding explicit estimation of the real-world element’s parameters that motivate the need for a priori map data, there is another problem with explicit estimation. Consider the factors $p(x_z | C = \text{true}, O = \text{false})$ and $p(x_z | C = \text{false})$ in equations 6.44 and 6.45 respectively. These are the probability densities of the parameters for the real-world element that is the origin of the latest measurement for two different cases in which this origin must be outside A (the case where the map is correct but the origin is not the mapped real-world element and the case of an incorrect map). Since there can be multiple real-world elements in the environment, it is possible that the current measurement has a different origin than the previous measurement. Consequently, instead of recursively estimating the parameters of a single environment element,

the algorithm has to estimate the set of elements in the environment. What makes such a task far from trivial, is the fact that one does not know in advance the number of environment elements and that it requires a solution for the association of measurements with real-world elements. Although this is not an impossible problem to solve in specific cases, it is hard to come up with a general solution that does not introduce a dramatic increase of the problem's complexity. Furthermore such a verification method will differ little or nothing from a method that estimates the parameters of the environment elements and that computes the probability of the map being correct based on this estimation. Again the absence of such an estimation method is one of the motivations for map verification.

From the above can be concluded that it is far from trivial to come up with a probabilistic verification model that does not assume the conditional measurement independence of equation 6.3 and the question can be asked whether, due to lack of knowledge about the real world, probability theory offers us a suitable framework to practically solve the map verification problem at all.

6.8 Conclusions and Recommendations for Future Work

One of the contributions of this chapter is the discussion of the motivation for map verification, possible ways to cope with incorrect map data, and the requirements on a map verification system derived from this discussion. The list of requirements is by no means meant to be complete, but should be seen as an initial attempt to specify the new field of research of verifying a priori map data for highly automated driving. Future research has to be done on how the automated vehicle system copes with blind spots in its perception as the result of incorrect map data. Specifically, research should focus on the development of behaviour planning algorithms that can generate safe behaviour in case of blind spots. This kind of research can help in making the requirements on a map verification system more specific.

This chapter also investigated a probabilistic model for map verification that assumes sensor measurements to be conditionally independent given the correctness of the map. The advantage of such an approach would be that the map verification problem can be solved by means of recursive state estimation (where the map's correctness is the state to be estimated). From the analysis, based on an implementation of the model for the verification of roundabout central islands, it can be concluded that the model does not provide satisfactory results.

One unsatisfactory effect that was observed, is the rapid increase of the probability that the central island is correctly described by the map after only a single measurement that leaves the hypothesis of an incorrect map is still plausible. This is mainly caused by the lack of a priori knowledge we have about the measurement in case of an incorrect map.

A second problem that was observed, is that in some cases of an incorrect map, the probability that the map is correct increases during a series of measurements that seem more likely to support an incorrect-map hypothesis. This problem can be attributed to the assumption of conditional independence between the measurements. Adapting the proposed verification model such that the conditional dependence between measurements is taken into account will give rise to three new problems. First, the computation of the map correctness probability involves computing integrals of the product of two functions which in practice will be hard to solve analytically. Thus such a computation will be computationally expensive. Second, the resulting verification model performs an estimation of the parameters of the environment elements that were observed by the sensors. The lack of perception algorithms to estimate the parameters of an environment element is the main motivation for using a priori map data and the verification of map data. It will therefore be very unlikely that a map verification method will be practically useful if it is based on parameter estimation. The third problem, is that such parameter estimation requires the system to be able to distinguish different environment elements as origins of measurements and to correctly associate a new measurement to existing measurements of the same environment element. It will be very unlikely that this can be done for environment elements for which we want to use an a priori map for the perception (i.e. those environment elements for which no sensor-based perception can be used). Overall it can be concluded that adapting the verification model such that the dependence

between measurements is considered, leads to a practically unusable map verification model.

Future research can consider other ways to model the verification problem probabilistically, where the biggest challenge will be to model our lack of a priori knowledge about measurements in case of an incorrect map. Alternatively, non-probabilistic models can be formulated to solve the map verification problem. The main challenge will then be to develop a measure for the output of the verification system to express the uncertainty about the correctness of the map, that allows a proper risk analysis by the behaviour planning system.

One of the requirements for a map verification system that was not addressed by the proposed verification model is the consideration of the interdependence between the map verification problem and the map-relative localization problem. Future research should consider this interdependence and a solution may be based on simultaneous localization and verification.

Chapter 7

Conclusions and Recommendations for Future Research

This chapter summarizes the contributions of the thesis, its main conclusions and recommendations for future research. Detailed conclusions for each individual topic can be found in the corresponding chapters.

Multi-Hypothesis Road Representation. Chapter 3 explained that perception of a roadway can be subject to uncertainty about the structure of a roadway and that consequently the perception system can have multiple plausible hypotheses about the roadway. The chapter presented a multi-hypothesis road representation with the goal of eliminating the requirement on the perception system to present only a single roadway hypothesis to the behaviour planning system. Consequently, with this representation the behaviour planning system can consider all plausible (possibly conflicting) hypotheses. The representation allows multiple hypotheses about the roadway and is designed in such a way that it is easy to infer where and how the hypotheses about the roadway differ and where they are the same. Although the representation itself is not a perception method, it can be an enabler for sensor-based perception of more complex roads and therefore contributes to research objective 1: improve the capabilities of sensor-based perception in urban environments using close-to-market sensors. The added value of multiple hypotheses for behaviour planning, still needs to be evaluated, and is currently a subject of ongoing research.

The representation can deal with uncertainty about the structure of the roadway by representing multiple hypotheses, but cannot yet express the degree of uncertainty about each such hypothesis. The development of a mathematical model to express the degree of uncertainty, is a topic for future research.

The representation as it is presented in this thesis, can only represent roadways (i.e. the portion of the road in-between junctions) and no junctions. The reason for this is that sensor-based perception, without depending on a priori maps, will most likely be possible for roadways before it is possible for junctions. As soon as the multi-hypothesis representation proved itself useful (or maybe even necessary) at roadways, future research on extending the representation with the representation of junctions will be an important next step.

Roundabout Central Island Detection. A more ready-to-use contribution to research objective 1 is made by the roundabout central-island detector presented in chapter 4. Sensor-based perception of roundabouts for highly automated driving has not been a topic of research so far and therefore the central-island detector makes a first contribution to it. The detection algorithm is based on the detection of circular arcs in single-layer laser scans and fitting circles to those arcs. It hereby relies on scans from a type of laser scanner that only has a few layers and is a

close-to-production automotive component. Apart from contributing to research objective 1, the detector also makes a contribution to research objective 4: develop methods for the perception of roundabouts.

An extensive evaluation of the detection method was done at six different roundabouts based on a highly accurate ground truth for the roundabout's geometry. The evaluation showed that if a laser scan contains a circular arc corresponding to a central island, then in 98% of the cases the algorithm successfully detected the central island. The detected circles provide an accurate description of the central island's geometry at those locations where it was measured. At other locations the circle can be less accurate caused by errors in the estimation of the circle's radius. Although these results are considered very satisfying, there is still room for improvement.

One possible point of improvement is the laser scanner's capability of measuring central island borders. Not only the performance of the detection algorithm was analysed but also the quality of its input data (i.e. laser scans). These analyses showed that the fraction of laser scans that contain circular arcs corresponding to the central island differs heavily among the different roundabouts. There were even roundabouts where certain recordings, in which the vehicle is approaching the roundabout, contained no arcs corresponding to the central island. Future research is needed to identify the causes for these differences (a few ideas are discussed in section 4.6).

Another point of improvement is the number of circles that are detected but do not correspond to a roundabout central island. The evaluation showed that 22% of the detected circles does not correspond to a central island. It has to be noted, however, that each detection is still a correct detection of a circular arc, which means that as a circle detector the algorithm makes no false detections. Furthermore, chapter 5 shows that by considering all detections during the time the vehicle approaches a roundabout, the position of the central island can be correctly estimated despite false detections. Nevertheless, the reduction of the number of false detections can be useful, especially at those roundabouts where only a few laser scans contain the central island. One suggestion on how this can be done is to verify a detected central island using a camera. Since a detection consists of a sequence of points with a three-dimensional position, the positions of these points can be projected to image coordinates. The challenge is then to develop an image processing algorithm that can verify whether the region around the image coordinates of the detection indeed looks like the curb of a central island.

Roundabout Perception Supported by A Priori Map Data. Although the first steps in roundabout perception have been made with the central-island detector, it is still not possible to perceive the full roundabout based on sensor data only. Therefore, in quest of research objective 4, chapter 5 proposes a roundabout perception method that combines sensor-based observations with a detailed a priori roundabout geometry from a digital map. By this, it also contributes to research objective 2: develop methods to combine sensor-based perception with data from an a priori map in order to perceive all relevant environment elements. Only a part of the roundabout model's parameters are estimated using on-board sensor observations. The complete roundabout geometry relative to the ego-vehicle is computed by aligning a detailed geometrical description from an a priori digital map with the estimated parameters. The proposed method can be considered as a map-relative localization method that does not depend on a precise GNSS or additional landmarks in the map.

The sensor-based observations used by the method of chapter 5 consist of the output of the central-island detector and a state-of-the-art road marking detector that detects the left marking of the lane over which the vehicle approaches the roundabout. The road marking is tracked by a state-of-the-art road marking tracker, and is used to estimate the orientation of the roundabout. The central-island detector of chapter 4 is modified such that it detects central islands with a radius known from the a priori roundabout map. The centre of the detected central island is tracked by a particle filter, and is used for estimation of the roundabout's centre. Based on the analysis of the central-island detector's output, a probabilistic measurement model has been developed that is used by the particle filter. Overall it can be concluded that the presented roundabout perception method shows very promising results for use in automated driving.

Future research could be concerned with a performance analysis of a closed-loop automated driving system that integrates the roundabout perception method. Further analysis on the error characteristics of the central-island detector's output may lead to a more detailed measurement model for tracking the roundabout centre and thereby improve the overall performance. In order to make the system depend less on a priori map data, future research is necessary to extend the proposed system by integrating sensor observations of other environment like, for example, the road curbs splitter islands or the marking of the roundabout's outer border.

Map Verification. If a priori map data is used for environment perception, then the system must know whether the map is correct in order to achieve the high level of system robustness that is required for highly automated driving, hence research objective 3: Formulate and investigate the problem of verifying the correctness of a priori map data before the vehicle acts on it. For this purpose, chapter 6 focusses on the on-board verification of a priori map data using the sensors of the automated vehicle. One of the contributions of this chapter is the discussion about the motivation for map verification and the requirements on a map verification system. The presented list of requirements is by no means meant to be complete, but should be seen as an initial attempt to specify the new field of research of verifying a priori map data for highly automated driving. Not only is the verification of map data a relevant topic for environment perception, but also for the field of behaviour planning. Namely, a behaviour planning system must be able to cope with blind spots in the perception of the environment as the result of incorrect map data. Future research on such behaviour planning systems may result in more specific requirements on map verification. For example, given a certain behaviour planning strategy we can derive how far ahead of the vehicle the map must be verified.

Besides formulating requirements, chapter 6 investigated a probabilistic model for map verification that assumes sensor measurements to be conditionally independent given the correctness or incorrectness of the map. The advantage of such an approach would be that the map verification problem can be solved by means of recursive state estimation (where the map's correctness is the state to be estimated). However, from the analysis based on an implementation of the model for the verification of roundabout central islands, it can be concluded that the model does not provide satisfactory results. An evaluation identified the causes for the unsatisfactory results. Although the presented model did not prove to be a usable solution for map verification, the lessons learned from its evaluation can be useful for future research on map verification models.

Bibliography

- [Ame56] America's Independent Electric Light and Power Companies. Advertisement – Power Companies Build For Your New Electric Living. *Life Magazine*, 40(5):3, January 1956.
- [ARB⁺15] M. Aeberhard, S. Rauch, M. Bahram, G. Tanzmeister, J. Thomas, Y. Pilat, F. Homm, W. Huber, and N. Kaempchen. Experience, Results and Lessons Learned from Automated Driving on Germany's Highways. *IEEE Intelligent Transportation Systems Magazine*, 7(1):42–57, 2015.
- [BCD⁺15] A. Broggi, P. Cerri, S. Debattisti, M. C. Laghi, P. Medici, D. Molinari, M. Panciroli, and A. Prioletti. PROUD – Public Road Urban Driverless-Car Test. *IEEE Transactions on Intelligent Transportation Systems*, 16(6):3508–3519, December 2015.
- [BDF⁺14] K. Bengler, K. Dietmayer, B. Farber, M. Maurer, C. Stiller, and H. Winner. Three Decades of Driver Assistance Systems: Review and Future Perspectives. *IEEE Intelligent Transportation Systems Magazine*, 6(4):6–22, 2014.
- [BGD14] Sebastian Brechtel, Tobias Gindele, and Rudiger Dillmann. Probabilistic decision-making under uncertainty for autonomous driving using continuous POMDPs. In *2014 IEEE 17th International Conference on Intelligent Transportation Systems (ITSC)*, pages 392–399. IEEE, 2014.
- [BHG10] M. Baer, U. Hofmann, and S. Gies. Multiple track 4d-road representation. In *2010 IEEE Intelligent Vehicles Symposium (IV)*, pages 319–324, June 2010.
- [BSU09] Planungshinweise für Stadtstrassen: Knotenpunkte & Kreisverkehre. Technical report, Behörde für Stadtentwicklung und Umwelt, 2009. Freie und Hansestadt Hamburg.
- [BVC00] Nicolas Boichis, Jean-Marc Viglino, and Jean-Pierre Cocquerez. Knowledge based system for the automatic extraction of road intersections from aerial images. *International Archives of Photogrammetry and Remote Sensing*, 33(B3):27–34, 2000.
- [Che10] N. Chernov. *Circular and Linear Regression: Fitting Circles and Lines by Least Squares*. Chapman and Hall/CRC Monographs on Statistics and Applied Probability Series. CRC Press/Taylor & Francis, 2010.
- [CLC05] F. Chausse, J. Laneurit, and R. Chapuis. Vehicle localization on a digital map using particles filtering. pages 243 – 248, June 2005.
- [DH72] Richard O. Duda and Peter E. Hart. Use of the Hough transformation to detect lines and curves in pictures. *Communications of the ACM*, 15(1):1115, January 1972.
- [DKH⁺15] J. Dickmann, J. Klappstein, M. Hahn, M. Muntzinger, N. Appenrodt, C. Brenk, and A. Sailer. Present research activities and future requirements on automotive radar from a car manufacturer's point of view. In *2015 IEEE MTT-S International Conference on Microwaves for Intelligent Mobility (ICMIM)*, pages 1–4, April 2015.

- [DMC90] E. D. Dickmanns, B. Mysliwetz, and T. Christians. An integrated spatio-temporal approach to automatic visual guidance of autonomous vehicles. *IEEE Transactions on Systems, Man, and Cybernetics*, 20(6):1273–1284, November 1990.
- [DRS⁺15] F. Dierkes, M. Raaijmakers, M. T. Schmidt, M. E. Bouzouraa, U. Hofmann, and M. Maurer. Towards a Multi-hypothesis Road Representation for Automated Driving. In *2015 IEEE 18th International Conference on Intelligent Transportation Systems (ITSC)*, pages 2497–2504, Sept 2015.
- [EPBB09] Frank Ekpenyong, Dominic Palmer-Brown, and Allan Brimicombe. Extracting road information from recorded GPS data using snap-drift neural network. *Neurocomputing*, 73(1–3):24–36, December 2009.
- [FCR09] T.R.M. Freitas, A. Coelho, and R.J.F. Rossetti. Improving digital maps through gps data processing. In *2009 IEEE 12th International Conference on Intelligent Transportation Systems (ITSC)*, pages 1–6, Oct 2009.
- [Fen70] R. E. Fenton. Automatic vehicle guidance and control – A state of the art survey. *IEEE Transactions on Vehicular Technology*, 19(1):153–161, February 1970.
- [FHW10] Roundabouts. Technical Report FHWA-SA-10-006, U.S. Department of Transportation – Federal Highway Administration, 2010.
- [For12] Forschungsgesellschaft für Straßen- und Verkehrswesen (FGSV). *RASt - Directives for the Design of Urban Roads*, 2012. FGSV-Nr. 200 E.
- [FPR⁺13] U. Franke, D. Pfeiffer, C. Rabe, C. Knoeppel, M. Enzweiler, F. Stein, and R.G. Hertrich. Making Bertha See. In *2013 IEEE International Conference on Computer Vision Workshops (ICCVW)*, pages 214–221, December 2013.
- [GAA⁺12] Tom M. Gasser, Arzt, Clemens, Ayoubi, Mihiar, Bartels, Arne, Bürkle, Lutz, Eier, Jana, Flemisch, Frank, Häcker, Dirk, Hesse, Tobias, Huber, Werner, Lotz, Christine, Maurer, Markus, Ruth-Schumacher, Simone, Schwarz, Jürgen, and Vogt, Wolfgang, editors. *Rechtsfolgen zunehmender Fahrzeugautomatisierung: gemeinsamer Schlussbericht der Projektgruppe*. Number Heft F 83 in Berichte der Bundesanstalt für Strassenwesen. F, Fahrzeugtechnik. Wirtschaftsverlag NW, Bremerhaven, 2012.
- [GHT⁺16] Franz Gritschneider, Patrick Hatzelmann, Markus Thom, Felix Kunz, and Klaus Dietmayer. Adaptive Learning Based on Guided Exploration for Decision Making at Roundabouts. 2016.
- [Goo16a] Google Self-Driving Car Project Monthly Report. Technical report, X, September 2016.
- [Goo16b] Google Self-Driving Car Project Monthly Report. Technical report, X, March 2016.
- [HGS⁺14] O. Hartmann, M. Gabb, F. Schule, R. Schweiger, and K. Dietmayer. Robust and real-time multi-cue map verification for the road ahead. In *2014 IEEE 17th International Conference on Intelligent Transportation Systems (ITSC)*, pages 894–899, October 2014.
- [HLLR12] Aharon Bar Hillel, Ronen Lerner, Dan Levi, and Guy Raz. Recent progress in road and lane detection: a survey. *Machine Vision and Applications*, 25(3):727–745, February 2012.
- [HP62] V Hough and C Paul. Method and means for recognizing complex patterns, December 18 1962. US Patent 3,069,654.
- [Key21] John Maynard Keynes. *A Treatise on Probability*. Macmillan & Co., 1921.

- [KHB⁺14] Martin Kellner, Ulrich Hofmann, Mohamed Essayed Bouzouraa, Hannah Kasper, and Stephan Neumaier. Laserscanner based road curb feature detection and efficient mapping using local curb descriptions. In *2014 IEEE 17th International Conference on Intelligent Transportation Systems (ITSC)*, pages 2602–2609, Oct 2014.
- [KNW⁺15] F. Kunz, D. Nuss, J. Wiest, H. Deusch, S. Reuter, F. Gritschneider, A. Scheel, M. Stübler, M. Bach, P. Hatzelmann, C. Wild, and K. Dietmayer. Autonomous driving at Ulm University: A modular, robust, and sensor-independent fusion approach. In *2015 IEEE Intelligent Vehicles Symposium (IV)*, pages 666–673, June 2015.
- [LMT07] Jesse Levinson, Michael Montemerlo, and Sebastian Thrun. Map-based precision vehicle localization in urban environments. In *Robotics: Science and Systems*, 2007.
- [MDW04] R. Madhavan and H.F. Durrant-Whyte. Natural landmark-based autonomous vehicle navigation. *Robotics and Autonomous Systems*, 46(2):79 – 95, 2004.
- [MFZ09] Adriano Mancini, Emanuele Frontoni, and Primo Zingaretti. Automatic extraction of urban objects from multi-source aerial data. In *Proc. CMRT09: Object Extraction for 3D City Models, Road Databases and Traffic Monitoring-Concepts, Algorithms and Evaluation*, pages 13–18, 2009.
- [MPF13] M. Muffert, D. Pfeiffer, and U. Franke. A stereo-vision based object tracking approach at roundabouts. *IEEE Intelligent Transportation Systems Magazine*, 5(2):22–32, Summer 2013.
- [MSW10] Norman Mattern, Robin Schubert, and Gerd Wanielik. High-accurate vehicle localization using digital maps and coherency images. In *2010 IEEE Intelligent Vehicles Symposium (IV)*, pages 462–469. IEEE, June 2010.
- [Neu13] Stephan Neumaier. *Räumlich-zeitliche Szeneninterpretation einer videobasierten Umfelderkennung zur Fahrerassistenz*. PhD thesis, TU München, 2013.
- [NPM⁺10] S. Nedeveschi, V. Popescu, T. Marita, R. Danescu, M. Meinecke, M.-A. Obojski, and J. Knaup. Intersection representation enhancement by sensorial data and digital map alignment. In *2010 IEEE International Conference on Intelligent Computer Communication and Processing (ICCP)*, pages 393–400, Aug 2010.
- [NVMdT⁺06] P. Núñez, R. Vázquez-Martín, J.C. del Toro, A. Bandera, and F. Sandoval. Feature extraction from laser scan data based on curvature estimation for mobile robotics. In *2006 IEEE International Conference on Robotics and Automation (ICRA)*, pages 1167–1172, May 2006.
- [OJK⁺16] B. Okumura, M. James, Y. Kanzawa, M. Derry, K. Sakai, T. Nishi, and D. Prokhorov. Challenges in perception and decision making for intelligent automotive vehicles: a case study. *IEEE Transactions on Intelligent Vehicles*, PP(99):1–1, 2016.
- [PRMdPV11] Joshué Pérez Rastelli, Vicente Milanés, Teresa de Pedro, and Ljubo Vlacic. Autonomous driving manoeuvres in urban road traffic environment: a study on roundabouts. *IFAC Proceedings Volumes*, 44(1):13795–13800, January 2011.
- [Ram72] Urs Ramer. An iterative procedure for the polygonal approximation of plane curves. *Computer Graphics and Image Processing*, 1(3):244 – 256, 1972.
- [RB14] Marvin Raaijmakers and Mohamed Essayed Bouzouraa. Circle detection in single-layer laser scans for roundabout perception. In *2014 IEEE 17th International Conference on Intelligent Transportation Systems (ITSC)*, pages 2636–2643, Oct 2014.

- [RB15] Marvin Raaijmakers and Mohamed Essayed Bouzouraa. In-vehicle roundabout perception supported by a priori map data. In *2015 IEEE 18th International Conference on Intelligent Transportation Systems (ITSC)*, pages 437–443. IEEE, Sept 2015.
- [RF09] M. Ravanbakhsh and C.S. Fraser. Road roundabout extraction from very high resolution aerial imagery. In *Proc. CMRT09: Object Extraction for 3D City Models, Road Databases and Traffic Monitoring-Concepts, Algorithms and Evaluation*, pages 19–26, 2009.
- [RNS16] J. Rabe, M. Necker, and C. Stiller. Ego-lane estimation for lane-level navigation in urban scenarios. In *2016 IEEE Intelligent Vehicles Symposium (IV)*, pages 896–901, June 2016.
- [SAE14] Taxonomy and definitions for terms related to on-road motor vehicle automated driving systems. Technical Report SAE J3016, SAE International, 2014.
- [Sch15] Kai Schueler. *Model Based Multi Sensor 360 Degree Object Perception*. PhD thesis, TU München, 2015.
- [Sie03] Karl-Heinz Siedersberger. *Komponenten zur automatischen Fahrzeugführung in sehenden (semi-) autonomen Fahrzeugen*. PhD thesis, Universität der Bundeswehr München, 2003.
- [SKF13] M. Schreiber, C. Knoppel, and U. Franke. LaneLoc: Lane marking based localization using highly accurate maps. In *2013 IEEE Intelligent Vehicles Symposium (IV)*, pages 449–454, June 2013.
- [SWLM11] F. Saust, J. M. Wille, B. Lichte, and M. Maurer. Autonomous Vehicle Guidance on Braunschweig’s inner ring road within the Stadtpilot Project. In *2011 IEEE Intelligent Vehicles Symposium (IV)*, pages 169–174, June 2011.
- [TBF05] Sebastian Thrun, Wolfram Burgard, and Dieter Fox. *Probabilistic robotics*. MIT Press, 2005.
- [TPA07] M. Tsogas, A. Polychronopoulos, and A. Amditis. Using digital maps to enhance lane keeping support systems. In *2007 IEEE Intelligent Vehicles Symposium (IV)*, pages 148–153, June 2007.
- [UAB⁺08] Chris Urmson, Joshua Anhalt, Drew Bagnell, Christopher Baker, Robert Bittner, M. N. Clark, John Dolan, Dave Duggins, Tugrul Galatali, Chris Geyer, Michele Gittleman, Sam Harbaugh, Martial Hebert, Thomas M. Howard, Sascha Kolski, Alonzo Kelly, Maxim Likhachev, Matt McNaughton, Nick Miller, Kevin Peterson, Brian Pilnick, Raj Rajkumar, Paul Rybski, Bryan Salesky, Young-Woo Seo, Sanjiv Singh, Jarrod Snider, Anthony Stentz, William Red Whittaker, Ziv Wolkowicki, Jason Ziglar, Hong Bae, Thomas Brown, Daniel Demitrish, Bakhtiar Litkouhi, Jim Nickolaou, Varsha Sadekar, Wende Zhang, Joshua Struble, Michael Taylor, Michael Darms, and Dave Ferguson. Autonomous driving in urban environments: Boss and the Urban Challenge. *Journal of Field Robotics*, 25(8):425–466, 2008.
- [UM13] Simon Ulbrich and Markus Maurer. Probabilistic Online POMDP Decision Making for Lane Changes in Fully Automated Driving. In *2013 IEEE 16th International Conference on Intelligent Transportation Systems (ITSC)*, pages 2063–2067. IEEE, 2013.
- [WKD05] T. Weiss, N. Kaempchen, and K. Dietmayer. Precise ego-localization in urban areas using laserscanner and high accuracy feature maps. In *2005 IEEE Intelligent Vehicles Symposium (IV)*, pages 284–289, June 2005.

- [WS93] M. Worring and A.W.M. Smeulders. Digital curvature estimation. *CVGIP: Image Understanding*, 58(3):366 – 382, 1993.
- [WSM10] J. M. Wille, F. Saust, and M. Maurer. Stadtpilot: Driving autonomously on Braunschweig’s inner ring road. In *2010 IEEE Intelligent Vehicles Symposium (IV)*, pages 506–511, June 2010.
- [XPC⁺05] J. Xavier, M. Pacheco, D. Castro, A. Ruano, and U. Nunes. Fast line, arc/circle and leg detection from laser scan data in a player driver. In *2005 IEEE International Conference on Robotics and Automation (ICRA)*, pages 3930–3935, April 2005.
- [YL08] Ronald R. Yager and Liping Liu, editors. *Classic Works of the Dempster-Shafer Theory of Belief Functions*. Springer-Verlag Berlin Heidelberg, 2008.
- [ZBIG12a] C. Zinoune, P. Bonnifait, and J. Ibanez-Guzman. Detection of missing roundabouts in maps for driving assistance systems. In *2012 IEEE Intelligent Vehicles Symposium (IV)*, pages 123–128, June 2012.
- [ZBIG12b] C. Zinoune, P. Bonnifait, and J. Ibanez-Guzman. A sequential test for autonomous localisation of map errors for driving assistance systems. In *2012 IEEE 15th International Conference on Intelligent Transportation Systems (ITSC)*, pages 1377–1382, September 2012.
- [ZBS⁺14] J. Ziegler, P. Bender, M. Schreiber, H. Lategahn, T. Strauss, C. Stiller, Thao Dang, U. Franke, N. Appenrodt, C.G. Keller, E. Kaus, R.G. Herrtwich, C. Rabe, D. Pfeiffer, F. Lindner, F. Stein, F. Erbs, M.ENZweiler, C. Knöppel, J. Hipp, M. Haueis, M. Trepte, C. Brenk, A. Tamke, M. Ghanaat, M. Braun, A. Joos, H. Fritz, H. Mock, M. Hein, and E. Zeeb. Making Bertha Drive – An Autonomous Journey on a Historic Route. *IEEE Intelligent Transportation Systems Magazine*, 6(2):8–20, 2014.
- [Zin14] Clément Zinoune. *Autonomous Integrity Monitoring of Navigation Maps on board Intelligent Vehicles*. PhD thesis, UTC Compiègne, September 2014.
- [ZM15] J. Zeisler and H.-G. Maas. Analysis of the performance of a laser scanner for predictive automotive applications. *ISPRS Annals of Photogrammetry, Remote Sensing and Spatial Information Sciences*, II-3/W5:49–56, 2015.

Appendix A

Roundabouts Used for Evaluation

Below all roundabouts are listed that were used for the evaluations presented in this thesis. For each roundabout the left most entrance gate in the image, is the entrance gate with ID 0. The other entrance gates are number increasingly in counter-clockwise direction. All aerial images have scale 1 : 1200.



name: **Casino**
central island radius: 16.26 m
outer radius: 22.70 m
centre latitude: 48.7814022009°
centre longitude: 11.3638230157°
curb height: 32 cm
curb photo:



name: **MX303**
central island radius: 15.74 m
outer radius: 22.51 m
centre latitude: 48.7898582744°
centre longitude: 11.3768617386°
curb height: 12 cm
curb photo:





name: **Lana Grossa**
 central island radius: 13.31 m
 outer radius: 19.92 m
 centre latitude: 48.7915336044°
 centre longitude: 11.3774006896°
 curb height: 10 cm
 curb photo:



name: **Gymnasium**
 central island radius: 18.16 m
 outer radius: 24.90 m
 centre latitude: 48.7973385899°
 centre longitude: 11.3649696595°
 curb height: 16 cm
 curb photo:



name: **Gaimersheim Large**
 central island radius: 12.62 m
 outer radius: 19.99 m
 centre latitude: 48.8154349785°
 centre longitude: 11.3693231295°
 curb height: 19 cm
 curb photo:





name: **Gaimersheim Small**
central island radius: 6.32 m
outer radius: 14.21 m
centre latitude: 48.8107554450°
centre longitude: 11.3690717401°
curb height: 8 cm
curb photo:

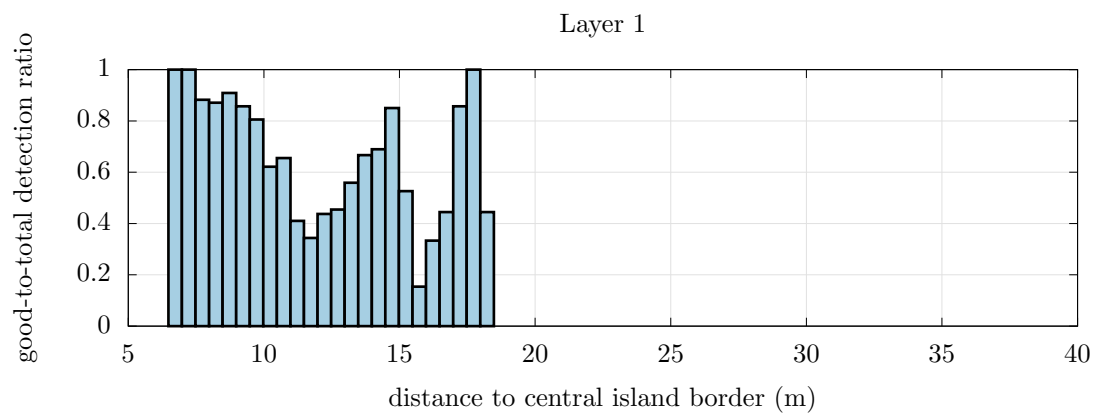
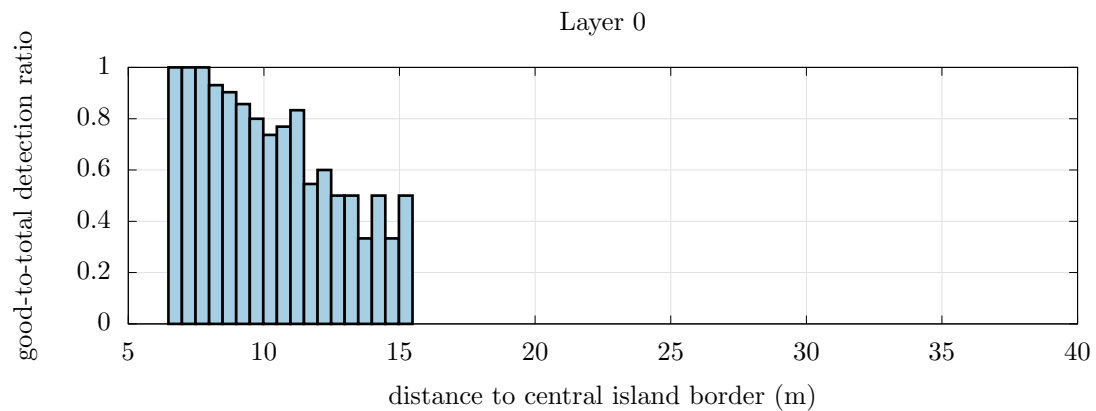


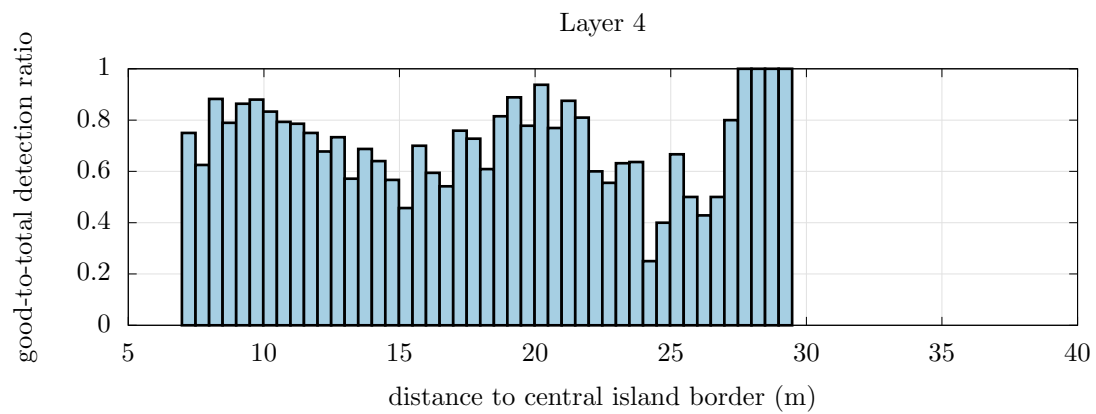
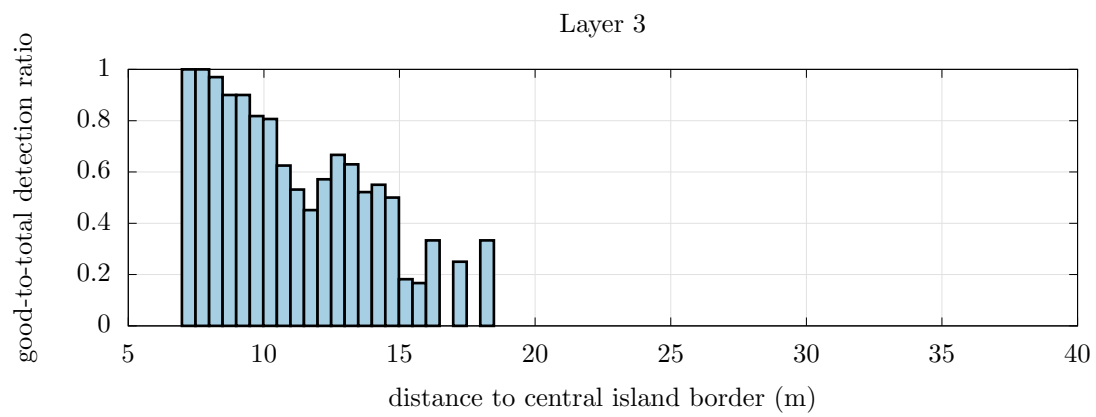
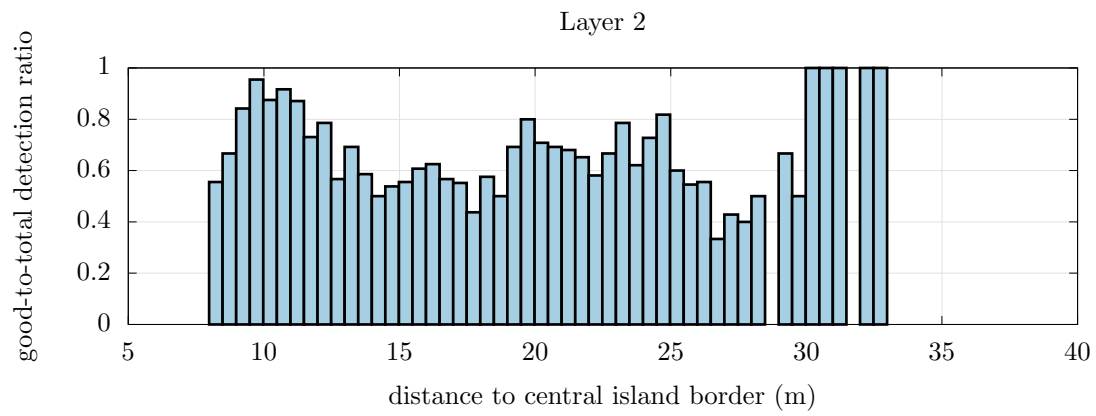
Source of the aerial photos:
Geobasisdaten: Bayerische Vermessungsverwaltung
(Bavarian Agency for Surveying and Geoinformation)

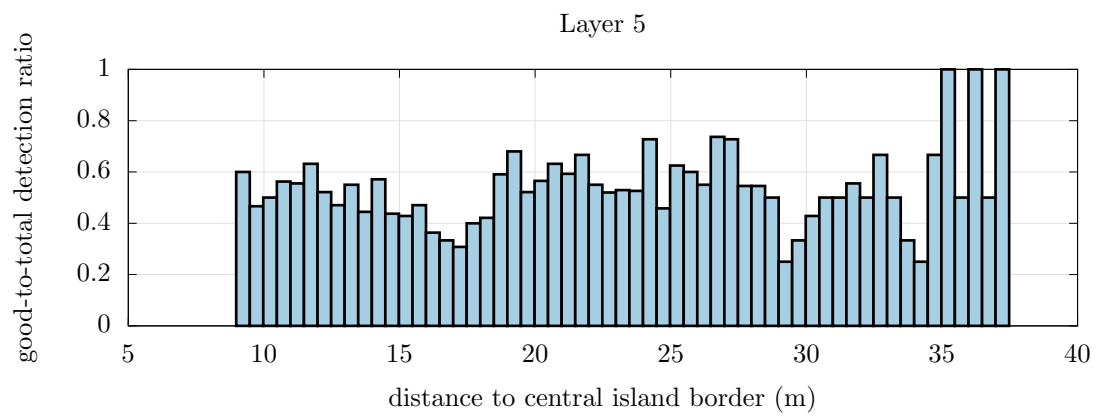
Appendix B

Good-to-Total Central Island Detection Ratio

Below you can find, for each layer of the laser scanner, the ratio between the number of good circle detections and the total number circle detections, by the central island detector presented in chapter 4, as a function of the vehicle's distance to the border of the central island. These graphs are based on the data from the evaluation discussed in section 4.6.







Appendix C

$\beta_{i,j}$ Sets

This appendix lists the $\beta_{i,j}$ sets that are used in section 6.6.3, where we consider the functions:

$$\begin{aligned} f_1(d, \theta) &= r_{k,\text{up}}(\theta), & f_2(d, \theta) &= d + R_{\text{max}}, & f_3(d, \theta) &= d + R_{\text{min}}, \\ f_4(d, \theta) &= r_{k,\text{max}}(d, \theta), & f_5(d, \theta) &= r_{k,\text{min}}(d, \theta), \\ h_1 &= \text{min}, & h_2 &= \text{max}, & h_3 &= \text{min}, & h_4 &= \text{max} \end{aligned}$$

For each i , $1 \leq i < 5$, and each j , $1 \leq j \leq i$, set $\beta_{i,j}$ is a $d\theta$ -interval set (see definition 10) such that $h_i(f_j(d, \theta), f_{i+1}(d, \theta)) = f_j(d, \theta)$ for each $(d, \theta) \in O(\beta_{i,j})$.

Note that these $d\theta$ -interval sets are evaluated for a certain range of θ values such that there exist a b for which either $r_{k,\text{up}}(\theta) = b/\sin(\beta + \theta)$ for each θ in the range, $r_{k,\text{up}}(\theta) = b/\cos(\beta + \theta)$ for each θ in the range. Observe that in both cases $r_{k,\text{up}}(\theta)$ can be written as $r_{k,\text{up}}(\theta) = a/\sin(\theta + \delta)$ for some positive value a and some value δ with $-\pi \leq \delta \leq \pi$. For the description of the $d\theta$ -interval sets below, we therefore assume that function $r_{k,\text{up}}$ can be written in this form.

Note that in the listed $d\theta$ -interval sets, some intervals are omitted for which $r_{k,\text{up}}(\theta)$, $r_{k,\text{min}}(d, \theta)$ or $r_{k,\text{min}}(d, \theta)$ is negative or for which all d values are less than $r_a - r_{\text{max}}$ or greater than $r_a - r_{\text{min}}$. This is done since these intervals will be masked out anyway by the interval intersections in equation 6.37.

- $\beta_{1,1}$ is the $d\theta$ -interval set for which $r_{k,\text{up}}(\theta) < d + R_{\text{max}}$. Here $r_{k,\text{up}}(\theta) = a/\sin(\theta + \delta)$ for some positive a and δ with $-\pi \leq \delta \leq \pi$.

– If $a > 0$ and $-\frac{\pi}{2} \leq \delta < 0$:

$$\begin{aligned} \beta_{1,1} = & \left\{ \left((-\infty, -a - R_{\text{max}}), \right. \right. & & \left. \left. \left\{ (d \mapsto -\pi - \delta, d \mapsto -\pi - \arcsin(\frac{a}{d+R_{\text{max}}}) - \delta), \right. \right. \right. \\ & & & \left. \left. \left. (d \mapsto \arcsin(\frac{a}{d+R_{\text{max}}}) - \delta, d \mapsto -\delta) \right\} \right), \right. \\ & \left((-a - R_{\text{max}}, a - R_{\text{max}}), \right. & & \left. \left\{ (d \mapsto -\pi - \delta, d \mapsto -\delta) \right\} \right), \\ & \left((a - R_{\text{max}}, \frac{-a}{\sin(\delta)} - R_{\text{max}}), \right. & & \left\{ (d \mapsto -\pi - \delta, d \mapsto -\delta), \right. \\ & & & \left. \left. (d \mapsto \arcsin(\frac{a}{d+R_{\text{max}}}) - \delta, \right. \right. \\ & & & \left. \left. d \mapsto \pi - \arcsin(\frac{a}{d+R_{\text{max}}}) - \delta) \right\} \right), \\ & \left((\frac{-a}{\sin(\delta)} - R_{\text{max}}, \infty), \right. & & \left. \left\{ (d \mapsto -\pi, d \mapsto -\pi - \arcsin(\frac{a}{d+R_{\text{max}}}) - \delta), \right. \right. \\ & & & \left. \left. (d \mapsto -\pi - \delta, d \mapsto -\delta), \right. \right. \\ & & & \left. \left. (d \mapsto \arcsin(\frac{a}{d+R_{\text{max}}}) - \delta, d \mapsto \pi) \right\} \right) \Big\} \end{aligned}$$

– If $a > 0$ and $-\pi \leq \delta < -\frac{\pi}{2}$:

$$\beta_{1,1} = \left\{ \left((-\infty, -a - R_{\max}), \quad \left\{ (d \mapsto -\pi - \delta, d \mapsto -\pi - \arcsin(\frac{a}{d+R_{\max}}) - \delta), \right. \right. \right. \\ \left. \left. \left. (d \mapsto \arcsin(\frac{a}{d+R_{\max}}) - \delta, d \mapsto -\delta) \right\} \right), \right. \\ \left(-a - R_{\max}, a - R_{\max} \right), \quad \left\{ (d \mapsto -\pi - \delta, d \mapsto -\delta) \right\}, \\ \left(a - R_{\max}, \frac{-a}{\sin(\delta)} - R_{\max} \right), \quad \left\{ (d \mapsto -2\pi + \arcsin(\frac{a}{d+R_{\max}}) - \delta, \right. \\ \left. d \mapsto -\pi - \arcsin(\frac{a}{d+R_{\max}}) - \delta), \right. \\ \left. (d \mapsto -\pi - \delta, d \mapsto -\delta) \right\}, \\ \left(\frac{-a}{\sin(\delta)} - R_{\max}, \infty \right), \quad \left\{ (d \mapsto -\pi, d \mapsto -\pi - \arcsin(\frac{a}{d+R_{\max}}) - \delta), \right. \\ \left. (d \mapsto -\pi - \delta, d \mapsto -\delta), \right. \\ \left. (d \mapsto \arcsin(\frac{a}{d+R_{\max}}) - \delta, d \mapsto \pi) \right\} \left. \right\}$$

– If $a > 0$ and $0 \leq \delta < \frac{\pi}{2}$:

$$\beta_{1,1} = \left\{ \left((-\infty, \frac{-a}{\sin(\delta)} - R_{\max}), \quad \left\{ (d \mapsto \arcsin(\frac{a}{d+R_{\max}}) - \delta, d \mapsto -\delta), \right. \right. \right. \\ \left. \left. \left. (d \mapsto \pi - \delta, d \mapsto \pi - \arcsin(\frac{a}{d+R_{\max}}) - \delta) \right\} \right), \right. \\ \left(\frac{-a}{\sin(\delta)} - R_{\max}, -a - R_{\max} \right), \quad \left\{ (d \mapsto -\pi, d \mapsto -\pi - \arcsin(\frac{a}{d+R_{\max}}) - \delta), \right. \\ \left. (d \mapsto \arcsin(\frac{a}{d+R_{\max}}) - \delta, d \mapsto -\delta), \right. \\ \left. (d \mapsto \pi - \delta, d \mapsto \pi) \right\}, \\ \left(-a - R_{\max}, a - R_{\max} \right), \quad \left\{ (d \mapsto -\pi, d \mapsto -\delta) \right. \\ \left. (d \mapsto \pi - \delta, d \mapsto \pi) \right\}, \\ \left(a - R_{\max}, \infty \right), \quad \left\{ (d \mapsto -\pi, d \mapsto -\delta), \right. \\ \left. (d \mapsto \arcsin(\frac{a}{d+R_{\max}}) - \delta, \right. \\ \left. d \mapsto \pi - \arcsin(\frac{a}{d+R_{\max}}) - \delta), \right. \\ \left. (d \mapsto \pi - \delta, d \mapsto \pi) \right\} \left. \right\}$$

– If $a > 0$ and $\frac{\pi}{2} \leq \delta \leq \pi$:

$$\beta_{1,1} = \left\{ \left((-\infty, \frac{-a}{\sin(\delta)} - R_{\max}), \quad \left\{ (d \mapsto \arcsin(\frac{a}{d+R_{\max}}) - \delta, d \mapsto -\delta), \right. \right. \right. \\ \left. \left. \left. (d \mapsto \pi - \delta, d \mapsto \pi - \arcsin(\frac{a}{d+R_{\max}}) - \delta) \right\} \right), \right. \\ \left(\frac{-a}{\sin(\delta)} - R_{\max}, -a - R_{\max} \right), \quad \left\{ (d \mapsto -\pi, d \mapsto -\delta) \right. \\ \left. (d \mapsto -\pi - \delta, d \mapsto \pi - \arcsin(\frac{a}{d+R_{\max}}) - \delta), \right. \\ \left. (d \mapsto 2\pi + \arcsin(\frac{a}{d+R_{\max}}) - \delta, d \mapsto \pi) \right\}, \\ \left(-a - R_{\max}, a - R_{\max} \right), \quad \left\{ (d \mapsto -\pi, d \mapsto -\delta) \right. \\ \left. (d \mapsto \pi - \delta, d \mapsto \pi) \right\}, \\ \left(a - R_{\max}, \infty \right), \quad \left\{ (d \mapsto -\pi, d \mapsto -\delta), \right. \\ \left. (d \mapsto \arcsin(\frac{a}{d+R_{\max}}) - \delta, \right. \\ \left. d \mapsto \pi - \arcsin(\frac{a}{d+R_{\max}}) - \delta), \right. \\ \left. (d \mapsto \pi - \delta, d \mapsto \pi) \right\} \left. \right\}$$

– If $a = 0$:

$$\beta_{1,1} = \left\{ \left((-R_{\max}, \infty), \quad \left\{ (d \mapsto -\pi, d \mapsto \pi) \right\} \right) \right\}$$

- $\beta_{2,1}$ is the $d\theta$ -interval set for which $r_{k,\text{up}}(\theta) > d + R_{\min}$. Here $r_{k,\text{up}}(\theta) = a/\sin(\theta + \delta)$ for some positive a and δ with $-\pi \leq \delta \leq \pi$.

– If $a > 0$ and $-\frac{\pi}{2} \leq \delta < 0$:

$$\beta_{2,1} = \left\{ \left((-\infty, -a - R_{\min}), \begin{array}{l} \{(d \mapsto -\pi, d \mapsto -\pi - \delta), \\ (d \mapsto -\pi - \arcsin(\frac{a}{d+R_{\min}}) - \delta, \\ d \mapsto \arcsin(\frac{a}{d+R_{\min}}) - \delta), \\ (d \mapsto -\delta, d \mapsto \pi)\} \end{array} \right), \right. \\ \left((-a - R_{\min}, a - R_{\min}), \begin{array}{l} \{(d \mapsto -\pi, d \mapsto -\pi - \delta), \\ (d \mapsto -\delta, d \mapsto \pi)\} \end{array} \right), \\ \left((a - R_{\min}, \frac{-a}{\sin(\delta)} - R_{\min}), \begin{array}{l} \{(d \mapsto -\pi, d \mapsto -\pi - \delta), \\ (d \mapsto -\delta, d \mapsto \arcsin(\frac{a}{d+R_{\min}}) - \delta), \\ (d \mapsto \pi - \arcsin(\frac{a}{d+R_{\min}}) - \delta, d \mapsto \pi)\} \end{array} \right), \\ \left. \left((\frac{-a}{\sin(\delta)} - R_{\min}, \infty), \begin{array}{l} \{(d \mapsto -\pi - \arcsin(\frac{a}{d+R_{\min}}) - \delta, d \mapsto -\pi - \delta), \\ (d \mapsto -\delta, d \mapsto \arcsin(\frac{a}{d+R_{\min}}) - \delta)\} \right) \right\}$$

– If $a > 0$ and $-\pi \leq \delta < -\frac{\pi}{2}$:

$$\beta_{2,1} = \left\{ \left((-\infty, -a - R_{\min}), \begin{array}{l} \{(d \mapsto -\pi, d \mapsto -\pi - \delta), \\ (d \mapsto -\pi - \arcsin(\frac{a}{d+R_{\min}}) - \delta, \\ d \mapsto \arcsin(\frac{a}{d+R_{\min}}) - \delta), \\ (d \mapsto -\delta, d \mapsto \pi)\} \end{array} \right), \right. \\ \left((-a - R_{\min}, a - R_{\min}), \begin{array}{l} \{(d \mapsto -\pi, d \mapsto -\pi - \delta), \\ (d \mapsto -\delta, d \mapsto \pi)\} \end{array} \right), \\ \left((a - R_{\min}, \frac{-a}{\sin(\delta)} - R_{\min}), \begin{array}{l} \{(d \mapsto -\pi, d \mapsto -2\pi + \arcsin(\frac{a}{d+R_{\min}}) - \delta), \\ (d \mapsto -\pi - \arcsin(\frac{a}{d+R_{\min}}) - \delta, d \mapsto -\pi - \delta), \\ (d \mapsto -\delta, d \mapsto \pi)\} \end{array} \right), \\ \left. \left((\frac{-a}{\sin(\delta)} - R_{\min}, \infty), \begin{array}{l} \{(d \mapsto -\pi - \arcsin(\frac{a}{d+R_{\min}}) - \delta, d \mapsto -\pi - \delta), \\ (d \mapsto -\delta, d \mapsto \arcsin(\frac{a}{d+R_{\min}}) - \delta)\} \right) \right\}$$

– If $a > 0$ and $0 \leq \delta < \frac{\pi}{2}$:

$$\beta_{2,1} = \left\{ \left((-\infty, \frac{-a}{\sin(\delta)} - R_{\min}), \begin{array}{l} \{(d \mapsto -\pi, d \mapsto \arcsin(\frac{a}{d+R_{\min}}) - \delta), \\ (d \mapsto -\delta, d \mapsto \pi - \delta), \\ (d \mapsto \pi - \arcsin(\frac{a}{d+R_{\min}}) - \delta, d \mapsto \pi)\} \end{array} \right), \right. \\ \left((\frac{-a}{\sin(\delta)} - R_{\min}, -a - R_{\min}), \begin{array}{l} \{(d \mapsto -\pi - \arcsin(\frac{a}{d+R_{\min}}) - \delta, \\ d \mapsto \arcsin(\frac{a}{d+R_{\min}}) - \delta), \\ (d \mapsto -\delta, d \mapsto \pi - \delta)\} \end{array} \right), \\ \left((-a - R_{\min}, a - R_{\min}), \{(d \mapsto -\delta, d \mapsto \pi - \delta)\} \right), \\ \left. \left((a - R_{\min}, \infty), \begin{array}{l} \{(d \mapsto -\delta, d \mapsto \arcsin(\frac{a}{d+R_{\min}}) - \delta), \\ (d \mapsto \pi - \arcsin(\frac{a}{d+R_{\min}}) - \delta, d \mapsto \pi - \delta)\} \right) \right\}$$

– If $a > 0$ and $\frac{\pi}{2} \leq \delta \leq \pi$:

$$\beta_{2,1} = \left\{ \left((-\infty, \frac{-a}{\sin(\delta)} - R_{\min}), \left\{ (d \mapsto -\pi, d \mapsto \arcsin(\frac{a}{d+R_{\min}}) - \delta), \right. \right. \right. \\ \left. \left. \left. (d \mapsto -\delta, d \mapsto \pi - \delta), \right. \right. \right. \\ \left. \left. \left. (d \mapsto \pi - \arcsin(\frac{a}{d+R_{\min}}) - \delta, d \mapsto \pi) \right\} \right), \\ \left((\frac{-a}{\sin(\delta)} - R_{\min}, -a - R_{\min}), \left\{ (d \mapsto -\delta, d \mapsto \pi - \delta), \right. \right. \\ \left. \left. \left. (d \mapsto \pi - \arcsin(\frac{a}{d+R_{\min}}) - \delta, \right. \right. \right. \\ \left. \left. \left. d \mapsto 2\pi + \arcsin(\frac{a}{d+R_{\min}}) - \delta \right\} \right), \\ \left((-a - R_{\min}, a - R_{\min}), \left\{ (d \mapsto -\delta, d \mapsto \pi - \delta) \right\} \right), \\ \left((a - R_{\min}, \infty), \left\{ (d \mapsto -\delta, d \mapsto \arcsin(\frac{a}{d+R_{\min}}) - \delta), \right. \right. \\ \left. \left. \left. (d \mapsto \pi - \arcsin(\frac{a}{d+R_{\min}}) - \delta, d \mapsto \pi - \delta) \right\} \right) \right\}$$

– If $a = 0$:

$$\beta_{2,1} = \left\{ \left((-\infty, -R_{\min}), \left\{ (d \mapsto -\pi, d \mapsto \pi) \right\} \right) \right\}$$

- $\beta_{2,2}$ is the $d\theta$ -interval set for which $d + R_{\max} > d + R_{\min}$. Since $R_{\max} > R_{\min}$, $d + R_{\max} > d + R_{\min}$ is true for each value of d and θ :

$$\beta_{2,2} = \left\{ \left((-\infty, \infty), \left\{ (d \mapsto -\pi, d \mapsto \pi) \right\} \right) \right\}$$

- $\beta_{3,1}$ is the $d\theta$ -interval set for which $r_{k,\text{up}}(\theta) < r_{k,\text{max}}(d, \theta)$. Here $r_{k,\text{up}}(\theta) = a/\sin(\theta + \delta)$ for some positive a and δ with $-\pi \leq \delta \leq \pi$. In this $d\theta$ -interval set we leave out the d, θ values for which $r_{k,\text{up}}(\theta)$ or $r_{k,\text{max}}(d, \theta)$ is negative.

Let, for a given value of d , $\theta_-(d)$ and $\theta_+(d)$ be the two angles in the range $\langle -\pi, \pi \rangle$ for which $r_{k,\text{up}}(\theta) = r_{k,\text{max}}(d, \theta)$:

$$\theta_{\pm}(d) = \text{atan2} \left(\frac{A(d) \pm \sqrt{B(d)}}{C(d)}, \frac{2a \cdot (d + r_{\max})E(d)^2 \cos^2(\delta) \mp \sqrt{B(d)}(2ar_a + E(d) \sin(\delta))}{E(d)C(d) \cos(\delta)} \right)$$

where

$$\begin{aligned} A(d) &= 4a^2 r_a \cdot (d + r_{\max}) + 2a \cdot (d + r_{\max})E(d) \sin(\delta) \\ B(d) &= E(d)^3 \cos^2(\delta)(E(d) + 4ar_a \sin(\delta) - 4a^2) \\ C(d) &= E(d)^2 + 4a^2 r_a^2 + 4ar_a E(d) \sin(\delta) \\ E(d) &= (d + r_{\max})^2 - r_a^2 \end{aligned}$$

and $\text{atan2}(x, y)$ is the angle in the range $\langle -\pi, \pi \rangle$ that vector (x, y) makes with the positive x -axis. In the following equations for $d\theta$ -interval set $\beta_{3,1}$ we use the following two values to refer to these intersection points:

$$\begin{aligned} \theta_1(d) &= \min(\theta_-(d), \theta_+(d)) \\ \theta_2(d) &= \max(\theta_-(d), \theta_+(d)) \end{aligned}$$

Now $d\theta$ -interval set $\beta_{3,1}$ can be described as follows:

– If $a = 0$:

$$\beta_{3,1} = \left\{ \left((r_a - r_{\max}, \infty), \left\{ (d \mapsto -\pi, d \mapsto \pi) \right\} \right) \right\}$$

– If $\delta \geq 0$ and $r_a \sin(\delta) < a$ and $a > 0$:

$$\beta_{3,1} = \left\{ \left((\sqrt{4a^2 + r_a^2 - 4ar_a \sin(\delta)} - r_{\max}, \infty), \left\{ (d \mapsto \theta_1, d \mapsto \theta_2) \right\} \right) \right\}$$

– If $\delta \geq 0$ and $r_a \sin(\delta) \geq a$ and $a > 0$:

$$\beta_{3,1} = \left\{ \left((r_a - r_{\max}, \infty), \{(d \mapsto \theta_1, d \mapsto \theta_2)\} \right) \right\}$$

– If $\delta < 0$ and $a > 0$:

$$\beta_{3,1} = \left\{ \left(\left(\sqrt{4a^2 + r_a^2 - 4ar_a \sin(\delta)} - r_{\max}, r_a - \frac{2a}{\sin(\delta)} - r_{\max} \right), \{(d \mapsto \theta_1, d \mapsto \theta_2)\} \right), \right. \\ \left. \left(\left(r_a - \frac{2a}{\sin(\delta)} - r_{\max}, \infty \right), \{(d \mapsto -\pi, d \mapsto \theta_1), \right. \right. \\ \left. \left. (d \mapsto \theta_2, d \mapsto \pi)\} \right) \right\}$$

• $\beta_{3,2}$ is the $d\theta$ -interval set for which $d + R_{\max} < r_{k,\max}(d, \theta)$

– If $r_a - r_{\max} < -R_{\max}$:

$$\beta_{3,2} = \left\{ \left((r_a - r_{\max}, r_{\max} - r_a - 2R_{\max}), \{(d \mapsto -\pi, d \mapsto \pi)\} \right), \right. \\ \left. \left((r_{\max} - r_a - 2R_{\max}, r_{\max} + r_a - 2R_{\max}), \{(d \mapsto -\theta_m, d \mapsto \theta_m)\} \right) \right\}$$

$$\text{where } \theta_m = \arccos \left(\frac{-(d+r_{\max})^2 + 2(d+r_{\max})(d+R_{\max}) + r_a^2}{2(d+R_{\max})r_a} \right).$$

– If $r_a - r_{\max} \geq -R_{\max}$:

$$\beta_{3,2} = \left\{ \left((r_a - r_{\max}, r_{\max} + r_a - 2R_{\max}), \{(d \mapsto -\theta_m, d \mapsto \theta_m)\} \right) \right\}$$

$$\text{where } \theta_m = \arccos \left(\frac{-(d+r_{\max})^2 + 2(d+r_{\max})(d+R_{\max}) + r_a^2}{2(d+R_{\max})r_a} \right).$$

• $\beta_{3,3}$ is the $d\theta$ -interval set for which $d + R_{\min} < r_{k,\max}(d, \theta)$

– If $r_a - r_{\max} < -R_{\min}$:

$$\beta_{3,3} = \left\{ \left((r_a - r_{\max}, r_{\max} - r_a - 2R_{\min}), \{(d \mapsto -\pi, d \mapsto \pi)\} \right), \right. \\ \left. \left((r_{\max} - r_a - 2R_{\min}, r_{\max} + r_a - 2R_{\min}), \{(d \mapsto -\theta_m, d \mapsto \theta_m)\} \right) \right\}$$

$$\text{where } \theta_m = \arccos \left(\frac{-(d+r_{\max})^2 + 2(d+r_{\max})(d+R_{\min}) + r_a^2}{2(d+R_{\min})r_a} \right).$$

– If $r_a - r_{\max} \geq -R_{\min}$:

$$\beta_{3,3} = \left\{ \left((r_a - r_{\max}, r_{\max} + r_a - 2R_{\min}), \{(d \mapsto -\theta_m, d \mapsto \theta_m)\} \right) \right\}$$

$$\text{where } \theta_m = \arccos \left(\frac{-(d+r_{\max})^2 + 2(d+r_{\max})(d+R_{\min}) + r_a^2}{2(d+R_{\min})r_a} \right).$$

• $\beta_{4,1}$ is the $d\theta$ -interval set for which $r_{k,\text{up}}(\theta) > r_{k,\text{min}}(d, \theta)$. Here $r_{k,\text{up}}(\theta) = a/\sin(\theta + \delta)$ for some positive a and δ with $-\pi \leq \delta \leq \pi$. In this $d\theta$ -interval set we leave out the d, θ values for which $r_{k,\text{up}}(\theta)$ or $r_{k,\text{min}}(d, \theta)$ is negative.

Let, for a given value of d , $\theta_-(d)$ and $\theta_+(d)$ be the two angles in the range $\langle -\pi, \pi \rangle$ for which $r_{k,\text{up}}(\theta) = r_{k,\text{min}}(d, \theta)$:

$$\theta_{\pm}(d) = \text{atan2} \left(\frac{A(d) \pm \sqrt{B(d)}}{C(d)}, \frac{2a \cdot (d + r_{\min})E(d)^2 \cos^2(\delta) \mp \sqrt{B(d)}(2ar_a + E(d) \sin(\delta))}{E(d)C(d) \cos(\delta)} \right)$$

where

$$\begin{aligned} A(d) &= 4a^2 r_a \cdot (d + r_{\min}) + 2a \cdot (d + r_{\min}) E(d) \sin(\delta) \\ B(d) &= E(d)^3 \cos^2(\delta) (E(d) + 4ar_a \sin(\delta) - 4a^2) \\ C(d) &= E(d)^2 + 4a^2 r_a^2 + 4ar_a E(d) \sin(\delta) \\ E(d) &= (d + r_{\min})^2 - r_a^2 \end{aligned}$$

and $\text{atan2}(x, y)$ is the angle in the range $\langle -\pi, \pi] \rangle$ that vector (x, y) makes with the positive x -axis. In the following equations for $d\theta$ -interval set $\beta_{4,1}$ we use the following two values to refer to these intersection points:

$$\begin{aligned} \theta_1(d) &= \min(\theta_-(d), \theta_+(d)) \\ \theta_2(d) &= \max(\theta_-(d), \theta_+(d)) \end{aligned}$$

In case there is only one positive intersection between the functions $r_{k,\text{up}}(\theta)$ and $r_{k,\text{min}}(d, \theta)$, the value for θ for this positive intersection is equal to $\theta_{p1}(d)$ if $\delta < \arccos\left(\frac{d+r_{\min}}{r_a}\right)$ and otherwise it is equal to $\theta_{p2}(d)$:

$$\begin{aligned} \theta_{p1}(d) &= \begin{cases} \theta_2(d) & \text{if } \delta = 0 \\ \theta_1(d) & \text{if } d \leq r_a - \frac{a}{\sin(\delta)} - r_{\min} \\ \theta_2(d) & \text{otherwise} \end{cases} \\ \theta_{p2}(d) &= \begin{cases} \theta_1(d) & \text{if } \delta = \pi \\ \theta_2(d) & \text{if } d \leq r_a - \frac{a}{\sin(\delta)} - r_{\min} \\ \theta_1(d) & \text{otherwise} \end{cases} \end{aligned}$$

Now $d\theta$ -interval set $\beta_{4,1}$ can be described as follows:

– If $a = 0$:

$$\beta_{4,1} = \emptyset$$

– If $0 \leq \delta \leq \frac{\pi}{2}$ and $r_a \sin(\delta) \leq a$ and $a > 0$:

$$\begin{aligned} \beta_{4,1} = & \left\{ \left((-\infty, -r_a - r_{\min}), \{(d \mapsto -\delta, d \mapsto \pi - \delta)\} \right), \right. \\ & \left((-r_a - r_{\min}, -r_a \cos(\delta) - r_{\min}), \left\{ (d \mapsto -\arccos\left(\frac{d+r_{\min}}{r_a}\right), \right. \right. \\ & \left. \left. d \mapsto \arccos\left(\frac{d+r_{\min}}{r_a}\right)) \right\} \right), \\ & \left((-r_a \cos(\delta) - r_{\min}, r_a \cos(\delta) - r_{\min}), \{(d \mapsto -\delta, d \mapsto \theta_{p1}(d))\} \right), \\ & \left. \left((r_a \cos(\delta) - r_{\min}, r_a - r_{\min}), \{(d \mapsto \theta_1(d), d \mapsto \theta_2(d))\} \right) \right\} \end{aligned}$$

– If $0 \leq \delta \leq \frac{\pi}{2}$ and $a < r_a \sin(\delta) \leq 2a$ and $a > 0$:

$$\begin{aligned} \beta_{4,1} = & \left\{ \left((-\infty, -r_a - r_{\min}), \{(d \mapsto -\delta, d \mapsto \pi - \delta)\} \right), \right. \\ & \left((-r_a - r_{\min}, -r_a \cos(\delta) - r_{\min}), \left\{ (d \mapsto -\arccos\left(\frac{d+r_{\min}}{r_a}\right), \right. \right. \\ & \left. \left. d \mapsto \arccos\left(\frac{d+r_{\min}}{r_a}\right)) \right\} \right), \\ & \left((-r_a \cos(\delta) - r_{\min}, r_a \cos(\delta) - r_{\min}), \{(d \mapsto -\delta, d \mapsto \theta_{p1}(d))\} \right), \\ & \left. \left((r_a \cos(\delta) - r_{\min}, F - r_{\min}), \{(d \mapsto \theta_1(d), d \mapsto \theta_2(d))\} \right) \right\} \end{aligned}$$

where $F = \sqrt{4a^2 + r_a^2 - 4ar_a \sin(\delta)}$.

– If $0 \leq \delta \leq \frac{\pi}{2}$ and $r_a \sin(\delta) > 2a$ and $a > 0$:

$$\beta_{4,1} = \left\{ \left((-\infty, -r_a - r_{\min}), \quad \{(d \mapsto -\delta, d \mapsto \pi - \delta)\} \right), \right. \\ \left((-r_a - r_{\min}, -F - r_{\min}), \quad \left\{ (d \mapsto -\arccos\left(\frac{d+r_{\min}}{r_a}\right), \right. \right. \\ \left. \left. d \mapsto \arccos\left(\frac{d+r_{\min}}{r_a}\right) \right\} \right), \\ \left((-F - r_{\min}, -r_a \cos(\delta) - r_{\min}), \quad \left\{ (d \mapsto -\arccos\left(\frac{d+r_{\min}}{r_a}\right), d \mapsto \theta_1(d), \right. \right. \\ \left. \left. (d \mapsto \theta_2(d), d \mapsto \arccos\left(\frac{d+r_{\min}}{r_a}\right)) \right\} \right), \\ \left. \left((-r_a \cos(\delta) - r_{\min}, r_a \cos(\delta) - r_{\min}), \{(d \mapsto -\delta, d \mapsto \theta_{p1}(d))\} \right) \right\}$$

where $F = \sqrt{4a^2 + r_a^2 - 4ar_a \sin(\delta)}$.

– If $\frac{\pi}{2} < \delta \leq \pi$ and $r_a \sin(\delta) \leq a$ and $a > 0$:

$$\beta_{4,1} = \left\{ \left((-\infty, -r_a - r_{\min}), \quad \{(d \mapsto -\delta, d \mapsto \pi - \delta)\} \right), \right. \\ \left((-r_a - r_{\min}, r_a \cos(\delta) - r_{\min}), \quad \left\{ (d \mapsto -\arccos\left(\frac{d+r_{\min}}{r_a}\right), \right. \right. \\ \left. \left. d \mapsto \arccos\left(\frac{d+r_{\min}}{r_a}\right) \right\} \right), \\ \left((r_a \cos(\delta) - r_{\min}, -r_a \cos(\delta) - r_{\min}), \{(d \mapsto \theta_{p2}(d), d \mapsto \pi - \delta)\} \right), \\ \left. \left((-r_a \cos(\delta) - r_{\min}, r_a - r_{\min}), \{(d \mapsto \theta_1(d), d \mapsto \theta_2(d))\} \right) \right\}$$

– If $\frac{\pi}{2} < \delta \leq \pi$ and $a < r_a \sin(\delta) \leq 2a$ and $a > 0$:

$$\beta_{4,1} = \left\{ \left((-\infty, -r_a - r_{\min}), \quad \{(d \mapsto -\delta, d \mapsto \pi - \delta)\} \right), \right. \\ \left((-r_a - r_{\min}, r_a \cos(\delta) - r_{\min}), \quad \left\{ (d \mapsto -\arccos\left(\frac{d+r_{\min}}{r_a}\right), \right. \right. \\ \left. \left. d \mapsto \arccos\left(\frac{d+r_{\min}}{r_a}\right) \right\} \right), \\ \left((r_a \cos(\delta) - r_{\min}, -r_a \cos(\delta) - r_{\min}), \{(d \mapsto \theta_{p2}(d), d \mapsto \pi - \delta)\} \right), \\ \left. \left((-r_a \cos(\delta) - r_{\min}, F - r_{\min}), \{(d \mapsto \theta_1(d), d \mapsto \theta_2(d))\} \right) \right\}$$

where $F = \sqrt{4a^2 + r_a^2 - 4ar_a \sin(\delta)}$.

– If $\frac{\pi}{2} < \delta \leq \pi$ and $r_a \sin(\delta) > 2a$ and $a > 0$:

$$\beta_{4,1} = \left\{ \left((-\infty, -r_a - r_{\min}), \quad \{(d \mapsto -\delta, d \mapsto \pi - \delta)\} \right), \right. \\ \left((-r_a - r_{\min}, -F - r_{\min}), \quad \left\{ (d \mapsto -\arccos\left(\frac{d+r_{\min}}{r_a}\right), \right. \right. \\ \left. \left. d \mapsto \arccos\left(\frac{d+r_{\min}}{r_a}\right) \right\} \right), \\ \left((-F - r_{\min}, r_a \cos(\delta) - r_{\min}), \quad \left\{ (d \mapsto -\arccos\left(\frac{d+r_{\min}}{r_a}\right), d \mapsto \theta_1(d), \right. \right. \\ \left. \left. (d \mapsto \theta_2(d), d \mapsto \arccos\left(\frac{d+r_{\min}}{r_a}\right)) \right\} \right), \\ \left. \left((r_a \cos(\delta) - r_{\min}, -r_a \cos(\delta) - r_{\min}), \{(d \mapsto \theta_{p2}(d), d \mapsto \pi - \delta)\} \right) \right\}$$

where $F = \sqrt{4a^2 + r_a^2 - 4ar_a \sin(\delta)}$.

– If $-\frac{\pi}{2} \leq \delta < 0$ and $a > 0$:

$$\beta_{4,1} = \left\{ \left((-\infty, -r_a - r_{\min}), \quad \{(d \mapsto -\pi, d \mapsto -\pi - \delta) \right. \right. \\ \left. \left. (d \mapsto -\delta, d \mapsto \pi)\} \right), \\ \left((-r_a - r_{\min}, -r_a \cos(\delta) - r_{\min}), \quad \left\{ (d \mapsto \theta_1(d), d \mapsto -\pi - \delta), \right. \right. \\ \left. \left. (d \mapsto -\delta, d \mapsto \theta_2(d)) \right\} \right), \\ \left. \left((-r_a \cos(\delta) - r_{\min}, r_a \cos(\delta) - r_{\min}), \{(d \mapsto -\delta, d \mapsto \theta_2(d))\} \right) \right\}$$

– If $-\pi \leq \delta < \frac{\pi}{2}$ and $a > 0$:

$$\beta_{4,1} = \left\{ \left((-\infty, -r_a - r_{\min}), \quad \begin{aligned} &\{(d \mapsto -\pi, d \mapsto -\pi - \delta) \\ &(d \mapsto -\delta, d \mapsto \pi)\} \end{aligned} \right), \right. \\ \left((-r_a - r_{\min}, r_a \cos(\delta) - r_{\min}), \quad \begin{aligned} &\{(d \mapsto \theta_1(d), d \mapsto -\pi - \delta), \\ &(d \mapsto -\delta, d \mapsto \theta_2(d))\} \end{aligned} \right), \\ \left. \left((r_a \cos(\delta) - r_{\min}, -r_a \cos(\delta) - r_{\min}), \{(d \mapsto \theta_1(d), d \mapsto -\pi - \delta)\} \right) \right\}$$

• $\beta_{4,2}$ is the $d\theta$ -interval set for which $d + R_{\max} > r_{k,\min}(d, \theta)$

– If $r_a > 0$ and $R_{\max} \geq r_a + r_{\min}$:

$$\beta_{4,2} = \left\{ \left((-R_{\max}, -r_a - r_{\min}), \quad \{(d \mapsto -\pi, d \mapsto \pi)\} \right), \right. \\ \left((-r_a - r_{\min}, r_a - r_{\min}), \quad \begin{aligned} &\{(d \mapsto -\pi, d \mapsto -\arccos(\frac{d+r_{\min}}{r_a}))\}, \\ &(d \mapsto -\theta_m, d \mapsto \theta_m), \\ &(d \mapsto \arccos(\frac{d+r_{\min}}{r_a}), d \mapsto \pi)\} \end{aligned} \right) \left. \right\}$$

$$\text{where } \theta_m = \arccos \left(\frac{-(d+r_{\min})^2 + 2(d+r_{\min})(d+R_{\max}) + r_a^2}{2(d+R_{\max})r_a} \right).$$

– If $r_a > 0$ and $R_{\max} < r_a + r_{\min}$:

$$\beta_{4,2} = \left\{ \left((-r_a - r_{\min}, r_{\min} + r_a - 2R_{\max}), \quad \begin{aligned} &\{(d \mapsto -\pi, d \mapsto -\arccos(\frac{d+r_{\min}}{r_a}))\}, \\ &(d \mapsto \arccos(\frac{d+r_{\min}}{r_a}), d \mapsto \pi)\} \end{aligned} \right), \right. \\ \left((r_{\min} + r_a - 2R_{\max}, r_a - r_{\min}), \quad \begin{aligned} &\{(d \mapsto -\pi, d \mapsto -\arccos(\frac{d+r_{\min}}{r_a}))\}, \\ &(d \mapsto -\theta_m, d \mapsto \theta_m), \\ &(d \mapsto \arccos(\frac{d+r_{\min}}{r_a}), d \mapsto \pi)\} \end{aligned} \right) \left. \right\}$$

$$\text{where } \theta_m = \arccos \left(\frac{-(d+r_{\min})^2 + 2(d+r_{\min})(d+R_{\max}) + r_a^2}{2(d+R_{\max})r_a} \right).$$

– If $r_a = 0$ and $R_{\max} \geq r_a + r_{\min}$:

$$\beta_{4,2} = \left\{ \left((-R_{\max}, \infty), \{(d \mapsto -\pi, d \mapsto \pi)\} \right) \right\}$$

– If $r_a = 0$ and $R_{\max} < r_a + r_{\min}$ and $r_{\min} - 2R_{\max} \leq -r_a - r_{\min}$:

$$\beta_{4,2} = \left\{ \left((-r_a - r_{\min}, \infty), \{(d \mapsto -\pi, d \mapsto \pi)\} \right) \right\}$$

– If $r_a = 0$ and $R_{\max} < r_a + r_{\min}$ and $r_{\min} - 2R_{\max} > -r_a - r_{\min}$:

$$\beta_{4,2} = \left\{ \left((r_{\min} - 2R_{\max}, \infty), \{(d \mapsto -\pi, d \mapsto \pi)\} \right) \right\}$$

• $\beta_{4,3}$ is the $d\theta$ -interval set for which $d + R_{\min} > r_{k,\min}(d, \theta)$

– If $r_a > 0$ and $R_{\min} \geq r_a + r_{\min}$:

$$\beta_{4,3} = \left\{ \left((-R_{\min}, -r_a - r_{\min}), \quad \{(d \mapsto -\pi, d \mapsto \pi)\} \right), \right. \\ \left((-r_a - r_{\min}, r_a - r_{\min}), \quad \begin{aligned} &\{(d \mapsto -\pi, d \mapsto -\arccos(\frac{d+r_{\min}}{r_a}))\}, \\ &(d \mapsto -\theta_m, d \mapsto \theta_m), \\ &(d \mapsto \arccos(\frac{d+r_{\min}}{r_a}), d \mapsto \pi)\} \end{aligned} \right) \left. \right\}$$

$$\text{where } \theta_m = \arccos \left(\frac{-(d+r_{\min})^2 + 2(d+r_{\min})(d+R_{\min}) + r_a^2}{2(d+R_{\min})r_a} \right).$$

– If $r_a > 0$ and $R_{\min} < r_a + r_{\min}$:

$$\beta_{4,3} = \left\{ \left((-r_a - r_{\min}, r_{\min} + r_a - 2R_{\min}), \left\{ (d \mapsto -\pi, d \mapsto -\arccos\left(\frac{d+r_{\min}}{r_a}\right)), \right. \right. \right. \\ \left. \left. \left. (d \mapsto \arccos\left(\frac{d+r_{\min}}{r_a}\right), d \mapsto \pi) \right\} \right), \right. \\ \left. \left((r_{\min} + r_a - 2R_{\min}, r_a - r_{\min}), \left\{ (d \mapsto -\pi, d \mapsto -\arccos\left(\frac{d+r_{\min}}{r_a}\right)), \right. \right. \right. \\ \left. \left. \left. (d \mapsto -\theta_m, d \mapsto \theta_m), \right. \right. \right. \\ \left. \left. \left. (d \mapsto \arccos\left(\frac{d+r_{\min}}{r_a}\right), d \mapsto \pi) \right\} \right) \right\}$$

$$\text{where } \theta_m = \arccos\left(\frac{-(d+r_{\min})^2 + 2(d+r_{\min})(d+R_{\min}) + r_a^2}{2(d+R_{\min})r_a}\right).$$

– If $r_a = 0$ and $R_{\min} \geq r_a + r_{\min}$:

$$\beta_{4,3} = \left\{ \left((-R_{\min}, \infty), \left\{ (d \mapsto -\pi, d \mapsto \pi) \right\} \right) \right\}$$

– If $r_a = 0$ and $R_{\min} < r_a + r_{\min}$ and $r_{\min} - 2R_{\min} \leq -r_a - r_{\min}$:

$$\beta_{4,3} = \left\{ \left((-r_a - r_{\min}, \infty), \left\{ (d \mapsto -\pi, d \mapsto \pi) \right\} \right) \right\}$$

– If $r_a = 0$ and $R_{\min} < r_a + r_{\min}$ and $r_{\min} - 2R_{\min} > -r_a - r_{\min}$:

$$\beta_{4,3} = \left\{ \left((r_{\min} - 2R_{\min}, \infty), \left\{ (d \mapsto -\pi, d \mapsto \pi) \right\} \right) \right\}$$

- $\beta_{4,4}$ is the $d\theta$ -interval set for which $r_{k,\max}(d, \theta) > r_{k,\min}(d, \theta)$. Since this inequality holds per definition for each value of d and θ , we have:

$$\beta_{4,4} = \left\{ \left((-\infty, \infty), \left\{ (d \mapsto -\pi, d \mapsto \pi) \right\} \right) \right\}$$

Curriculum Vitae

Marvin Raaijmakers was born in the Netherlands on the 4th of September 1988. In 2009, he received his Bachelor of Science degree in Computer Science and Engineering (with a minor in Applied Physics) from the Eindhoven University of Technology. He completed his Master of Science degree in Computer Science and Engineering in 2012. Both the Bachelor and Master degree were completed cum laude. His Master's graduation project "Detecting Moving Vehicles via a Moving Monocular Omnidirectional Side Camera" was done at Audi AG, Germany. After obtaining his Master degree he started working at Audi AG on his PhD research of which the results are presented in this dissertation.

Index

- ε -circle-approximation, 29
- ε -semi-convex, 34
- active lane keeping assistance, 2
- active safety systems, 2
- adaptive cruise control, 2
- behaviour planning, 20
- behaviour planning system, 1
- benchmark sequence, 42
- boundary, 20
- boundary segment, 20
- boundary subsegment hypothesis, 24
- central island, 14, 27–58
- circle approximation, 29
- circulatory roadway, 14
- closed-interval interpretation, 95
- collision mitigation systems, 2
- convex, 33
- curvature partitioning, 38
- curvature-based partitioning, 31
- $d\theta$ -interval set, 94
- DARPA Urban Challenge, 1, 15
- DGPS, 7
- driver assistance systems, 2
- ego lane, 6, 74
- environment coordinate system, 13
- environment perception system, 1
- gate, 14
- GNNs, 6
- good fit, 30
- good-to-total detection ratio, 50, 63, 127
- highly automated driving, 3
- horizontal field of view, 15
- horizontal resolution, 15
- Hough transform, 7
- Jordan curve, 29
- lane, 20
- lane departure warning, 2
- laser scan, 15
- layer, 15
- least-square circle, 30
- lower convex hull, 33
- maximal ε -semi-convex subsequence, 36
- minimal convex partitioning, 34
- osculating circle, 38
- particle filter, 62
- partitioning, 31
 - curvature, 38
 - curvature-based, 31
 - semi-convex, 31, 33
- principle of indifference, 79, 107
- roadway, 19
- roadway hypothesis, 21
- roadway segment, 20
- roadway subsegment, 22
- roadway subsegment hypothesis, 22
- scan layer, 15
- segment, 31
- semi-convex partitioning, 31, 33
- semi-convexity error, 34
- splitter island, 14
- star-shaped, 29
- strip, 20
- strip segment, 20
- strip subsegment hypothesis, 24
- subsequence, 30
- traffic jam assistance, 2
- truck apron, 14
- vehicle control system, 2
- vertical field of view, 15
- vertical resolution, 15

zama gwigwi, gwigwe gwagwe
... that's African