Christian Frieß, BSc

Methodology for Parametrization and Validation of a Virtual Radar Sensor Model

MASTER'S THESIS

to achieve the university degree of

Diplom-Ingenieur

Master's degree programme: Mechanical Engineering and Business Economics

submitted to

Graz University of Technology

Supervisors

Assoc.Prof. Dipl.-Ing. Dr.techn. Arno Eichberger

M.Sc Zoltan Magosi

Institute of Automotive Engineering, Graz University of Technology
Member of [FSI]

Dipl.-Ing. (FH) Dr.techn. Stefan Bernsteiner

MAGNA STEYR Fahrzeugtechnik AG & CoKG

Graz, May 2018

Restricted access until May 2020

Acknowledgement

At this point, I would like to thank all those who supported me directly or indirectly during this thesis and the entire time of the studies.

I would like to thank Dr. Arno Eichberger and M.Sc Zoltan Magosi from the Institute of Automotive Engineering at Technical University of Graz for giving me the opportunity to work on this thesis and their support. I would also like to thank Dr. Stefan Bernsteiner from MAGNA STEYR Fahrzeugtechnik AG & CoKG for the good cooperation and support during this thesis.

I want to express my deep gratitude to my family, especially to my parents Elisabeth and Herbert, for enabling me this study and giving me mental support during the whole studies. Finally, a special thank goes to my friends and fellow students, which motivated me during busy months and have contributed to nice years of study.

Thank you!

Statutory Declaration

I declare that I have authored this thesis independently, that I have not used other than the declared sources / resources, and that I have explicitly marked all material which has been quoted either literally or by content from the used sources.

....
(date) (signature)

Abstract

Advanced Driver Assistance Systems (ADAS) are integrated in modern vehicles to support the driver in his\her driving task by providing him\her with information, warning him\her and influencing the longitudinal or lateral guidance of the vehicle in critical situations. High complexity of ADAS in combination with the trend towards *Automated Driving Systems* (ADS) lead to enormous testing and validation effort during the whole development process, to ensure accurate functionality of the systems and expected improvements regarding safety and comfort. To reduce the amount of time-consuming, expensive, hardly reproducible and partly safety critical test drives with real vehicles, methods for the virtual validation of ADAS and ADS by simulation have to be developed. Essential for the functionality of ADAS and ADS functions are environment perception sensors, e.g. *Light Detection and Ranging* (LIDAR) sensors, *Radio Detection and Ranging* (RADAR) sensors and cameras. Therefore, the key factor for meaningful simulation is the usage of valid environment perception sensor models, which are able to represent characteristics of real sensor hardware. To prove this, sensor models as such have to be validated before being used for validation of ADAS and ADS.

The current thesis presents a method for the parametrization and validation of sensor models based on real test drives. It is applied on the physical *High Fidelity* (HiFi) RADAR sensor model, as implemented in the software tool IPG CarMaker®. Therefore the impact of adjustable parameters on the output of the sensor model is analysed. This is achieved by reproducing the driven manoeuvre of the real test drive within a virtual scenario. Then comparison and assessment of synthetically generated output data of the virtual sensor model during simulation to recorded data from the real RADAR sensor hardware during the real test drive, by calculation and visualisation of deviations, within the software tool MATLAB® is realised.

Furthermore, a model of the test track on the proving ground at MAGNA STEYR Fahrzeugtechnik AG & CoKG is build up for future virtual testing applications of camera based ADAS and ADS functions, within this thesis. The modelling process which includes high accurate GPS measurement of the test track, transformation of coordinates into a usable coordinate system, calculation of the lane widths to the final modelling of the test track within the software tool IPG CarMaker® is presented in this thesis.

Kurzfassung

Fahrerassistenzsysteme (FAS) sind in modernen Fahrzeugen integriert, um die Fahrer bei ihrer Fahraufgabe zu unterstützen, indem sie sie informieren, warnen und in kritischen Situationen die Längs- oder Querführung des Fahrzeugs beeinflussen. Die hohe Komplexität von FAS in Kombination mit dem Trend zu autonomen Fahrfunktionen, führt zu einem enormen Test- und Validierungsaufwand während des gesamten Entwicklungsprozesses, um eine angemessene Funktionalität der Systeme und erwartete Verbesserungen in Bezug auf Sicherheit und Komfort sicherzustellen. Um die zeitaufwändigen, teuren, schwer reproduzierbaren und teilweise sicherheitskritischen Testfahrten mit realen Fahrzeugen zu reduzieren, müssen Methoden zur virtuellen Validierung von FAS durch Simulation entwickelt werden. Wesentlich für die Funktionalität von FAS Funktionen sind Sensoren zur Umgebungserfassung wie *Light Detection and Ranging* (LIDAR) Sensoren, *Radio Detection and Ranging* (RADAR) Sensoren und Kameras. Daher ist der Schlüssel für eine aussagekräftige Simulation die Verwendung von gültigen Modellen, welche die spezifischen Eigenschaften der realen Sensoren darstellen können. Um dies zu beweisen, müssen Sensormodelle als solche validiert werden, bevor sie für die Validierung von FAS eingesetzt werden können.

Die vorliegende Arbeit stellt eine Methode zur Parametrierung und Validierung von Sensormodellen vor, die auf realen Testfahrten basiert. Sie wird auf ein physikalisches *High Fidelity* (HiFi) Radarsensormodell angewendet, wie es im Softwaretool IPG CarMaker® implementiert ist. Hierfür wird der Einfluss der einstellbaren Parameter auf die Ausgabe des Sensormodells analysiert. Das abgefahrene Manöver der realen Testfahrt wird innerhalb eines virtuellen Szenarios aufgebaut. Anschließend erfolgt ein Vergleich und Bewertung von synthetisch erzeugten Ausgangsdaten des virtuellen Sensormodelles während der Simulation mit aufgezeichneten Daten des realen Radarsensors während der realen Testfahrt durch Berechnung und Visualisierung mit dem Softwaretool MATLAB®.

Darüber hinaus wird im Rahmen dieser Arbeit ein Modell der Teststrecke auf dem Testgelände von MAGNA STEYR Fahrzeugtechnik AG & CoKG, für zukünftige virtuelle Testanwendungen von kamerabasierten FAS, aufgebaut. Der Modellierungsprozess beinhaltet eine hochgenaue GPS-Messung der Teststrecke, Koordinatentransformation in ein verwendbares Koordinatensystem, Berechnung der Fahrbahnbreiten bis zur Modellierung der Teststrecke innerhalb des Softwaretools IPG CarMaker® und wird im Rahmen dieser Arbeit vorgestellt.

Contents

Abbreviations

ACC	Adaptive Cruise Control
ADAS	Advanced Driver Assistance System
ADMA	Automotive Dynamic Motion Analyser
ADS	Automated Driving System
AEB	Automatic Emergency Brake
DAU	Data Acquisition Unit
DDT	Dynamic Driving Task
DiL	Driver in the Loop
ECU	Electronic Control Unit
ESC	Electronic Stability Control
FMCW	Frequency Modulated Continuous Wave
FMSK	Frequency Modulated Shift Keying
FNBW	First Null Beamwidth
FOV	Field of View
FSK	Frequency Shift Keying
GPS	Global Positioning System
HiL	Hardware in the Loop
HiFi	High Fidelity
HLF	High Level Fusion
HPBW	Half Power Beamwidth
IFF	Input From File
KML	Keyhole Markup Language
LIDAR	Light Detection and Ranging
LLF	Low Level Fusion
LKA	Lane Keeping Assistant
LRR	Long Range Radar
MRR	Mid Range Radar
NS	Navigation Systems
ODD	Operational Design Domain
OEDR	Object and Event Detection and Response
OL	Object List
PDF	Probability Density Function
RADAR	Radio Detection and Ranging
RCS	Radar Cross Section
RTK	Real Time Kinematic
SAE	Society of Automotive Engineers

SNR	Signal to Noise Ratio
SRR	Short Range Radar
ViL	Vehicle in the Loop
WGS84	World Geodetic System 1984
WLAN	Wireless Local Area Network

Symbols

Parameters and Variables

a	Semi-major axis of reference ellipsoid
A	Area, Amplitude
b	Semi-minor axis of reference ellipsoid
B	Bandwidth, Meridian arc length
c	Speed of light, Translation parameter
d	Distance
D	Power density, Directivity
F	Factor
f	Frequency
G	Antenna gain
h	Geographic height
k	Boltzmann's constant
L	Loss
N	Noise, Radius of curvature
P	Power
r	Radius, Rotation parameter
$\dot{r}$	Relative velocity
R	Distance, Range
S	Signal
t	Time
T	Temperature, Periodic time
U	Radiation intensity
X	Cartesian coordinate in x - direction
x	Gauß-Krüger coordinate (easting)
Y	Cartesian coordinate in y - direction
y	Gauß-Krüger coordinate (northing)
Z	Cartesian coordinate in z - direction

ϕ	Azimuth angle, Geographic latitude
Δ	Difference
ε	Error angle, Numerical eccentricity
δ	Phase angle
π	Pi
τ	Propagation time
σ	Radar Cross Section
μ	Scale factor
Σ	Sum
λ	Wavelength, Geographic longitude

Vectors

$\mathbf{a}$	Position vector of point A
$\mathbf{b}$	Position vector of point B
$\mathbf{c}$	Position vector of point C
$\mathbf{p}$	Position vector of point P
$\mathbf{u}$	Direction vector
$\mathbf{x}$	Line vector

Indices

a	Atmospheric
ave	Average
B	Bessel ellipsoid
CR	Corner cube reflector
D	Doppler
DR	Doppler RADAR
FSK	Frequency Shift Keying
i	Input
max	Maximum
min	Minimum
n	Noise
O	Object, Output
r	Receive
R	Receiver
sp	Signal processing
tot	Total
t	Transmit
T	Transmitter
WGS	World Geodetic System

1 Introduction

1.1 Motivation

Due to the increasing demand for safety and comfort of road users and performance of modern vehicles, the development and integration of *Advanced Driver Assistant Systems* (ADAS) play an important role within the vehicle development process. One main driver is the set goal from the European Commission, to reduce the number of 31500 road fatalities in 2010 by half until the year 2020 [9]. Figure 1.1 illustrates fatalities within the 28 European countries from 2001 until 2017 including a forecast for 2020. It indicates that the number of fatalities has shown a stagnating tendency over the last 4 years. According to [7], in Germany 88 % of road accidents with personal damage are caused by human error. Therefore, to reach the set goal, among others, increased deployment of ADAS is stipulated.

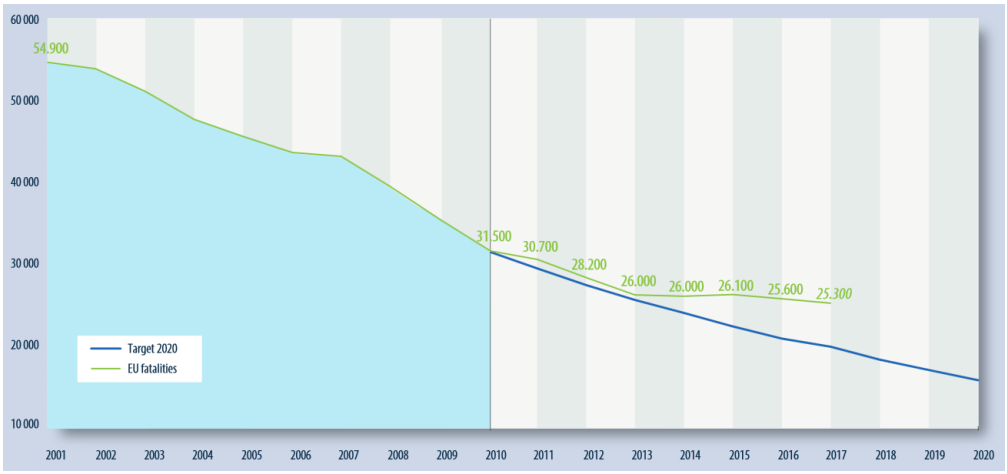


Figure 1.1: Fatalities from 2001 to 2017 and forecast for 2020 within the EU28 [10]

The European *New Car Assessment Programme* (NCAP) partly adopts the *Automatic Emergency Brake* (AEB) and *Lane Keeping Assist* (LKA) in their assessment program [28]. This will shift ADAS from optional to standard equipment of modern vehicles. High complexity of ADAS in combination with the next big step towards the development of *Automated Driving Systems* ADS makes exhaustive testing and validation of these systems necessary in every step of the development process, in order to ensure accurate

functionality of the systems and expected improvements regarding safety and comfort. This often requires high effort in performing real test drives, which are time-consuming and expensive. Particular attention within the test drives is paid to the testing of critical driving situations, which can no longer be controlled by the human driver, what in turn leads to dangerous situations for the test driver. Furthermore, the reproducibility of test scenarios is often not satisfactory, due to changing environment such as other road users or weather conditions. All points mentioned above lead to the demand for the development of methods for the virtual validation of ADAS and ADS by simulation.

Since perception sensors for environment recognition e.g. *Light Detection and Ranging* (LIDAR), *Radio Detection and Ranging* (RADAR) and camera are essential for the functionality of ADAS and ADS functions, the prerequisite for virtual development and validation of these systems is to have valid models of the perception sensors, which are able to represent the specific characteristics and behaviour of the real sensor hardware in a sufficient way. The output of these virtual sensor models is the basis for the algorithms of ADAS and ADS functions. Therefore, they have to be validated and compared to real sensor data before being implemented into a simulation framework [37].

1.2 Structure and Goal of this Work

Chapter 2 gives an overview about state of the art ADAS and different levels of automation, including perception sensors for environment recognition with a focus on RADAR sensors and their basic physical principles for distance and velocity measurement. An overview of existing approaches for RADAR sensor modelling within the virtual development process is given and suggested methods for their validation are presented.

Chapter 3 is divided into two parts. The first part deals with the modelling process of the test track on the proving ground of MAGNA STEYR Fahrzeugtechnik AG & CoKG. All steps of the process, starting with high accurate GPS measurement of the test track, over transformation of the coordinates into a usable coordinate system, calculation of the lane widths, to final modelling of the test track within the simulation tool IPG CarMaker® are described within this chapter.

Within the second part of this chapter a method for the parametrization and validation of a physical *High Fidelity* (HiFi) RADAR sensor model, as implemented in IPG CarMaker®, is presented. The goal of this part of the thesis is to analyse the impact of available and adjustable parameters within the RADAR sensor model on its output. This is realised by reproducing the driven manoeuvre of the real test drive, in particular trajectory of participating vehicles, in a virtual scenario and comparison of recorded data from the real RADAR sensor hardware during real test drives with synthetically generated data from the virtual sensor model during simulation.

Chapter 4 summarizes the outcome of the applied methodology and gives an outlook as well as recommendations for further steps and considerations to continue, develop and improve these methods and outcomes.

2 State of the Art

2.1 Advanced Driver Assistance Systems

Advanced Driver Assistance Systems (ADAS) are complex systems, which are expected to support the driver in his\her driving task in certain driving situations by

- providing him\her with information that goes beyond his\her own perception,
- warning him\her in critical situations,
- influencing lateral or longitudinal motion of the vehicle in critical situations, to prevent an accident or at least reduce the consequences,

considering that the warnings can be optical, acoustical or haptic and the interventions of ADAS can be initiated by the driver or automatically by the system.

The term ADAS covers a wide field, starting from systems with only informative purpose, to systems with limited possibility of intervention, up to potential autonomous driving functions within *Automated Driving Systems* (ADS) in the future [38]. A classification of ADAS can be made according to a variety of criteria. Donges in [14], classifies ADAS based on the three levels

- navigation,
- guidance and
- stabilization

of the driving task, on which the system is operating. Examples for ADAS that support the driver on the different levels are *Navigation Systems* (NS) for navigation, *Adaptive Cruise Control* (ACC) for guidance and *Electronic Stability Control* (ESC) for stabilization.

According to Bernsteiner [5], another way to classify ADAS is to differentiate between

- safety-oriented and
- comfort-oriented systems.

While the main task of safety-oriented ADAS such as *Automatic Emergency Brake* (AEB) is to prevent collisions between vehicles or other road users, comfort-oriented systems take over driving tasks to support the driver. ACC assists the driver in longitudinal direction, while the *Lane Keeping Assistant* (LKA) supports in lateral direction of the vehicle. The systems mentioned here are only a small selection of existing ADAS for all

three levels of the driving task which are well described in literature like Winner et al [47].

Due to the growing trend towards automated driving, standards for classification of levels of driving automation have been developed. Figure 2.1 gives an overview about levels of driving automation, according to the *Society of Automotive Engineers* (SAE). It ranges from level 0 (no driving automation) to level 5 (full driving automation). The classification is based on whether driver or system is responsible for the *Dynamic Driving Task* (DDT) and/or the fallback for the DDT. The DDT can be further divided into lateral and longitudinal control of the vehicle and *Object and Event Detection and Response* (OEDR), in other words monitoring of the driving environment. The *Operational Design Domain* (ODD) defines for which conditions, including driving modes, an ADAS or ADS is designed to operate accurately [31].

Level	Name	Narrative definition	DDT		DDT fallback	ODD
			Sustained lateral and longitudinal vehicle motion control	OEDR		
Driver performs part or all of the DDT						
0	**No Driving Automation**	The performance by the *driver* of the entire *DDT*, even when enhanced by *active safety systems*.	*Driver*	*Driver*	*Driver*	n/a
1	**Driver Assistance**	The *sustained* and *ODD*-specific execution by a *driving automation system* of either the *lateral* or the *longitudinal vehicle motion control* subtask of the DDT (but not both simultaneously) with the expectation that the *driver* performs the remainder of the *DDT*.	*Driver* and *System*	*Driver*	*Driver*	Limited
2	**Partial Driving Automation**	The *sustained* and *ODD*-specific execution by a *driving automation system* of both the *lateral* and *longitudinal vehicle motion control* subtasks of the *DDT* with the expectation that the *driver* completes the *OEDR* subtask and *supervises* the *driving automation system*.	***System***	*Driver*	*Driver*	Limited
***ADS* ("*System*") performs the entire *DDT* (while engaged)**			*System*	***System***	*Fallback-ready user (becomes the driver during fallback)*	Limited
3	**Conditional Driving Automation**	The *sustained* and *ODD*-specific performance by an *ADS* of the entire DDT with the expectation that the *DDT fallback-ready user* is *receptive* to *ADS*-issued *requests to intervene*, as well as to *DDT performance-relevant system failures* in other *vehicle* systems, and will respond appropriately.				
4	**High Driving Automation**	The *sustained* and *ODD*-specific performance by an *ADS* of the entire *DDT* and *DDT fallback* without any expectation that a *user* will respond to a *request to intervene*.	*System*	*System*	***System***	Limited
5	**Full Driving Automation**	The *sustained* and unconditional (i.e., not *ODD*-specific) performance by an *ADS* of the entire *DDT* and *DDT fallback* without any expectation that a *user* will respond to a *request to intervene*.	*System*	*System*	*System*	**Unlimited**

Figure 2.1: SAE levels of automation [31]

Currently available ADAS can be classified up to level 2, which means that the driver has the final responsibility for the vehicle and his interventions in longitudinal and lateral guidance always have priority compared to the actions of the ADAS.

2.1.1 Environment Recognition Sensors

To provide the described support for the driver, many ADAS use sensors for environment recognition. With the collected data, systems are able to evaluate the situation and provide information for the driver or perform active intervention in the vehicle guidance. The used type of sensor depends on the field of application. To determine distance and relative velocity to another vehicle, *Radio Detection and Ranging* (RADAR) and *Light Detection and Ranging* (LIDAR) sensors are used, whereas ADAS for e.g. lane-, and traffic sign detection are equipped with camera systems [38]. In this context it is mentionable that RADAR sensors have the main advantage of resistance to different weather conditions (fog, snow, rain) compared to LIDAR and camera systems. Ultrasound sensors are widely used to monitor the near environment of the car. Figure 2.2 gives an overview of sensor types used for common ADAS. Modern vehicles are equipped with

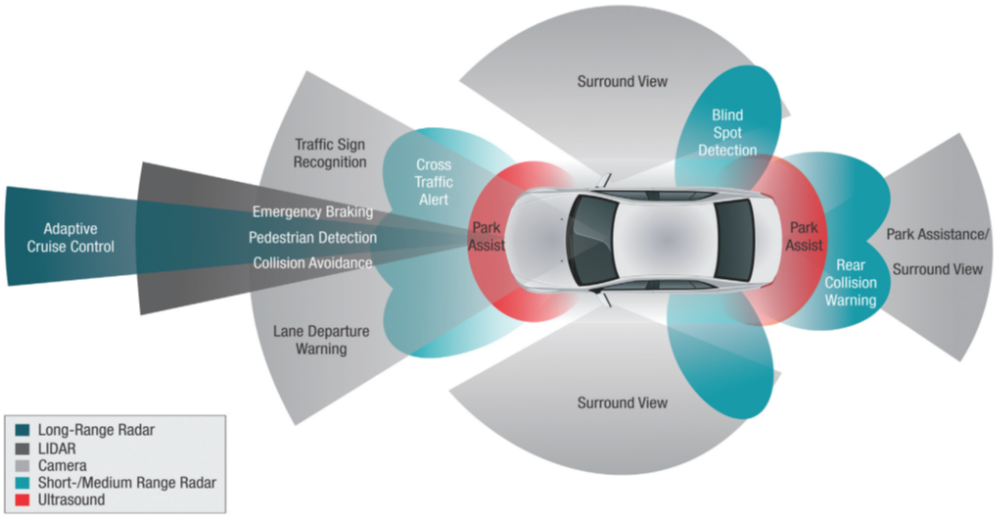


Figure 2.2: ADAS sensors for environment recognition [24]

ADAS that use more than one sensor to ensure observation of the environment in every direction on the one hand, and near and far field on the other hand. Therefore, sensor fusion algorithms are necessary in order to combine data of different sensors to generate a reliable and consistent representation of the environment. Based on the resulting map, possible actions of the ADAS are figured out by the step called situational analysis. Figure 2.3 gives an overview about the concept of sensor-fusion in ADAS.

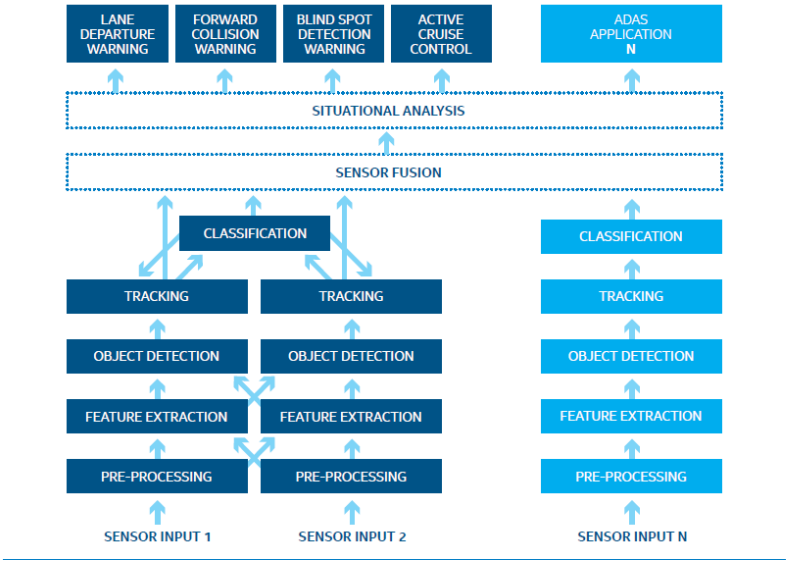


Figure 2.3: Concept of sensor-fusion [24]

Due to the fact that the main focus of this work lies on the parametrization and validation of a RADAR sensor model, RADAR sensors as they are used in automotive applications will be described in the following section. More information about other perception sensor types mentioned above can be found in literature such as Winner et al. [47].

2.2 Automotive RADAR Sensors

As described before, automotive RADAR sensors, in combination with other perception sensors are the basis for ADAS and ADS functions. Their objective is to

- detect objects and obstacles,
- determine their position (distance, angle),
- estimate their velocity relative to the sensor,
- define the state of the object (e.g. moving or stationary) and
- estimate their class [33].

The measurement principle of RADAR sensors is based on the transmission of electromagnetic waves and the interpretation of the reflected echo signal from the object regarding propagation time and frequency shift based on the Doppler Effect [43]. A more detailed description of different principles for distance, velocity and angle measurement follows in chapter 2.3.

Automotive RADAR sensors can be classified according to their maximum range into

Long-Range RADAR (LRR), *Mid-Range RADAR* (MRR) and *Short-Range RADAR* (SRR). Table 2.1 gives an overview about the main properties detection range, azimuthal field of view and elevation field of view, including some examples of use.

Table 2.1: Classification of automotive RADAR sensors [33]

RADAR Type	**LRR**	**MRR**	**SRR**
Range [m]	10-250	1-100	0.15-30
Azimuthal field of view [°]	±15	±40	±80
Elevation field of view [°]	±5	±5	±10
Applications	Adaptive Cruise Control	Lane Departure Warning, Cross Traffic Alert, Blind Spot Detection, Rear Collision Warning	Park Assist, Obstacle Detection

Initially, LRR sensors were operated within the 77 GHz frequency band (76-77 GHz) while MRR and SRR sensors used the 24 GHz band (22-26.625 GHz), which allows a high resolution. Due to limited transmission power and interference caused by its use from other radio services, the European Union has introduced the 79 GHz band (77-81 GHz) as successor. This band can be used for LRR, MRR and SRR applications [43],[13].

2.3 Principles of RADAR Technology

The goal of this chapter is to give a basic understanding of RADAR theory with focus on automotive RADAR sensors. Beside derivation of the RADAR equation and discussion of two important terms *Signal-to-Noise Ratio* (SNR) and *Radar Cross Section* (RCS), an overview about antennas including their essential parameters is given. The last section of this chapter focuses on the discussion of different methods for distance, velocity and angle measurement, as applied in today's RADAR sensors.

2.3.1 RADAR equation

The RADAR equation describes the physical relationship between transmitted and received power as a function of the system parameters of a RADAR system and the propagation phenomena of electromagnetic waves. It is used to estimate the achievable range for a given minimum *Signal-to-Noise Ratio* (SNR) at the receiver input. The equation can also be useful for designing a RADAR system since all device-specific parameters, apart from the Radar Cross Section (RCS) of the object to be detected, can be influenced by the developer [23].

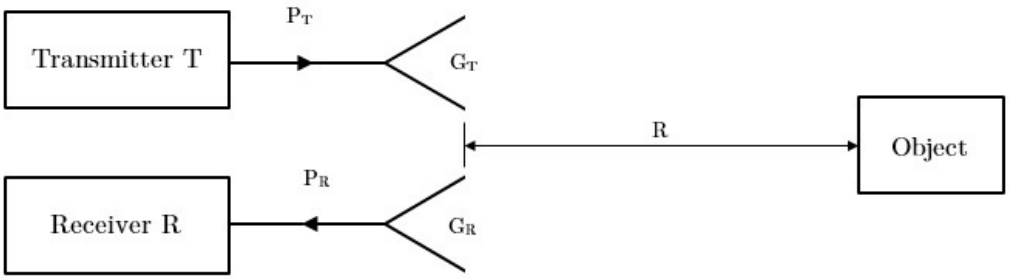


Figure 2.4: Derivation of the RADAR equation

Figure 2.4 shows the schematic geometry for the derivation of the RADAR equation. The transmitter T radiates the power P_T via the antenna. If P_T is radiated isotropic, the power density $D_T(r)$ on a spherical surface with the radius r can be expressed with

$$D_T(r) = \frac{P_T}{4 \cdot \pi \cdot r^2}. \tag{2.1}$$

In eq. (2.1) the term $1/(4 \cdot \pi \cdot r^2)$ corresponds to the transmission attenuation of electromagnetic energy caused by propagation. Usually RADAR systems use bundling antennas to increase the power density in one specific direction. The ratio of the maximum power density of such a directional antenna to the power density of an isotropic antenna is described with the antenna gain G. Under consideration of the gain G_T of the transmitting antenna the power density $D_T(R)$ reaching the object at the distance R is obtained.

$$D_T(R) = \frac{P_T \cdot G_T}{4 \cdot \pi \cdot R^2} \tag{2.2}$$

Multiplying D_T from eq. (2.2) with the RCS σ of the object results in the reflected power P_O of the object.

$$P_O = D_T(R) \cdot \sigma = \frac{P_T \cdot G_T \cdot \sigma}{4 \cdot \pi \cdot R^2} \tag{2.3}$$

Under consideration of the transmission attenuation, the power density $D_R(R)$ at the receiver with the distance R from the object can be defined.

$$D_R(R) = \frac{P_O}{4 \cdot \pi \cdot R^2} = \frac{P_T \cdot G_T \cdot \sigma}{(4 \cdot \pi \cdot R^2)^2} \tag{2.4}$$

With taking into account the effective area A_R of the receiving antenna, the received power P_R results in

$$P_R(R) = D_R(R) \cdot A_R = \frac{P_T \cdot G_T \cdot \sigma \cdot A_R}{(4 \cdot \pi)^2 \cdot R^4}. \tag{2.5}$$

The relation between effective area A_R and gain G_R of the receiving antenna is described by

$$A_R = \frac{\lambda^2}{(4 \cdot \pi)} \cdot G_R. \tag{2.6}$$

Applying eq. (2.6) in eq. (2.5) leads to the following expression for the received power P_R

$$P_R = \frac{P_T \cdot \lambda^2 \cdot \sigma \cdot G_T \cdot G_R}{(4 \cdot \pi)^3 \cdot R^4}. \tag{2.7}$$

In eq. (2.7) it can be seen that the received power of a RADAR sensor is proportional to the

- transmitted power P_T
- radar cross section σ of the object
- squared wavelength
- antenna gain of the transmitting G_T and receiving G_R antenna

and indirect proportional to the fourth potency of the distance R from the object.

The detection range R_{max} of a RADAR system defines the theoretically maximum distance, in which a detection of a target is possible. In eq. (2.8) the total system losses that occur in the transmitter/receiver path are considered with the factor L_tot. The factor L_{tot} includes the transmit loss L_t, the atmospheric loss L_a, the receiver loss L_r and the signal processing loss L_{sp} which are further described in [34].

$$R_{max} = \sqrt[4]{\frac{P_T \cdot G_T \cdot G_R \cdot \lambda^2 \cdot \sigma}{(4 \cdot \pi)^3 \cdot P_R \cdot L_{tot}}} \tag{2.8}$$

2.3.2 Signal to Noise Ratio (SNR)

For each application specific requirements on detection reliability and measurement accuracy are determined. In reality the received signal is always influenced by disturbances and noise, which have an impact on the detection reliability and measurement accuracy. To ensure a reliable and useful detection of the target by the RADAR sensor, the ratio of signal to noise (SNR) must not fall below a respective threshold [35].The following definitions are based on [44].

The thermal noise N describes the theoretical limit of the noise power at the input of the receiver. It results from the motion of electrons and is proportional to the temperature.

$$N_i = k \cdot T \cdot B_n, \tag{2.9}$$

including the Boltzmann's constant k (1.38 x 10^{-23} J/K), the temperature T in Kelvin (K) and the system noise bandwidth B_n in Hz. A standard room temperature of 290 K, results in the available noise power at receiver input of 4 x 10^{-21} W/Hz. The noise power at the receiver output will always be higher than predicted by eq. (2.9) due to the additional noise generated within the receiver. Taking that into account, the noise at

the receiver output N_o results from multiplying the ideal noise power N_i with the noise factor F_n and gain G of the receiver to

$$N_o = G \cdot F_n \cdot k \cdot T \cdot B_n. \tag{2.10}$$

The receiver gain can also be written as the ratio of signal output to signal input of the receiver ($G = S_o/S_i$). Solving eq. (2.10) for the noise factor F_n leads to expression

$$F_n = \frac{S_i/N_i}{S_o/N_o}. \tag{2.11}$$

The minimum detectable signal S_{min} at the receiver input to overcome the system noise can be obtained by rearranging eq. (2.10) to

$$S_{min} = k \cdot T_o \cdot B_n \cdot F_n \cdot SNR_{min}, \tag{2.12}$$

where SNR_{min} is the minimum signal to noise ratio the receiver processor needs to detect the signal. By equating the received power P_R, eq. (2.7) and S_{min}, eq. (2.12)

$$S_{min} = k \cdot T_o \cdot B_n \cdot F_n \cdot SNR_{min} = \frac{P_T \cdot \lambda^2 \cdot \sigma \cdot G_T \cdot G_R}{(4 \cdot \pi)^3 \cdot R^4}, \tag{2.13}$$

the expression for the maximum detection range R_{max} of the RADAR in eq. (2.8) can be extended to

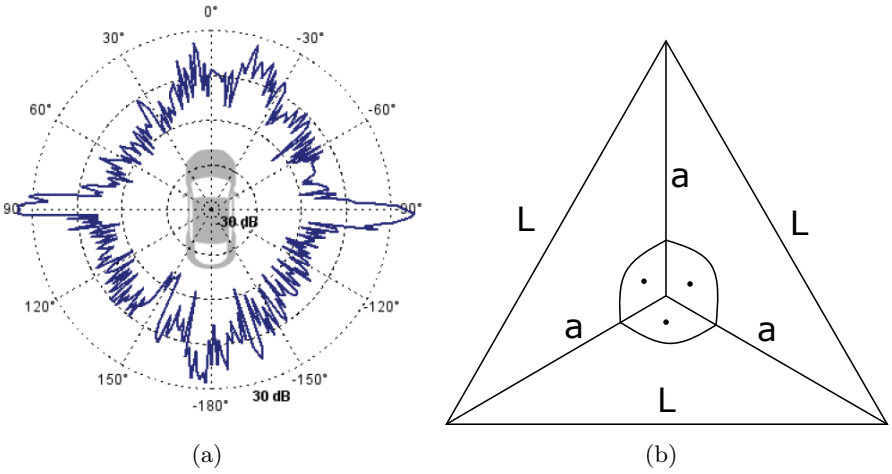
$$R_{max} = \sqrt[4]{\frac{P_T \cdot G_T \cdot G_R \cdot \lambda^2 \cdot \sigma}{k \cdot T_o \cdot B_n \cdot F_n \cdot SNR_{min} \cdot (4 \cdot \pi)^3 \cdot L_{tot}}}. \tag{2.14}$$

2.3.3 RADAR Cross Section (RCS)

Skolnik in [41] defines the *RADAR Cross Section* (RCS) σ of a target as the fictional area intercepting that amount of power which produces an echo at the RADAR equal to that from the target, when scattered equally in all directions. As eq. (2.5) shows, the RCS σ of the target is included in the received power P_R to the same extent such as the antenna gain or the transmission power P_T. Therefore, it is desirable to specify σ as exactly as possible. For real objects this involves considerable difficulties, since the RCS depends on various parameters [39]:

- dimension and form of target,
- direction to the antenna (aspect angle),
- frequency,
- polarisation,
- materials (electromagnetic properties) and
- surface structure,

whereby the dependence of RCS from real objects on the aspect angle seems to be significant as illustrated in fig. 2.5(a).



(a) (b)

Figure 2.5: (a) RCS depending on aspect angle [20] (b) Corner Cube Reflector [47]

In order to be able to perform meaningful, comparable and reproducible measurements or simulations, usually a standard target becomes defined. As standard targets certain geometric shapes such as sphere, cylinder or planar surfaces are used. Their RCS is defined by exact mathematical derivations. Particularly suitable as a standard target is the so-called corner cube reflector, fig. 2.5(b), which is composed of three surfaces which are perpendicular to each other [22]. A corner cube reflector has the property that its RCS is almost independent of the aspect angle and the polarization of the incident wave in limited angle ranges.

2.3.4 RADAR Antennas

As defined in [1], the antenna is the part of a transmitting/receiving system that is designed to radiate or to receive electromagnetic waves. In other words it is responsible for the transmission of electromagnetic energy into the environment and reception of the energy that has been reflected from a target [34]. There exists a large variety of antenna types for the respective fields of application which are well described in [34] and [3]. In automotive industry various types of antennas have been used in the development of RADAR systems. Mentionable in this context are patch or planar array antennas as well as antennas with lens or reflectors. The ones mentioned first are widely used in automotive RADAR systems because of their simple structure, small size and low costs. On the other side, insufficient efficiency in the used frequency band may influence their competitiveness in a negative way. The strength of lens or reflector antennas is the provision of high gain, which is beneficial for long range detection. Due to difficulties

with assembly and the bulky structure of these antenna forms, their applications are limited [48]. In chapter 2.3.6 further details about measurement principles of RADAR antennas used in automotive industries will be given.

2.3.4.1 Parameters of Antennas

The performance of RADAR antennas is characterized by different parameters. In this chapter the essential parameters

- Radiation pattern
- Directivity
- Antenna gain
- Beam width

will be described shortly to get a basic idea about them. More detailed information can be found in [34] and [3], whereon this section is based.

Radiation Pattern: The radiation pattern, also called power- or antenna pattern describes the radiation characteristics as a function of the direction. Due to the fact that antennas, to some extent, radiate energy in all directions, radiation patterns are actually three dimensional. In fig. 2.6(a) an example of a symmetrical three-dimensional radiation pattern is illustrated. It can be seen that the pattern consists of various parts, the so called lobes, which are classified into the main lobe in direction of maximum radiation, the side lobes and the back lobe in opposite direction of the main lobe. Side and back lobes are representing undesired radiation, which should be suppressed if possible. However, the radiation pattern is usually described with a horizontal plane (azimuth) and a vertical plane (elevation). In fig. 2.6(b) a two dimensional linear pattern (one plane of fig. 2.6(a)) is shown.

Directivity: The parameter directivity D is defined as the ratio of radiation intensity U in the main direction of the antenna to the averaged radiation intensity U_0 over all directions.

$$D = \frac{U}{U_0} \tag{2.15}$$

Antenna Gain: The antenna gain G is also a useful measure to describe the antenna performance. It is closely linked to the directivity, but in contrast the gain considers both the directional properties and efficiency of the antenna. The gain can be defined as the ratio of radiation intensity in the main direction to the intensity of an isotropic antenna without any losses, but with same input power. G is obtained by subtraction of internal losses from the maximum directivity $D_{\max}$.

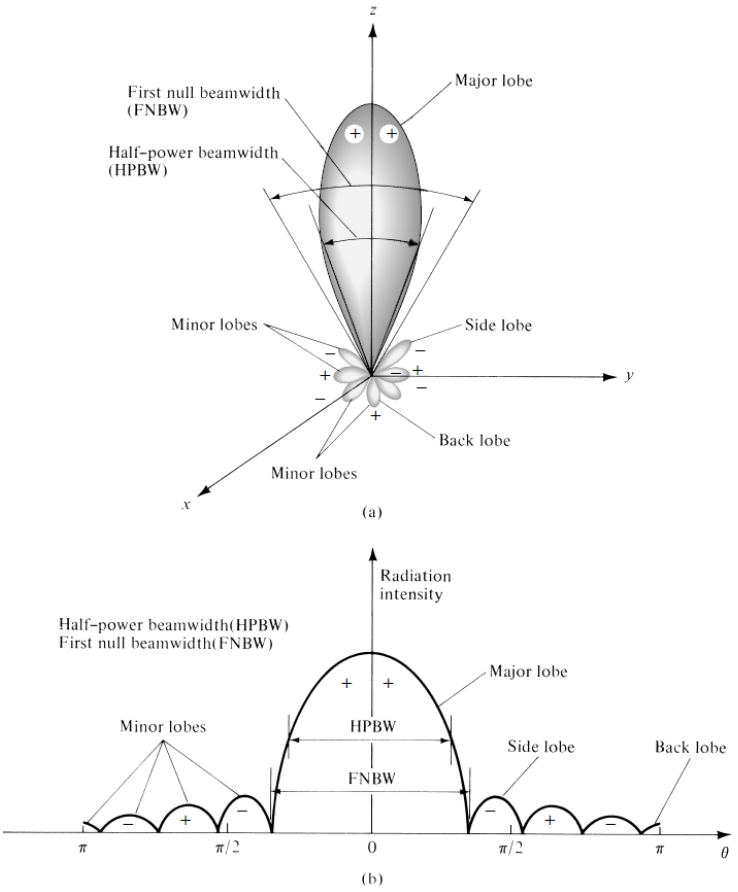


Figure 2.6: (a) Antenna pattern with lobes and beam widths (b) Linear pattern [3]

Beam Width: The beam width of an antenna defines the angular distance between two points on the opposite side of the main beam. Different types of beam width exist. The most frequently used one is the so called *Half Power Beamwidth* (HPBW), which is defined as the angle between two points in which the radiation intensity reaches half value. The second considerable beam width is the *First Null Beamwidth* (FNBW) which represents the angular distance between first nulls of the radiation pattern. HPBW as well as FNBW are illustrated in fig. 2.6.

2.3.5 Distance and Velocity Measurement

In this chapter the principles of distance and velocity measurement of RADAR sensors will be described. As mentioned earlier, RADAR measurements are based on the transmission of an electromagnetic wave and the detection of its echo signal. In general, the distance R to an object is determined by the travel time Δt, which is taken by the

RADAR wave to the target and back, and the speed of light c as illustrated in eq. (2.16) [41].

$$R = \frac{c \cdot \Delta t}{2} \tag{2.16}$$

Calculation of the target's relative velocity is based on the so called Doppler effect, which will be described in the following section.

2.3.5.1 Doppler Effect

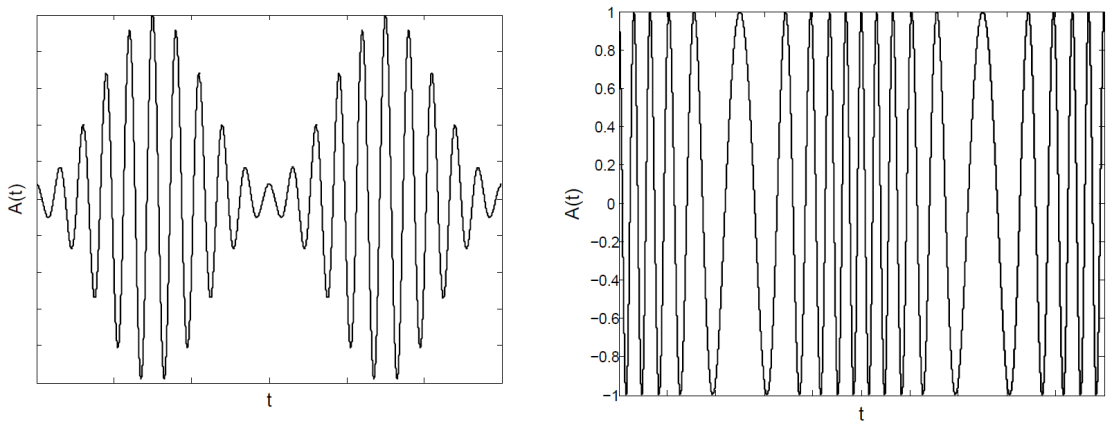
In 1842 the Austrian physicist Christian Doppler found out that electromagnetic waves discover a frequency shift if source and observer are moving relatively to one another. The same phenomenon occurs if the RADAR beam is reflected from an object which executes a relative movement to the RADAR. The Doppler Effect is expressed as the frequency change f_D (Doppler frequency or Doppler shift) between received and transmitted signal, which is proportional to the relative velocity $\dot{r}$ and inversely proportional to the wavelength $\lambda = c/f_0$.

$$f_D = -\frac{2 \cdot \dot{r}}{\lambda} = -\frac{2 \cdot \dot{r} \cdot f_0}{c} \tag{2.17}$$

The occurring frequency shift is positive if the object is approaching to the RADAR ($\dot{r} < 0$) and negative if they move away from each other ($\dot{r} > 0$)[47], [33].

2.3.5.2 Principles of Modulation

For distance and velocity measurement, two different measuring methods are used in automotive engineering. These are based on amplitude modulation and frequency modulation, which are illustrated in fig. 2.7. In both cases, a fixed frequency is used as a carrier signal. As described before, in the case of pulse based modulation, the distance to the object is determined by a time-of-flight measurement of the electric wave, i.e. the signal propagation time is measured. In contrast, in the frequency modulation method, a continuous wave is emitted and its wavelength is linearly changed over time. By evaluating the frequency and phase shift between the transmitted and received signal, it is possible to deduce the distance and relative velocity. The pulse method requires high-quality components due to the need of high sampling of the signal which would lead to high costs for automotive RADAR sensors. Therefore, frequency modulated systems are primarily used for automotive approaches, even if modern RADARs are able to combine both methods [29]. The following paragraphs, which are based on [47], [5], [22] and [27], give a short overview about the state of the art modulation methods for automotive RADAR sensors.



(a) Amplitude modulation example.

(b) Frequency modulation example.

Figure 2.7: Principles of modulation [21]

Pulse Modulation: The *Pulsed Doppler* RADAR is one of the simplest sensor types used in automotive industry. The main characteristic of a pulse RADAR is the successive transmission and reception, which is achieved by constant switching of the operating mode [27]. The sensor emits a pulse, the object reflects it and is therefore sent back to the receiver. The distance r is calculated by the relationship

$$r = \frac{c \cdot \tau}{2} \tag{2.18}$$

which was already mentioned before in eq. (2.16), in which τ is the propagation time of the signal and c is the speed of light. The relative velocity $\dot{r}$ is determined by using the Doppler Effect, which was already described before in eq. (2.17) with the emitted carrier frequency $f_0 = \frac{1}{T_0}$. By addition of carrier frequency f_0 and frequency change f_D the frequency of the received signal f_{DR} is obtained by

$$f_{DR} = f_0 + f_D \tag{2.19}$$

in which $f_{DR} = \frac{1}{T_{DR}}$, as illustrated in fig. 2.8. By rearranging eq. (2.17), the relative velocity $\dot{r}$ is expressed by

$$\dot{r} = -\frac{f_D \cdot \lambda}{2}. \tag{2.20}$$

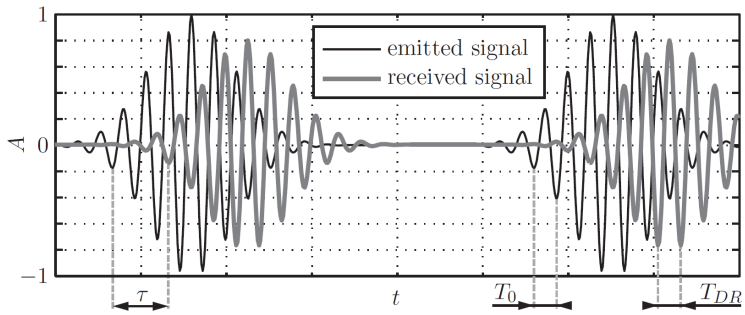


Figure 2.8: Transmitted and received signals of a Pulsed Doppler RADAR [5]

Frequency Modulated Continuous Wave: The main characteristic of the *Frequency Modulated Continues Wave* (FMCW) RADAR is the continuous emission and simultaneous reception of electromagnetic waves. In contrast to pulse RADARs they are able to evaluate the phase relation between transmitted and received signal. The FMCW RADAR sensor sends a time-continuous, frequency-modulated signal with the frequency $f(t)$ and the lenght T [22]. The sending frequency $f(t)$ can be described with the linear equation

$$f(t) = f_1 + \frac{f_2 - f_1}{T} t, \tag{2.21}$$

in which f_1 is the minimum and f_2 the maximum frequency within $f(t)$ is modulated. In fig. 2.9 three generic frequency ramps, as emitted by a FMCW RADAR sensor, are shown as an example. Due to the finite propagation speed of electromagnetic waves, as described in eq. (2.18), a frequency difference Δf between emitted and received signal occurs.

$$\Delta f = f_1 - f(t) = -\frac{f_2 - f_1}{T} \frac{2r}{c} \tag{2.22}$$

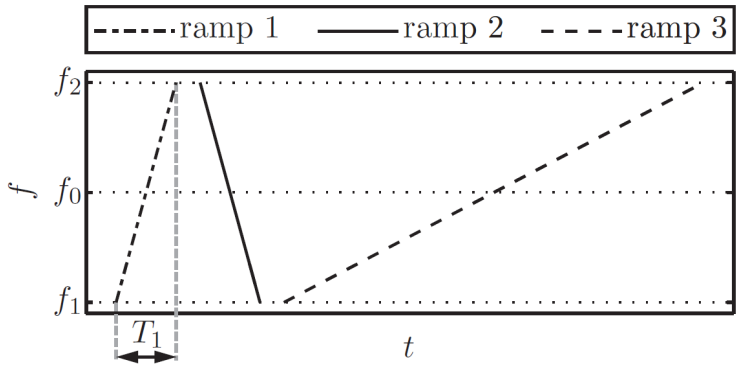


Figure 2.9: Three frequency ramps $f(t)$ of an FMCW RADAR [5]

Moving objects additionally cause a frequency shift of the reflected signals based on the Doppler Effect, as described in eq. (2.17). The resulting receiving frequency Δf_R of an object in the distance r with relative velocity $\dot{r}$ is therefore given with

$$\Delta f_R = -\frac{f_2 - f_1}{T}\frac{2r}{c} - \frac{2\dot{r}f_0}{c}. \tag{2.23}$$

To detect a single object, at least two different ramps are needed. For the detection of more objects three or more ramps are required to avoid the detection of unreal objects. To solve the ambiguity of measured Δf_R regarding distance and velocity of the target, all received frequencies are linked to a line in the $r - \dot{r}$ plane of the object. The desired solutions for distance and relative velocity result in the points of intersection between the different ramps [25]. In fig. 2.10 an example of an $r - \dot{r}$ plane with four objects and three ramps is illustrated.

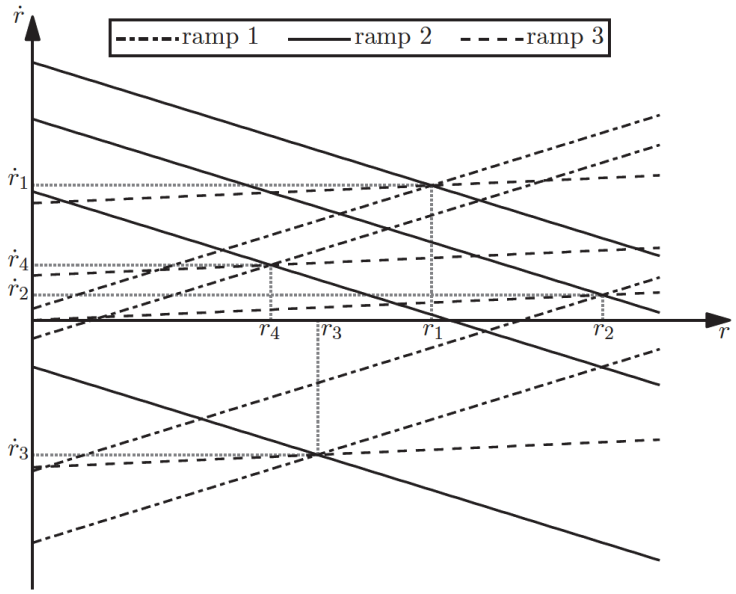


Figure 2.10: Generic $r - \dot{r}$ plane of an FMCW RADAR [5]

Frequency Shift Keying: *Frequency Shift Keying* (FSK) RADAR sensors belong to the group of continues wave RADARs. The essential difference to the FMCW RADAR is the fact that the sending frequency is not modulated linearly, but shifts between the constant frequencies $f_{1,2} = f_0 \pm \frac{\Delta f_{FSK}}{2}$. In this RADAR principle the phase difference $\Delta\delta$ between the received signals $A_{1,2} = \sin(2\pi f_{1,2}t)$ is calculated with

$$\Delta\delta = 2\pi(f_2 - f_1)\tau. \tag{2.24}$$

By implementing eq. (2.18) in eq. (2.24) the expression for the phase difference can be rewritten to

$$\Delta\delta = \frac{4\pi}{c}(f_2 - f_1)r. \tag{2.25}$$

Rearranging eg. (2.25) leads to the mathematical expression of the distance r to an object with

$$r = \frac{\Delta\delta \cdot c}{4\pi(f_2 - f_1)}. \tag{2.26}$$

Due to the fact that the frequency step Δf_{FSK} is a lot less than f_0 ($\Delta f_{FSK} << f_0$), the relative speed $\dot{r}$ can be found by using the frequency shift f_D of the Doppler Effect as described in eq. (2.17). In fig. 2.11 the amplitudes of the normalized signals $A_{1,2}$ and $\Delta A = \sin(\Delta\delta)$ are shown.

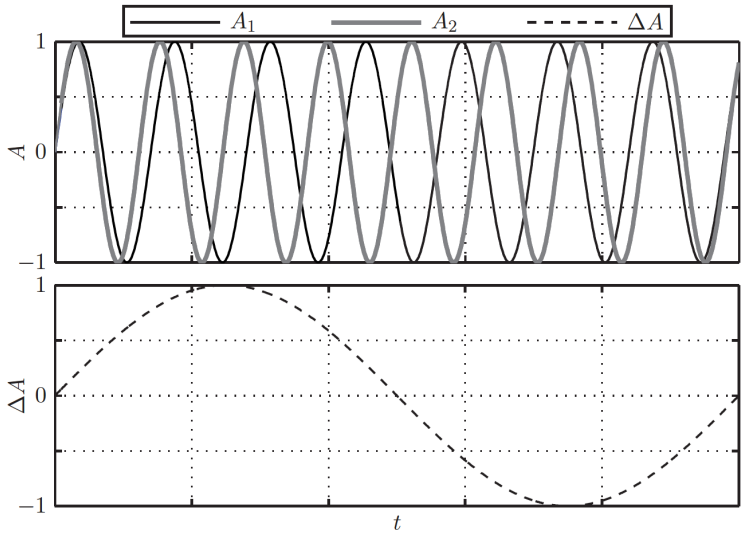


Figure 2.11: Principle of distance measurement by phase comparison [5]

Frequency Modulated Shift Keying: *Frequency Modulated Shift Keying* (FMSK) is a combination of the principles used in FSK and FMCW RADAR sensors. The transmitted wave consists of two frequency modulated signals A and B. These two chirps are transmitted in a merged sequence. The frequency modulated step A is followed by the time and frequency shifted step B, as illustrated in fig. 2.12. The FMSK principle leads to the advantages of short measurement time and also high resolution and accuracy for range and velocity estimation in multi target environment. Further details as well as mathematical contexts of this modulation principle are well described in [26].

Chirp Sequence Modulation: The *Chirp Sequence Modulation* or *Pulse Compression* combines the advantages of all sensor principles described before. It consists of a sequence of identical linear frequency ramps, as described in eq. (2.21), which are repeated in short intervals of duration T_{Chrip}. Due to the short intervals, the frequency shift based on the Doppler Effect f_D is negligibly compared to the frequency change of the chirps. This

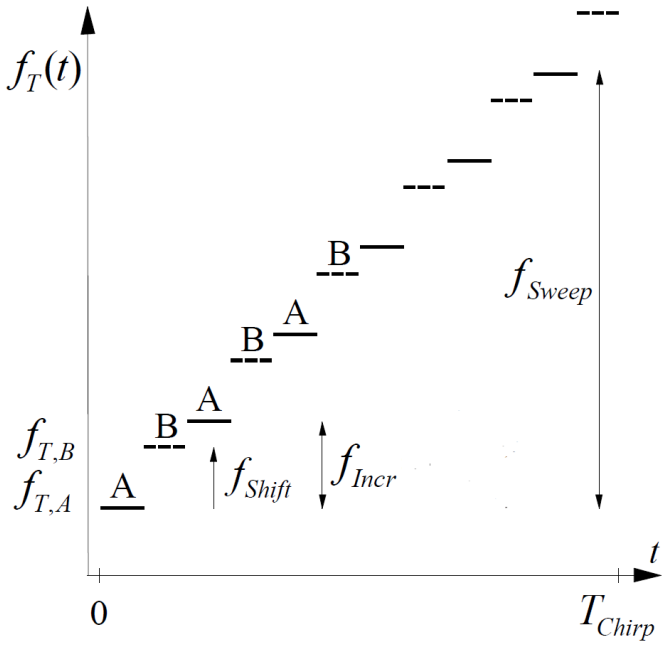


Figure 2.12: FMSK waveform principle [26]

means that the received frequency is linked to the object's distance R, which is calculated according to eg.(2.22) with

$$r = -\frac{c \cdot T_{Chrip} \cdot \Delta f}{2(f_2 - f_1)}. \tag{2.27}$$

The calculation of the relative velocity is based on the frequency shift f_D caused by the Doppler Effect. Instead of measuring at a single point, the phase difference between chirps over a longer period is observed to determine f_D.

2.3.6 Angle Measurement

Beside measurement of distance and relative velocity between target and RADAR sensor, the target's angle relative to the RADAR sensors bore-sight in horizontal (azimuth) and vertical (elevation) direction is of interest in order to identify the target's position in the *Field of View* (FOV). The detection of a target requires, that it is located within the antenna beam. The beam width and therefore the directivity of the antenna is given by its aperture. The used width of the antenna beam depends on the designated application. Antennas for ACC systems, which have to detect faraway objects, use narrow beams, whereas for environment detection in near range antennas with wide beams are used [15]. For automotive applications different methods for angle measurement are state of the art. These methods are briefly described in the following paragraphs.

2.3.6.1 Scanning

Scanning RADAR sensors use the easiest principle for angle determination. A beam deflection unit or a planar antenna is quickly pivoted and a narrow beam oscillates through the whole detection range in azimuthal direction within one measurement cycle, as illustrated in fig. 2.13. The beam is moved continues over the detection range, but then the measured values are linked to a discrete angular position. The target's angle in azimuthal direction is indicated by the local intensity maximum of the reflected energy $P(\varphi)$. Beside high accuracy due to the narrow beam, the ability to separate objects regarding their angle is a main advantage of the scanning method [47].

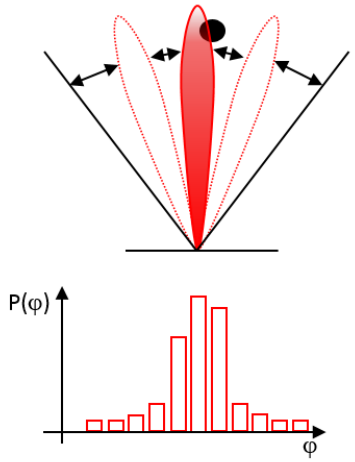


Figure 2.13: Scanning principle for angle determination [25]

2.3.6.2 Monopulse

The *Monopulse Method* is based on a dual antenna arrangement for receiving the reflected signals, whereby the sending beam is emitted by a separate single antenna. An angle measurement can be realised by comparing amplitudes (*Amplitude Monopulse*) or phase differences (*Phase Monopulse*) between the simultaneously measured signals of the individual antennas [47],[25].

Amplitude Monopulse: The *Amplitude Monopulse* method uses two overlapping beams with slightly different directions, see fig. 2.14. Due to the symmetrical arrangement to the boresight, the resulting amplitudes A_1 and A_2 of the received signals for a single target are different from each other, unless the target is located in the center line of the beams. This issue can be used to estimate the azimuth angle. In a signal processing

step, sum $|\Sigma|$ and difference $|\Delta|$ of the received signals are generated. The error angle $\varepsilon(\varphi)$ is defined with

$$\varepsilon = \frac{|\Delta|}{|\Sigma|} = \frac{|A_1| - |A_2|}{|A_1| + |A_2|}, \tag{2.28}$$

which can be used as a measure for the azimuth angle [47], [15].

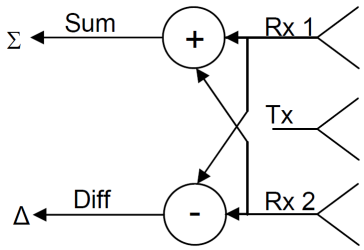


Figure 2.14: Amplitude Monopulse principle, [25]

Phase Monopulse: The *Phase Monopulse* method, well described in [15], is also based on analysing the sum $|\Sigma|$ and difference $|\Delta|$ of the received signals. Compared to the amplitude monopulse method, instead of different antenna beams, two closely arranged receiving antennas with the distance d are used, fig. 2.15. Due to the slight difference

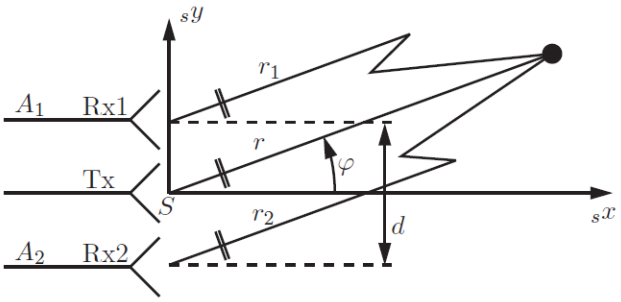


Figure 2.15: Phase Monopulse principle, [5]

between the lengths r_1 and r_2, the received signals A_1 and A_2 have the same amplitude, but a different phase. The phase difference $\Delta\delta$ depends on the azimuthal angle φ, the wavelength λ and distance d between the antennas,

$$\Delta\delta = \frac{2\pi}{\lambda} \cdot (r_2 - r_1) = \frac{2\pi}{\lambda} \cdot d \cdot \sin(\varphi). \tag{2.29}$$

To determine $\Delta\delta$, the quotient of $|\Delta|$ and $|\Sigma|$ of the complex receiving signals A_1 and A_2 is used,

$$\frac{|\Delta|}{|\Sigma|} = \tan\left(\frac{\Delta\delta}{2}\right), \tag{2.30}$$

where the complex signal A_2 is described as

$$A_2 = A_1 \cdot e^{-i\Delta\delta} \tag{2.31}$$

and the difference Δ and sum Σ reads

$$\Delta = A_1 - A_2 = A_1 \left(1 - e^{-i\Delta\delta}\right) \tag{2.32}$$

$$\Sigma = A_1 + A_2 = A_1 \left(1 + e^{-i\Delta\delta}\right) \tag{2.33}$$

Including eq. (2.30) in eq. (2.29), the azimuthal angle φ can be determined by

$$\varphi = \arcsin\left[\frac{\lambda}{\pi d} \arctan\left(\frac{|\Delta|}{|\Sigma|}\right)\right]. \tag{2.34}$$

2.3.6.3 Multibeam

The device for the *Multibeam Method* emits several side by side arranged beams. The evaluation of the angle is based on comparison of normalized sensor-specific antenna characteristics. In fig. 2.16 the basic principle of this method is illustrated. On top, the overlapping beams, in the middle the azimuthal angle characteristic and at the bottom the reflecting energy $P(\varphi)$ of the single beams are shown.

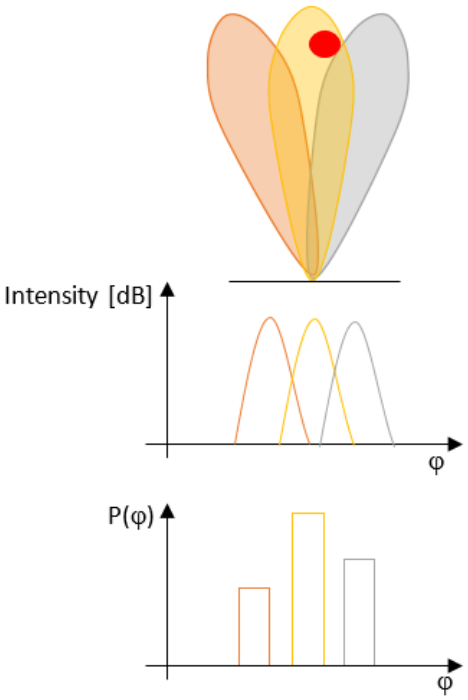


Figure 2.16: Multibeam principle for angle determination [25]

2.4 Virtual Development Process

The main parts of ADAS functions are usually implemented in software. Therefore in most cases the so called *V-Model* is used within the development process of ADAS, which is already approved in the field of software engineering [47]. Figure 2.17 illustrates the application of the V-Model in the development and integration process of ADAS at MAGNA STEYR Fahrzeugtechnik AG & CoKG. It can be seen that the two basic branches *Development* on the left side and *Validation* on the right side are extended with the additional branch *Virtual Validation* on full-vehicle level. This extension is done with the objective to assess the requirements and the functional concept in the early development phase, before validating ADAS functions with real test drives [6]. This gives the opportunity to reduce the execution of time-consuming, expensive and partly safety critical test drives with real vehicles.

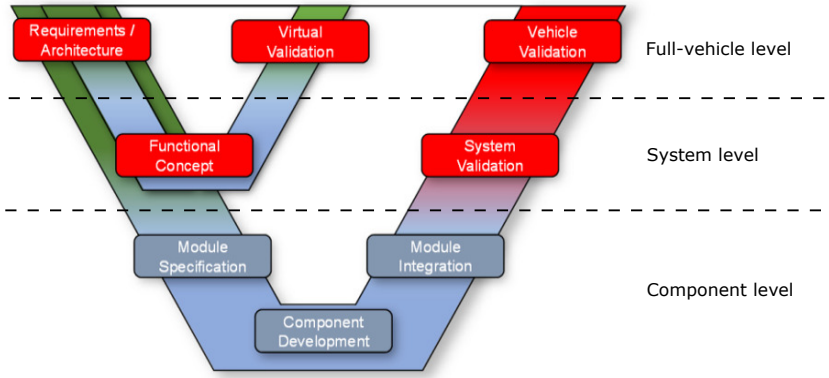


Figure 2.17: V-Model used at MAGNA STEYR Fahrzeugtechnik AG & CoKG

Driving simulation tools range from commercial software for personal computers, over *Hardware in the Loop* (HiL) test benches to expensive and high complex *Driver in the Loop* (DiL) driving simulators at research institutes, such as the NADS-1 driving simulator from the University of Iowa, illustrated in fig. 2.18(a). The Institute of Automotive Engineering at Technical University of Graz developed a driving simulator with the focus on testing of the *Human Machine Interface* (HMI), see fig. 2.18(b). In addition to that, research institutes in cooperation with automotive industry make huge effort in development of *Vehicle in the Loop* (ViL) test benches for testing ADAS and ADS functions. As shown in fig. 2.19, the *Electronic Control Unit* (ECU) in ViL test benches is either provided with simulated data from a sensor model or with data from real sensor hardware, which is stimulated with signals generated by an over the air target stimulator. In the second case, the sensor model provides environment and traffic data as input for the target stimulation.

(a) NADS-1 driving simulator [45] (b) Simulator at TU Graz [32]

Figure 2.18: MiL driving simulators at research institutes

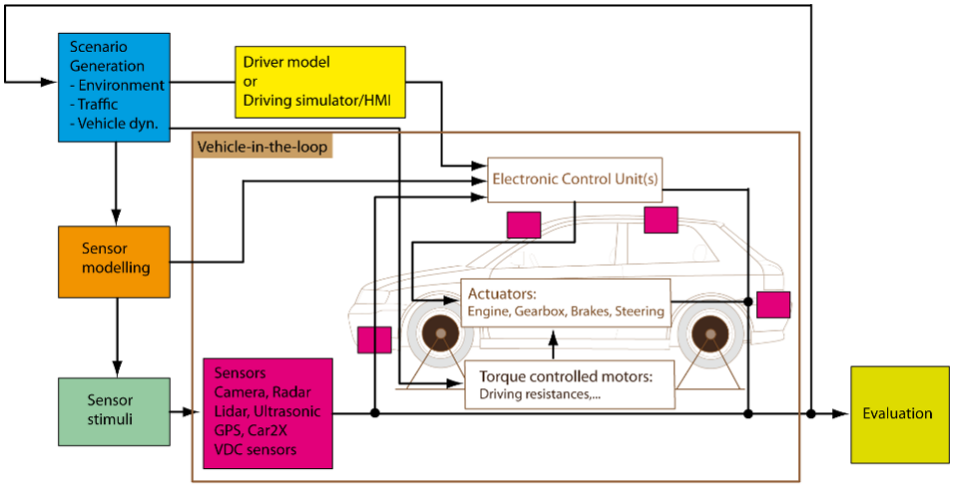


Figure 2.19: ViL block diagram

The validity of simulation-based evaluations significantly depends on the quality of simulation. In other words, it depends to which extent real and simulated scenarios lead to a comparable performance of the tested system. While vehicle models already reach a high degree of realism, modelling the vehicle environment and in particular its detection with sensors is still a great challenge [40].

As already mentioned in a previous section of this chapter, RADAR sensors for environment recognition are a essential part of ADAS. The provided data of sensor models is used as input for algorithms of ADAS functions in the virtual simulation environment or in case of sensor stimulation as input for signal generation. Therefore, it is necessary to use valid virtual sensor models that are able to represent the specific characteristics of real RADAR sensors, including real sensor effects, in sufficient detail [37], [46].

2.4.1 RADAR Sensor Models

Modelling of automotive RADAR sensors is complex due to occurring effects such as interference, multipath propagation, obstruction, latency and attenuation, just to mention some of them [42]. The main challenge in this area is to find a compromise between efficiency, real-time capability, and fidelity of the simulation [8]. Nowadays various different approaches for RADAR sensor models already exist, see fig. 2.20, and a number of researches have been published in this field [6], [8], [40] and [46], will be described in the following section.

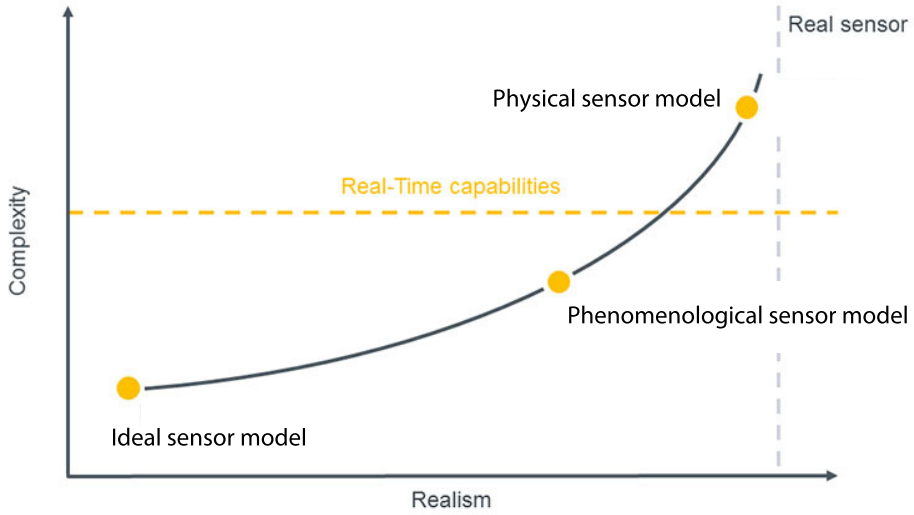


Figure 2.20: Sensor models for virtual validation, adapted from [4]

2.4.1.1 Ideal Sensor Model

Almost every commercial simulation tool for virtual test drives is equipped with sensor models for environmental recognition, such as the object sensor implemented in IPG CarMaker®, see fig. 2.21. In most cases *Ideal Models*, also called *Black Box Models*, based on position data from traffic simulation, are used. These models describe the dynamic behaviour of objects in a perfect way, without considering real radar effects and therefore slight fidelity. On the other hand, these models bring the advantage of easy implementation, parametrization and low computational effort. As a consequence of the pros and cons, ideal sensor models are used in early stages of development and serve as a basis for more complex sensor models.

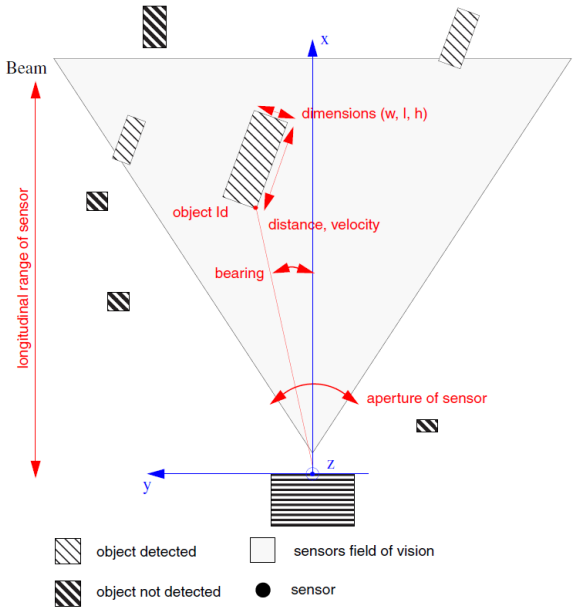


Figure 2.21: Ideal object sensor model in IPG CarMaker® [19]

2.4.1.2 Physical Based Sensor Model

In contrast to ideal models, *Physical Based Models* or *White Box Models* are available. They try to reproduce the sensor's internal behaviour and its interaction with the environment based on physical relations. The aim is to model physical effects of electromagnetic wave propagation, the object's reflection properties and all electronic components within the sensor [8]. Therefore this type of model is in high fidelity and provides the closest-to-reality result in simulation. Detailed modelling of the environment to observe RADAR effects such as multipath propagation or interference is computation-extensive and makes physical-based models not real-time capable [46]. Due to strong reliance of physical effects to the used technology, these types of models are specialised and their adaptability to different sensor technologies is limited [40].

2.4.1.3 Probabilistic or Phenomenological Sensor Model

To face the dilemma between fidelity and efficiency, researchers developed *Probabilistic or Phenomenological Models*, in literature also named *Grey Box Models*, which can be classified on a level between the two models described above. These types of sensor models approximate the characteristics of real world sensors by modification of data from an ideal sensor.

Schubert et al [40] describe an approach for a probabilistic model, in which data from an ideal sensor model is superimposed with an error signal, in which stochastic error values

are generated by an *Probability Density Function* (PDF). With this approach, typical sensor errors such as noise, false-positive and false-negative detections can be simulated in a sufficient way.

A further approach is described by Bernsteiner et al in [6]. Researchers from Institute of Automotive Engineering at Technical University of Graz in cooperation with MAGNA STEYR Fahrzeugtechnik AG & CoKG developed a phenomenological RADAR sensor model for the early concept phase which is easily parametrizable with data sheets. It is designed with the objective to run simulations time efficient while still taking real RADAR effects such as latency and signal losses into account. As illustrated in fig. 2.22 the model is divided into three modules: *Geometric model*, *Real characteristics* and *Signal processing*.

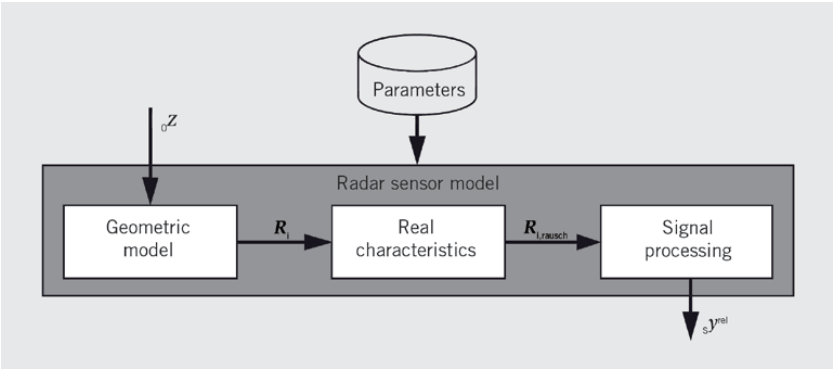


Figure 2.22: Modules of the phenomenological sensor model [6]

In the *Geometric model* the sub-steps coordinate transformation from global to sensor coordinates, verification if objects are in the sensor's FOV, possible obstruction of objects and definition of object's reference points are included. Within the *Real characteristics* module, the ideal signal is superimposed with real-world sensor effects, which include signal losses, measurement noise and weather dependent minimum SNR to detect an object. These properties are modelled by the use of simple mathematical relations with the goal to keep the computational effort low. In the last module, *Signal processing* the noisy signal gets filtered with an *Extended Kalman Filter*, the relevant target is selected and then communicated to the ADAS algorithm..

In [46], Wheeler et al. present a rather new approach for RADAR sensor models. The principle of the described sensor model works on the basis of deep-learning, connected to real world data. The aim of manipulating the ideal object list in real-time is realised by deep neural networks, which are trained with real data.

To summarize, phenomenological sensor models consider many properties of real RADAR sensors due to simplified representation of physical effects, while staying real-time capable and being parametrizable with low effort. In addition to the described models, further approaches for sensor modelling are presented in [4] and [12].

2.4.2 Validation of Sensor Models

As described in the previous sections, various approaches for sensor models usable in development and validation processes for ADAS already exist. Since the output of these virtual sensor models is the basis for the algorithms of ADAS functions, they have to be validated and compared to real sensor data, before being implemented into a simulation framework [37].

In literature different methodologies for model validation can be found. In [30], Oberkampf and Trucano define the term *validation* as a method to evaluate whether a model represents the real world accurately, under consideration of its intended application. As illustrated in fig. 2.23, they propose an approach composed of three steps. In the *first step* simulation results and data determined in experiments are compared by the use of a validation metric to quantify accuracy of the model. In many cases the term validation is only defined with this step. However, the authors think that two further steps are necessary to ensure a model's validity. Within the *second step* the model is inter-, or extrapolated to conditions which correspond to its intended use. This step has to include prediction uncertainties, which are compared with accuracy requirements in *step three* to decide if the prediction accuracy of the model is adequate or not.

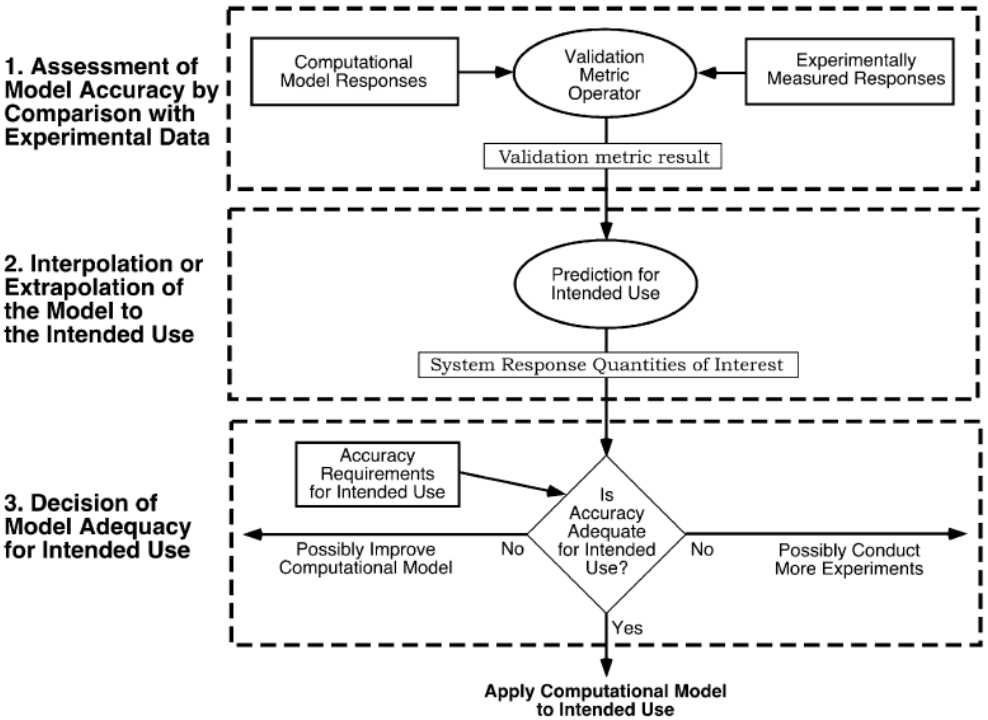


Figure 2.23: Three steps of model validation [30]

Roth et al. in [36] describe a methodology for validating virtual sensor models in the development of ADAS. In contrast to the generic method presented above this procedure, consisting of six steps, is related to automotive application. The six steps, as illustrated in fig. 2.24, are:

1. Performing real test drives based on a manoeuvre catalogue and recording data from perception and position reference sensor.
2. Generation of a virtual test scenario for the simulation framework based on the position reference data.
3. Parametrization of perception sensor models according to the parameters of the real world sensors used in the test drives in step 1.
4. Running of virtual test drive simulations based on the scenario generated in step 2.
5. Recording of synthetically generated perception sensor model data during the virtual test run.
6. Comparison of data recorded from real test drive with recorded synthetic data of virtual test drive.

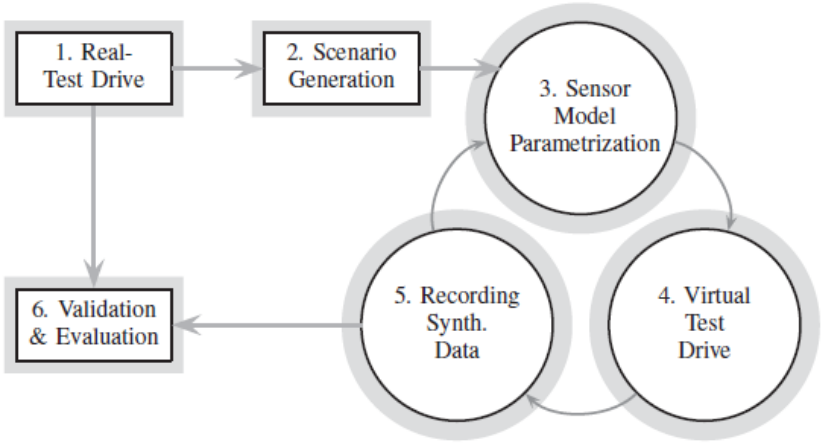


Figure 2.24: Validation of perception sensor models according to [36]

Even though both methods presented deal with the validation of sensor models, each approach is clearly different. While the first one describes a general guideline and suggestion for a validation process, the second method focuses on the question how synthetic model data can be captured. Schaermann et al. mention in [37] that only a direct, qualitative comparison of real test data and synthetic sensor model data, as described by Roth et al. is not sufficient for validation. Therefore, the authors developed a validation method that includes, besides direct comparison of data from experiments and virtual simulation, evaluation of outputs on higher levels of the overall system, filled with real

and modelled data. Figure 2.25 visualises the described method, consisting of seven steps, in which the subscripts $_{R}$ and $_{S}$ stands for real-test and simulation.

1. Performing of real test drives and capturing output data from perception sensors and environment model.
2. Recording of reference data (Ref_R), sensor data represented as object list (OL_R) or raw data (Raw_R) and high-level fusion (HLF_R) or low-level fusion (LLF_R) data from the environment model.
3. Provision of reference data for the simulation in order to generate a virtual scenario according to the real-test drive.
4. Re-simulation of the real-test drive using sensor models that provide synthetic data for the environment model, which is similar to the one implemented in the test vehicle.
5. Recording of synthetic data from perception sensor models (OL_S, Raw_S) and environment model output-data (HLF_S, LLF_S) during simulation.
6. Implementation of suitable metrics to compare recorded data from real-test drives and synthetic data from simulation.
7. Decision making whether the examined model is valid or not.

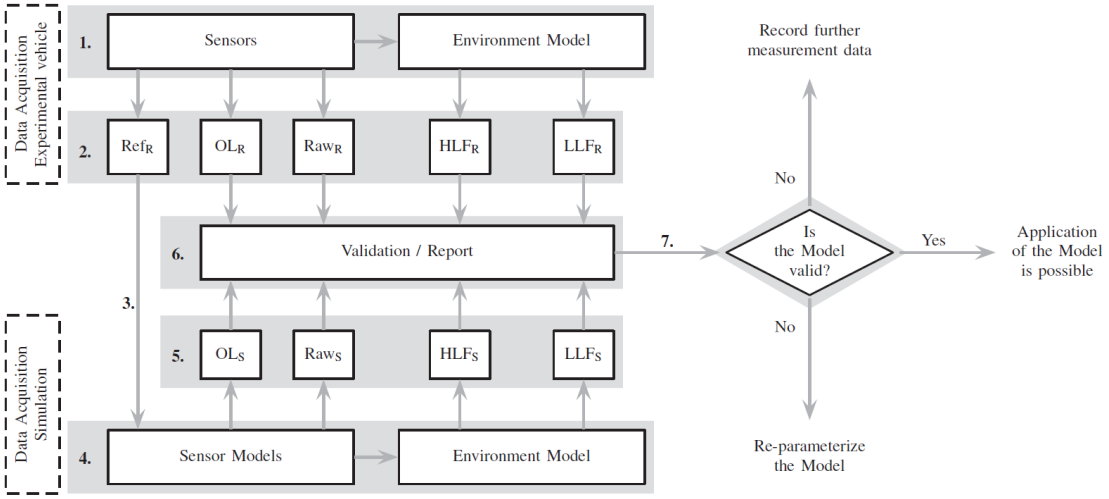


Figure 2.25: Validation of perception sensor models according to [37]

The prerequisite for this method is the availability of a valid environment model for the simulation framework, which represents the environment model used in reality in a sufficient way.

3 Methodology

This chapter is divided into two main parts. The first part deals with the modelling process of the test track on the proving ground at MAGNA STEYR Fahrzeugtechnik AG & CoKG. Starting with high accurate GPS measurement of the test track, over transformation of the coordinates into a usable coordinate system and calculation of the lane widths, it finally leads to the modelling of the test track within the software tool IPG CarMaker®.

Within the second part of this chapter, a method for the parametrization and validation of a physical RADAR sensor model is presented. The impact of available and adjustable parameters within the HiFi RADAR sensor model on its output is examined. This is realised by reproducing the driven manoeuvre of the real test drive, in particular trajectory of participating vehicles, in a virtual scenario and comparison of recorded data from a real RADAR sensor during real test drives with synthetically generated data from the virtual sensor model during simulation.

3.1 Modelling of the Test Track

One main task within this thesis is to build up a model of the test track at the proving ground of MAGNA STEYR Fahrzeugtechnik AG & CoKG, see fig. 3.1. The model is needed for future virtual testing applications of camera based ADAS functions. Therefore, high accuracy of geometrical properties of the test track and of the road markings is required. To obtain a satisfactory result, high effort has been put in this part of the thesis, consisting of various process steps, see fig. 3.2, which are described in the following part of the thesis.

Figure 3.1: Test track of MAGNA STEYR Fahrzeugtechnik AG & CoKG

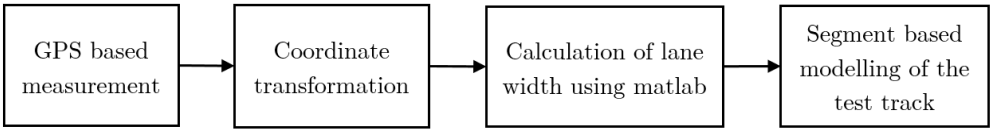


Figure 3.2: Modelling process steps

3.1.1 GPS Measurement of the Test Track

The basis for the model is the *Global Positioning System* (GPS) based measurement of the test track including road markings, which was carried out in cooperation with researchers from Institute of Geodesy at TU Graz.

(a) (b)

Figure 3.3: Measurement of test track

3.1.1.1 Measurement Setup

GPS points: Two GPS-receivers have been placed at random points of the test track. Moreover, one GPS-receiver was placed at a known position at the roof of Institute of Geodesy. By applying this method, high measurement accuracy is possible compared to common single point positioning. The prerequisite for this is that the receivers (on the roof and on the test track) observe the same satellites at the same time, whereby accuracy increases with the duration of observation. The GPS-based position of these devices is used as fix-points which describe the reference coordinates for the terrestrial measurement of the test track by the use of a theodolite and a reflector.

Tachymetry: The detail points (distance and direction) of the test track were measured with a total station (theodolite) from different viewpoints. A traverse (polygonal line) connects the individual viewpoints and GPS points with each other and ensures that all detail points are located in the same local coordinate system.

Net adjustment, Transition from local to global coordinate system: The coordinates of the detail points are calculated with the commercial software Geosi VERM®, whereby previously determined GPS points are entered as reference points. Thus, a transition from the local to the global *World Geodetic System 1984* (WGS84) coordinate system is possible.

Measurement devices and software:

- Theodolite: Leica TPS 1200 (TCRA 1201), see fig. 3.4(a)
- GPS receiver: Ashtec® Z-Xtreme™, see fig. 3.4(b)
- Software: Geosi VERM®

Further details of the used devices and software, as well as technical descriptions, can be found in the reference manuals [2], [11] and [18].

(a) (b)

Figure 3.4: Measurement devices

As a result of measurement and data analysis, the geographical coordinates longitude (λ) and latitude (φ) of 519 measurement points in the WGS84 coordinate system are obtained. They are the basis for oncoming steps.

3.1.2 Coordinate Transformation

Since the available points in ellipsoid WGS84 coordinates can not be directly used as input for modelling the test track in the simulation tool, they have to be transformed into plane *Gauß-Krüger* coordinates. Within the transformation process several sub-steps are necessary.

1. Transformation from ellipsoid *WGS84* coordinates φ_{WGS}, λ_{WGS} and h_{WGS} into cartesian coordinates X_{WGS}, Y_{WGS} and Z_{WGS}.

$$X_{WGS} = \left(\frac{c}{V} + h\right) \cdot \cos(\varphi_{WGS}) \cdot \cos(\lambda_{WGS}) \tag{3.1}$$

$$Y_{WGS} = \left(\frac{c}{V} + h\right) \cdot \cos(\varphi_{WGS}) \cdot \sin(\lambda_{WGS}) \tag{3.2}$$

$$Z_{WGS} = \left(\frac{b}{V} + h\right) \cdot \sin(\varphi_{WGS}) \tag{3.3}$$

$$c = \frac{a^2}{b} \tag{3.4}$$

$$V = \sqrt{1 + \varepsilon'^2 \cdot \cos^2(\varphi_{WGS})} \tag{3.5}$$

$$\varepsilon'^2 = \frac{a^2 - b^2}{b^2} \tag{3.6}$$

   In the given formulas ε' is the second numerical eccentricity and a and b represent the lengths of semi-axes of the reference ellipsoid, in this case *WGS84*.

2. *Helmert transformation* of the cartesian coordinates in the *WGS84* ellipsoid to cartesian coordinates X_B,Y_B and Z_B in the *Bessel* ellipsoid.

$$\begin{bmatrix} X \\ Y \\ Z \end{bmatrix}^B = \begin{bmatrix} c_x \\ c_y \\ c_z \end{bmatrix} + \mu \cdot \begin{bmatrix} 1 & r_z & -r_y \\ -r_z & 1 & r_x \\ r_y & -r_x & 1 \end{bmatrix} \cdot \begin{bmatrix} X \\ Y \\ Z \end{bmatrix}^{WGS} \tag{3.7}$$

$$\mu = 1 + \frac{m}{10^6} \tag{3.8}$$

   In eq. (3.7) c_x, c_y and c_z are the translation parameters, r_x, r_y and r_z are the rotation parameters and μ stands for the scale factor.

3. Transformation from cartesian coordinates X_B, Y_B and Z_B to ellipsoid coordinates φ_B, λ_B and h_B in the *Bessel* ellipsoid.

$$\varphi_B = \arctan\left(\frac{Z_B + \varepsilon'^2 \cdot b \cdot \sin^3\vartheta}{p - \varepsilon^2 \cdot a \cdot \cos^3\vartheta}\right) \tag{3.9}$$

$$\lambda_B = \arctan\frac{Y_B}{X_B} \tag{3.10}$$

$$h_B = \frac{p}{\cos\varphi_B} - \frac{c}{V} \tag{3.11}$$

$$p = \sqrt{X_B^2 + Y_B^2} \tag{3.12}$$

$$\vartheta = \arctan\left(\frac{Z_B \cdot a}{p \cdot b}\right) \tag{3.13}$$

$$\varepsilon^2 = \frac{a^2 - b^2}{a^2} \tag{3.14}$$

In this transformation step, ε is the first numerical eccentricity. It is important to consider that the semi-axis lengths a and b of the *Bessel* ellipsoid have to be used.

4. Transformation from ellipsoid coordinates φ_B, λ_B and h_B into metric *Gauß-Krüger* coordinates x and y.

$$B(\varphi_B) = \alpha\left[\varphi_B + \beta\sin 2\varphi_B + \gamma\sin 4\varphi_B + \delta\sin 6\varphi_B\right] \tag{3.15}$$

$$\eta^2 = \varepsilon'^2\cos^2\varphi_B \tag{3.16}$$

$$N = \frac{a^2}{b\sqrt{1+\eta^2}} \tag{3.17}$$

$$t = \tan\varphi_B \tag{3.18}$$

$$l = \lambda_B - \lambda_0, \tag{3.19}$$

Using these relations, in which $B(\varphi_B)$ is the meridian arc length from the equator to the geographical width φ, N is the radius of curvature and λ_0 is the geographical length of the main meridian, the *Gauß-Krüger* coordinates x (easting) and y (northing) can be calculated with eq. (3.20) and (3.21).

$$\begin{aligned} x =& B(\varphi_B) + \frac{t}{2}N\cos^2\varphi_B l^2 \\ &+ \frac{t}{24}N\cos^4\varphi_B(5 - t^2 + 9\eta^2 + 4\eta^4)l^4 \\ &+ \frac{t}{720}N\cos^6\varphi_B(61 - 58t^2 + t^4 + 270\eta^2 - 330t^2\eta^2)l^6 \\ &+ \frac{t}{40320}N\cos^8\varphi_B(1385 - 3111t^2 + 543t^4 - t^6)l^8 \end{aligned} \tag{3.20}$$

$$\begin{aligned} y =& N\cos\varphi_B l + \frac{1}{6}N\cos^3\varphi_B(1 - t^2 + \eta^2)l^3 \\ &+ \frac{1}{120}N\cos^5\varphi_B(5 - 18t^2 + t^4 + 14\eta^2 - 58t^2\eta^2)l^5 \\ &+ \frac{1}{5040}N\cos^7\varphi_B(61 - 479t^2 + 179t^4 - t^6)l^7 \end{aligned} \tag{3.21}$$

In table 3.1 the values for the parameters used in above equations are listed.

Table 3.1: Transformation parameters

Parameter	Value	Unit
a_{WGS}	6378137.00000	m
b_{WGS}	6356752.31425	m
a_B	6377397.15508	m
b_B	6356078.96290	m
c_x	-577.326	m
c_y	-90.129	m
c_z	-463.919	m
r_x	5.137	arcsec
r_y	1.474	arcsec
r_z	5.297	arcsec
m	-2.423	ppm

3.1.3 Calculation of Lane Width

The next step of the modelling process was to calculate the width of the lanes on the test track by using the *Gauß-Krüger* coordinates from the measurement points determined in the previous step. The lane width results from the calculation of the distance d between point P and intersection point C of line between two points A and B and the perpendicular through P. In fig. 3.5 the calculation principle is illustrated. The general

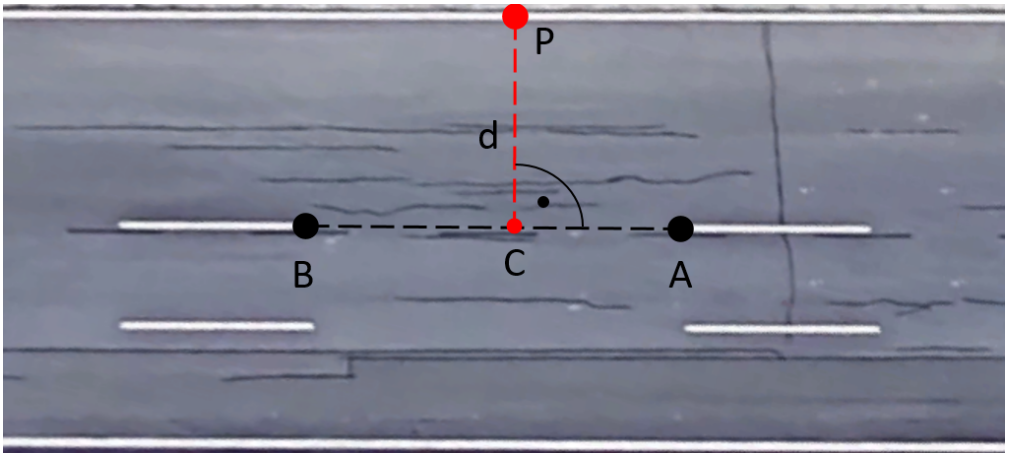


Figure 3.5: Calculation of lane width

form of the equation of a straight line is expressed with eq. (3.22), in which $\mathbf{a}$ is the position vector of point A on the line, t is the running parameter and $\mathbf{u}$ is the direction vector that runs along the line.

$$\mathbf{x}(t) = \mathbf{a} + t\mathbf{u} \tag{3.22}$$

If two points, A and B, on the line are given, the direction vector $\mathbf{u}$ is calculated by subtracting the corresponding position vectors $\mathbf{a}$ and $\mathbf{b}$ from each other.

$$\mathbf{u} = \mathbf{b} - \mathbf{a} \tag{3.23}$$

To find the point of intersection C on $\mathbf{x}(t)$ the principle of orthogonal projection is applied. The orthogonal projection of position vector $\mathbf{b}$ of point P on line $\mathbf{x}(t)$ needs to fulfil two requirements:

- Point C has to be located on line $\mathbf{x}(t)$

$$\mathbf{c}(t_c) = \mathbf{a} + t_C\mathbf{u} \tag{3.24}$$

- Line between the points P and C has to be orthogonal to line $\mathbf{x}(t)$

$$(\mathbf{c} - \mathbf{p}) \cdot \mathbf{u} = 0 \tag{3.25}$$

Plugging in eq. (3.24) into eq. (3.25) and solving for t_0, which is the value of the running parameter at the point of intersection, leads to

$$t_c = \frac{\mathbf{u} \cdot (\mathbf{p} - \mathbf{a})}{\mathbf{u} \cdot \mathbf{u}}. \tag{3.26}$$

The orthogonal projection of a point $\mathbf{p}$ on a straight line $\mathbf{x}(t)$, which is also the point of intersection, is therefore given with

$$\mathbf{c}(t_c) = \mathbf{a} + \frac{\mathbf{u} \cdot (\mathbf{p} - \mathbf{a})}{\mathbf{u} \cdot \mathbf{u}}\mathbf{u}. \tag{3.27}$$

The distance d between the points P and C is then calculated through the following equation.

$$d = |\mathbf{p} - (\mathbf{a} + t_c\mathbf{u})| \tag{3.28}$$

The described mathematical derivations and equations have been implemented into the commercial software MathWorks® MATLAB to determine lane widths of the test track.

3.1.4 Modelling of the Test Track with Scenario Editor

The commercial simulation software IPG CarMaker® offers a graphical user interface, the *Scenario Editor*, that enables the modelling of complex road networks for driving simulation. It is the front end of the road module, which uses the back end module *IPGRoad* as a software library to build roads. Beside designing and modelling road networks, the Scenario Editor is intended for definition of traffic objects and routes for the ego vehicle.

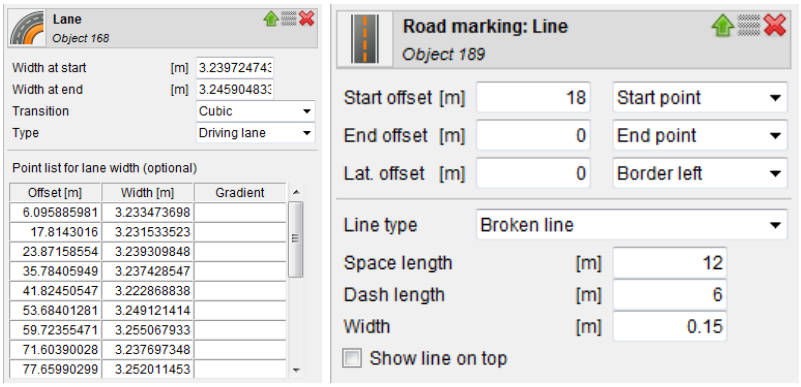
The road network is defined by so called *Links* that are groups of one or more road segments. Scenario Editor provides different segment types such as straight sections,

turns, clothoids, point lists, connects or files. To connect the links, features such as junctions or ramps can be used. Table 3.2 provides an overview of the elements mentioned with a short description. More detailed explanations can be found in [20].

Table 3.2: Overview of segment types [20]

Element	Description
Junction	Connect different Links through the junction arms
Ramp	Clothoid turning to the left. Properties of this element: angle, start and end radius, lateral and longitudinal slope, camber.
Straight	Straight roads
Turn	Define curved roads
Clothoid	Define curved roads with different starting and ending angles
Point list	Define roads through (x,y)-coordinate points
Connect	Automatically connect any two existing segments
File	Use an existing digitized road

With the segment type *File*, digitized road files in different formats can be loaded into the scenario. Among others, IPG CarMaker® supports the import of WGS84 coordinates within a *Keyhole Markup Language* (KML) file, which are automatically translated into cartesian coordinates during generation of the road. This feature was used to define the reference line of the test track based on its WGS84 coordinates φ_{WGS}, λ_{WGS} and h_{WGS}, determined with the GPS-measurement. In this context it is important to mention that the reference line of the roundabout was not modelled with the segment type File, because of occurring problems at the junctions. Instead, the segment type *Turn* with a constant radius was used. The radius was determined by applying the general form of the equation of a circle with three measurement points placed on the circle.



(a) (b)

Figure 3.6: Lane and Road Marking features in Scenario Editor

The next step of the modelling process was the definition of lanes with their corresponding lane width, which has been calculated in the previous step. Within the *Lane* feature, existing lanes can be modified and new lanes can be added. The option *Point list for lane width*, is used to specify the width at specific offset from the beginning of the lane section, see fig. 3.6(a). The calculation of the lane width in the roundabout is based on the same method, used for determining the reference line's radius described before, including an additional calculation of the difference between inner and outer circle.

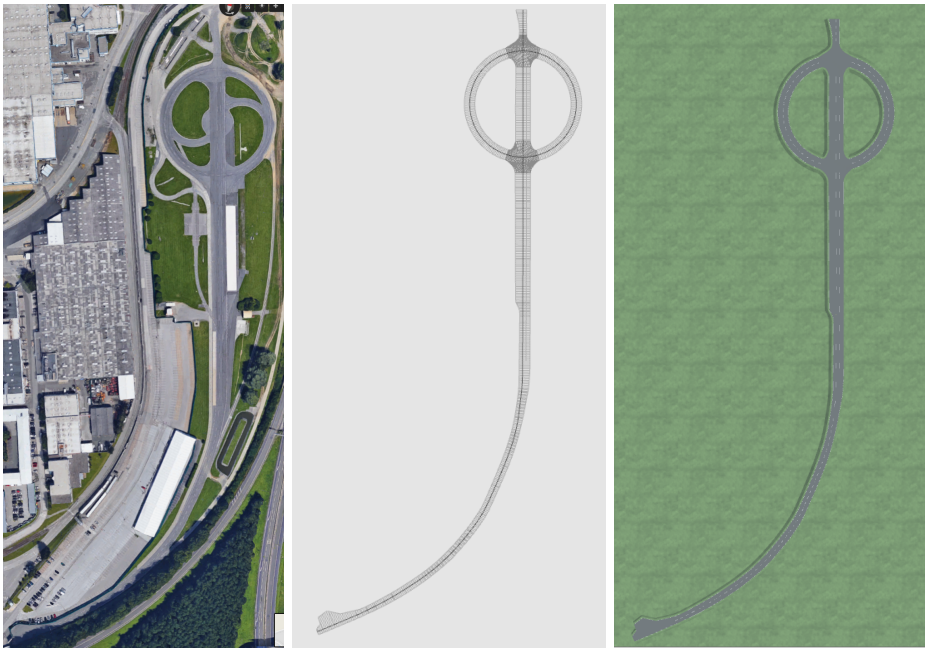
After defining the width off all lanes, the last step was to draw the road markings according to the measured points of the markings of the test track. IPG CarMaker® offers the feature *Road marking* which allows the user to draw lines with different shape and colour. By using the road marking type *Line*, the parameters listed in table 3.3 need to be defined in the GUI, see fig. 3.6(b).

Table 3.3: Parameters for road marking [20]

Name	Unit	Description
Start offset	m	Offset for the starting point of the line with reference to the start/end of a Link
End offset	m	Offset for the end point of the line with reference to the start or end lane section
Lat. offset	m	Lateral offset for the line with reference to any of the lanes or the reference line. Positive values implies left offset section
Line type	-	Can be chosen from - none - single line - broken line - dotted line - double broken line - double dotted line - left-broken double line - right-broken double line - left-dotted double line - right-dotted double time
Space length	m	Length of the gap when the line is of the type dotted or broken
Dash length	m	Length of the line when the line is of the type dotted or broken
Width	m	Width of the line. Line always spreads equally on the left and right side with the Lat. Offset as the reference

The result of the previously described steps of the modelling process is a basic model of the test track at MAGNA STEYR Fahrzeugtechnik AG & CoKG without considering any static objects from the environment, as presented in fig. 3.7. The model can be

used as a basis for further modelling steps with more detailed representation of the environment such as trees and buildings or extension with further parts of the proving ground. Basically it was planned to use this model for the parametrization and validation



(a) (b) (c)

Figure 3.7: Comparison of aerial view and IPG model of the test track

process presented in the following part of the thesis. Since in the end other measurements, which were more suitable for this purpose have been available, this plan was rejected.

3.2 Parametrization and Validation of the High Fidelity Radar Sensor Model

In IPG CarMaker®, beside an ideal object sensor model, a physical *High Fidelity* (HiFi) RADAR sensor model is implemented. It creates a RADAR-specific object list on the basis of priorly defined objects. The detection is based on *Signal to Noise Ratio* (SNR) taking into account

- detection threshold,
- antenna characteristics,
- object specific *RADAR Cross Section* (RCS),
- propagation respectively atmospheric losses and

- object occlusion.

In addition to that, processing effects such as

- measurement noise,
- latency time,
- object merging caused by lack of separability and
- false negatives

are considered in this model. Besides relative position, velocity and acceleration of detected objects, further information such as classification of the object or probability of detection can be provided [19].

Within this thesis, a method for the parametrization and validation of the RADAR sensor model on the basis of real test drives is described. Based on methods for validation of sensor models presented in literature [36], [37] and adapted to the given boundary conditions regarding available models and software tools, the following approach for parametrization and validation of a sensor model has been defined, see fig. 3.8.

1. Definition of a set of n manoeuvres i ($i = 1, ..., n$).
2. Performing real test drives based on the defined manoeuvres and recording real sensor data and reference data.
3. Generation of virtual test scenario for manoeuvre i within the used simulation framework based on recorded data from the measurement system during real test drive.
4. Parametrization of the sensor model according to the data-sheet of the used real sensor hardware.
5. Running of virtual test drive based on the virtual scenario for manoeuvre i generated in step 3 and recording synthetic output data from the sensor model implemented in software.
6. Analysis and comparison of data from real test drive and synthetic data from virtual test drive.
7. Assessment whether the set parameters in step 4 lead to reliable results for all considered output data or not.

In case of finding significant deviation between sensor output data of real test drive and simulation, the parameters of the sensor model need to be varied within step 8, *Parameter variation*, and the process continues again at step 5. This sequence is repeated until results for all reviewed output data are assessed as reliable. Then i is increased by one ($i = i+1$) and the process for the next manoeuvre continues with step 3. The parameters of the sensor model for the oncoming virtual test drive remain the same according to the last state. This can be seen as a validation step for the determined parameters within the parameter variation process before.

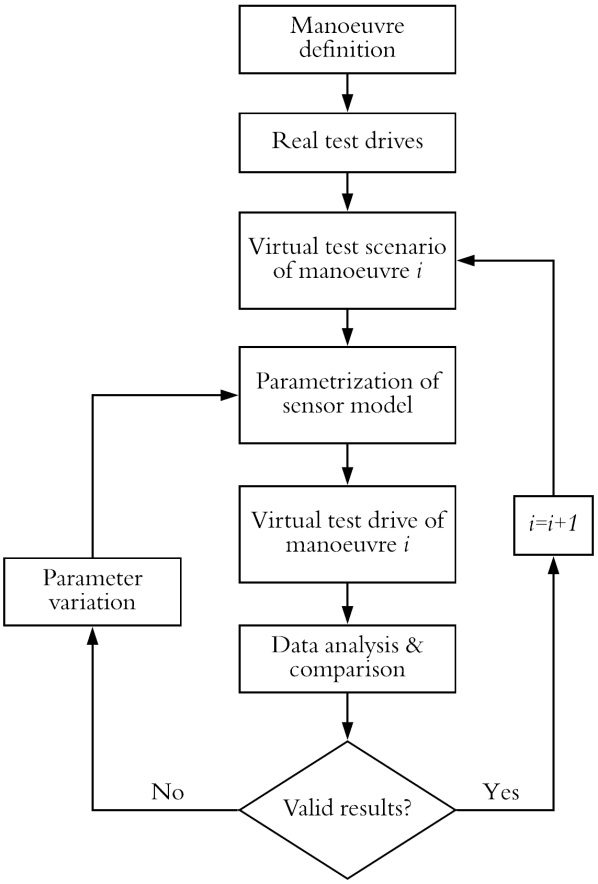


Figure 3.8: Parametrization and validation process

In the following sections the different process-steps of the defined method are described in further detail and explained by means of being applied to the HiFi RADAR sensor model in IPG CarMaker®.

3.2.1 Definition of Manoeuvres

The first step is to define a set of manoeuvres. The main difficulty within this task is to decide which scenarios have to be considered in order to proof whether the sensor model is valid or not. Therefore, it has to be defined which characteristics and outputs of the tested sensor model are relevant for the accurate function of its intended use and have to be validated. In case of a RADAR sensor, the most relevant outputs are informations regarding the distance of detected objects and their relative velocity, their geometrical properties length and width, as well as further informations such as measurement status

or dynamic properties. Mentionable in this context is the importance of the ability of RADAR sensors to separate objects if they are moving next to each other. Within this work a driving manoeuvre with low complexity was used to parametrize and validate the sensor model in a first step. The ego vehicle with activated ACC follows the target vehicle on a straight lane. During the manoeuvre, the set minimal distance between ego and target vehicle is changed. Distance between ego and target vehicle ranges from 2.80 to 27.80 m and velocities up to 40 km/h are reached within this manoeuvre.

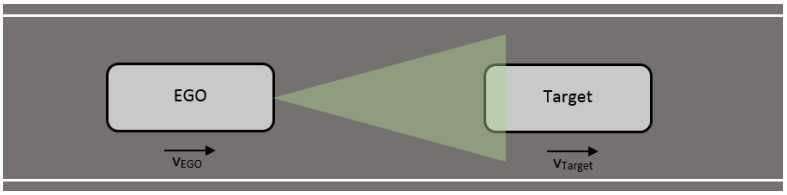


Figure 3.9: Manoeuvre definition

3.2.2 Performing Real Test Drives

Based on the manoeuvre catalogue defined in the previous section, real test drives have to be performed. To record needed data of the test drive, the ego vehicle is equipped with a production RADAR sensor from Continental and a measurement system, which consists of an *Automotive Dynamic Motion Analyser* (ADMA) from GeneSys and the *Data Acquisition Unit* (DAU) from DEWETRON. Additionally, a video camera is mounted behind the windscreen of the ego vehicle, whereby the recorded video data is only used for better traceability what happened during the recorded manoeuvre. The same equipment, excluding RADAR sensor and video camera, is placed in the target vehicle. The measurement systems of the two vehicles are connected by *Wireless Local Area Network* (WLAN). In fig. 3.10 the schematic measurement set-up is illustrated.

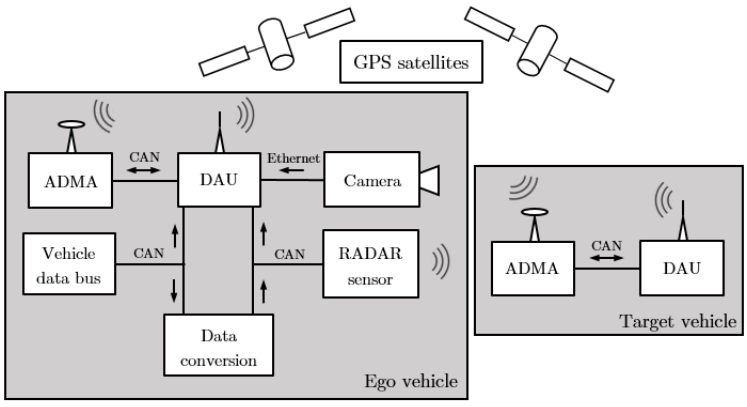


Figure 3.10: Measurement setup, adapted from [5]

The ADMA is used to measure relative distance, velocity and acceleration between ego and target vehicle. Basis of the ADMA are three gyroscopes, which record rotational motion in space. Furthermore, it includes three acceleration sensors for recording linear motions and a *Global Positioning System* (GPS) receiver for determination of absolute position, using the *Real Time Kinematic* (RTK) method for position correction. Therefore, the rough position data is sent via *Global System for Mobile Communications* (GSM) to the base of a service that has a net of reference stations at known positions. Based on observation data of at least three reference stations, the service generates a virtual reference station near the GPS system. This virtual station generates virtual position data and sends it to the GPS device via GSM, which is then able to calculate the position with high accuracy [5]. This method is applied for the ego and target vehicle. The relative distance between ego and target vehicle is then calculated within the DAU of the ego vehicle, which receives the target's position data via WLAN.

The used RADAR sensor of type ARS308 can be operated in the target or object mode, which provides different output signals [17]. Within the parametrization and validation process of this work, data output from the sensor in object mode is used as summarized in table 3.4.

Table 3.4: Output quantities ARS308 [17]

Name	Unit	Description
NoOfObjects	-	Number of measured objects
Obj_LongDispl	m	Longitudinal displacement
Obj_VrelLong	m/s	Relative longitudinal speed
Obj_AccelLong	m/s^2	Relative longitudinal acceleration
Obj_ProbOfExist	-	Probability of existence 0: invalid 1: <25% 2: <50% 3: <75% 4: <90% 5: <99% 6: <99.9% 3: <99.99%
Obj_DynProp	-	Dynamic property 0: unclassified 1: standing (has never been moving before) 2: stopped (has been moving before) 3: moving 4: oncoming
Table continues on next page		

Name	Unit	Description
Obj_Length	-	Object length 0:unknown 1: <0.5m 2: <1m 3: <2m 4: <3m 5: <4m 6: <6m 7: exceeds
Obj_Width	-	Object width 0:unknown 1: <0.5m 2: <1m 3: <2m 4: <3m 5: <4m 6: <6m 7: exceeds
Obj_MeasStat	-	Object measurement status 0: no object 1: new object 2: object not measured 3: object measured
Obj_RCSValue	dBm^2	Radar cross section (RCS)
Obj_ObstacleProbability	%	Probability that the object is an obstacle

3.2.3 Generation of Virtual Test Scenarios

In this step the performed real test drives are reproduced in the virtual simulation environment based on recorded reference data. In IPG CarMaker® within the so called *TestRun* the modules vehicle, driver, road, manoeuvre, traffic and environment have to be parametrized.

3.2.3.1 Vehicle

Within the *Vehicle* module the parameters for the vehicle model regarding body, suspension, steering, tyres, brake, powertrain, aerodynamics and also the sensors are defined, see fig. 3.11. Parametrization and generation of a valid vehicle model, according to the one used in real test drives, is connected with high effort and would reach beyond the scope of this work. For this approach a predefined vehicle model is used and extended with a RADAR sensor, which is parametrized within this module.

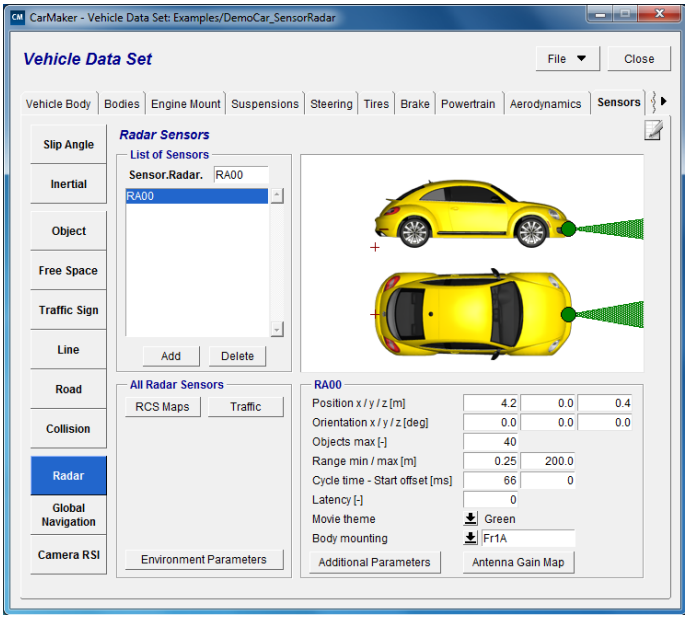


Figure 3.11: Vehicle Data Set

3.2.3.2 Driver

In this module parameters for the driver model implemented in IPG CarMaker® such as maximum acceleration and maximum deceleration are set. The model consists of a controller to follow a course and a speed controller. As well as for the vehicle model, within this work minor effort in parametrizing the driver model was expended and the default settings are used since these two models get passed over by the implemented c-code extension as described in the manoeuvre module.

3.2.3.3 Road

As already described in chapter 3.1.4, IPG CarMaker® offers the tool *Scenario Editor* for definition of roads, which supports the import of WGS84 coordinates from real test drives. For the example manoeuvre described before, see fig. 3.9, a simple road with one segment of type *Straight* without any static objects such as trees was defined.

3.2.3.4 Manoeuvre

Within this module the driving manoeuvres of the ego vehicle during the test run in longitudinal and lateral direction are defined, which are then executed by a driver model. Alternatively, IPG CarMaker® provides the *Input From File* (IFF) function to use real measurements for manoeuvre definition. Using the IFF function has the limitation that

only a target speed for each time step can be defined. The driver model tries to reach this speed as exact as possible. Since no valid driver model is available, this method leads to significant deviations between velocity profile of real and virtual test drive and in further consequence to differences between the covered distance by time and is therefore not suitable for this application.

To work around this problem a c-code extension within the *User.c* file and the following compilation of a new executable was made with the goal to pass over the vehicle and driver model implemented in IPG CarMaker®. As a result, the ego vehicle is pushed along on a custom trajectory defined by its velocity profile, measured during the real test drive, which is exported from the data acquisition and processing software DEWESoft X2 as a text (*.txt*) file and can then be used as input for the IFF function.

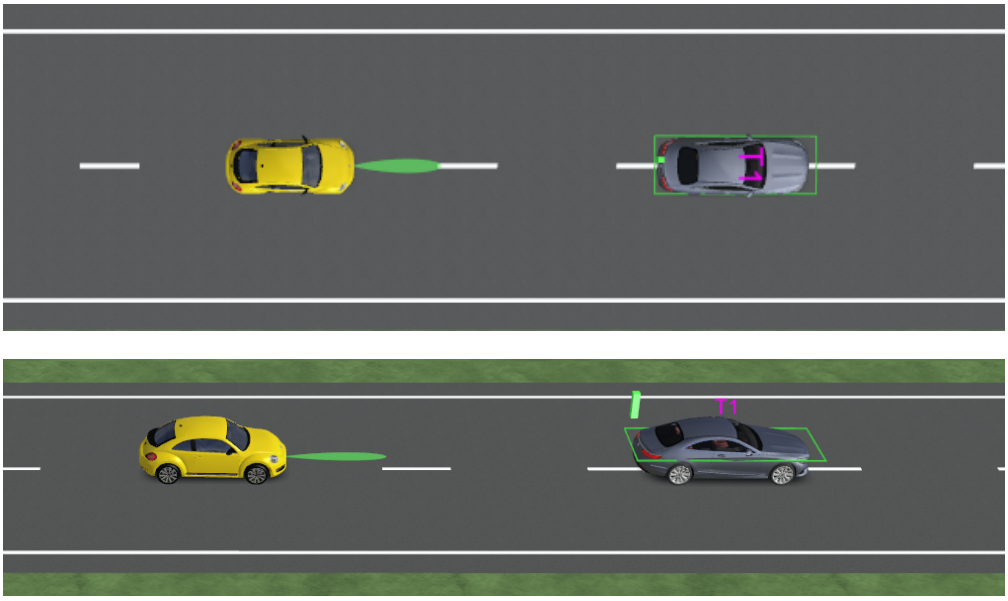


Figure 3.12: Virtual test scenario

3.2.3.5 Traffic

In the *Traffic* module general parameters and the manoeuvre of the target vehicle are defined. In this case, dimensions of the target's bounding box are set, a predefined RCS map is selected, since no RCS map from the real target vehicle is available, and the starting position of the target vehicle relative to the ego vehicle, based on recorded reference data, is defined. The manoeuvre of the target vehicle is defined similarly to that of the ego vehicle by using the velocity profile of the real test as input for the IFF function with the difference that a c-code extension is not necessary.

3.2.3.6 Environment

The *Environment module* enables the user to define environmental conditions such as temperature, rain rate or wind velocity for the virtual test drive. Models take into account different environment parameters, which have an impact on the simulation results. The environment parameters considered by the RADAR sensor model are described in the following section.

3.2.4 Parametrization of the Sensor Model

The RADAR sensor model implemented in IPG offers users the possibility to adjust a variety of parameters with the goal to parametrize the sensor model according to specific parameters of real world sensors, environmental conditions and reflection properties of traffic objects. The following section gives an overview about the parameters available for the RADAR sensor model, which can be found in the tab *Sensors* within the *Vehicle Data Set*, see fig. 3.11. More detailed descriptions of the listed parameters can be found in [19] and [20].

Seed value:
Influences the stochastic part of the model and causes different results for each test run. For comparable results the seed value has to be set to a positive integer.

RCS maps:
Within this dialogue different RCS maps can be defined, which are then assigned to traffic objects in the *Traffic* GUI, see also fig. 2.5(a).

Environment parameters:
The RADAR sensor model considers three parameters which are defined in the *Environment* module.

- **Reference temperature** [**°C**]: Specifies the environmental air temperature at mean sea level.
- **Rain rate** [**mm/h**]: Specifies the amount of rain per hour. Influences the calculation of atmospheric damping.
- **Visual range in fog** [**m**]: Specifies the visibility in a foggy environment. Influences the calculation of atmospheric damping.

Sensor specific parameters:

- **Position (x,y,z)** [**m**]: Position of the sensor frame with respect to vehicle frame (FrD).
- **Orientation (x,y,z)** [**deg**]: Orientation of sensor frame with respect to vehicle frame (FrD).
- **Objects max** [**-**]: Defines the max. size of the object list (the max. number of detectable objects).

- **Range min/max [m]**: Specifies the minimum and maximum radius of the observation area, see fig. 3.13.

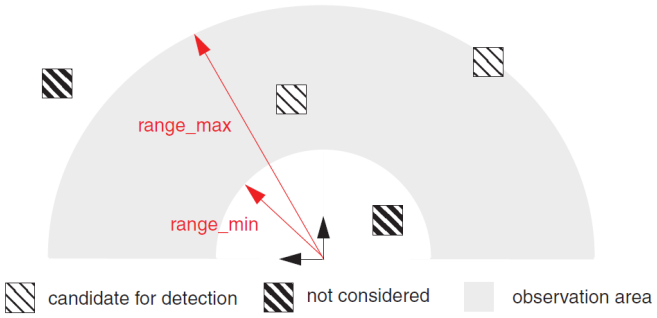


Figure 3.13: Observation area [19]

- **Cycle time [ms]**: Specifies the period of detecting the environment.
- **Start offset [ms]**: Delays the calculation once at the beginning of the simulation.
- **Latency [-]**: Factor (0..1), that is multiplied with the cycle time, resulting in a time offset for updating the output quantities.
- **Movie theme [-]**: Defines the colour of the sensor beam for visualization purposes in IPGMovie.
- **Body Mounting [-]**: Indicates the reference frame on which the sensor is mounted.

Additional parameters:

- **Distance accuracy [m]**: Specifies the standard deviation for the Gaussian distribution curve that is used for distance calculation.
- **Distance resolution [m]**: Defines the minimum radial distance of two objects to be identified as individual ones.
- **Azimuth accuracy [deg]**: Specifies the standard deviation for the Gaussian distribution curve that is used for azimuth angle calculation.
- **Azimuth resolution [deg]**: Defines the minimum angular deviation of two objects to be identified as individual ones.
- **Speed accuracy [km/h]**: Specifies the standard deviation for the Gaussian distribution curve that is used for velocity calculation.
- **Speed resolution [km/h]**: Defines the minimum speed difference of two objects to be identified as individual ones.
- **Separability [-]**: Factor, which adds additional cell units to the resolution.
- **Transmit frequency [GHz]**: Specifies the sensor's operating frequency.

- **Transmit power [dBm]**: The higher the range, the higher the transmitting power should be.
- **System losses [dB]**: Includes all power and system losses and significantly influences the signal strength.
- **Noise bandwidth [Hz]**: Noise bandwidth of the receiver.
- **Noise figure [dB]**: Degradation of the SNR within the signal chain, for a given bandwidth.
- **Probability of detection min [-]**: Defines threshold for probability of detection.
- **Probability of false alarm [-]**: Input for error function to calculate probability of detection.

Antenna gain map:
The antenna gain map is used to consider the characteristic of the antenna. The characteristic is usually split into one main lobe and several side lobes, see fig. 3.14. IPG CarMaker® supports a fixed rectangular aperture antenna model without beam steering.

- **Field of view azimuth / elevation [deg]**: Describe the width of the main lobe in horizontal (azimuth) and vertical (elevation) direction.
- **Antenna efficiency [-]**: Scales the antenna gain map during generation.

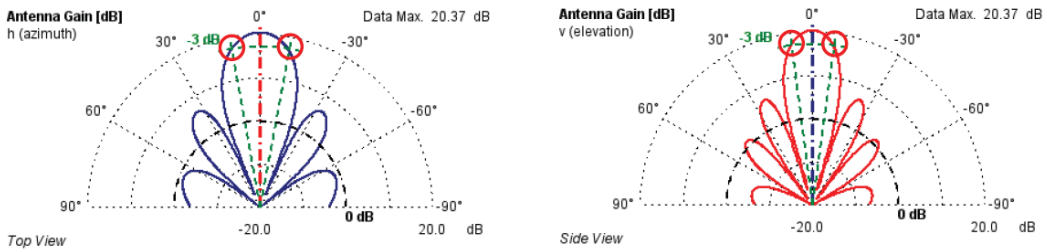


Figure 3.14: Horizontal and vertical section of a antenna gain map [20]

Within the first simulation of the TestRun, parameters of the RADAR sensor model are set according to the data-sheet of the sensor hardware [16], which was used during the real test drive. In case a parameter is not defined in the data-sheet, the predefined default value of the sensor model is used.

Within the parametrization and validation process, illustrated in fig. 3.8, during the step *Parameter variation*, one parameter after another gets varied while all other parameters remain set according to the data-sheet. In table 3.5 the values for parameters according to the data-sheet and the variations are listed.

Table 3.5: Variation of model parameters

Parameter [Unit]	Datasheet/Default	V1	V2	V3	V4	V5	V6
Reference temperature [°C]	20	-10	0	10	30	40	100
Rain rate [mm/h]	0	5	10	20	40	80	100
Visual range in fog [m]	1000	0	10	100	500	1500	2000
Distance accuracy [m]	0.25	0	0.01	0.1	0.5	1	10
Distance resolution [m]	2	0	0.1	1	2.5	10	100
Azimuth accuracy [°]	0.1	0	0.01	0.05	0.5	1	10
Azimuth resolution [°]	1	0	0.1	0.5	1.5	10	100
Speed accuracy [km/h]	0.5	0	0.01	0.025	0.75	1	10
Speed resolution [km/h]	2.76	0	0.1	1.5	3.5	10	100
Separability [-]	1.5	0	0.1	1	3	10	100
Transmit frequency [GHz]	77	0	0.1	1	10	100	1000
Transmit power [dBm]	10	0	0.1	1	9	12	100
System losses [dB]	0	1	5	10	20	50	100
Noise bandwidth [Hz]	25000	30000	35000	40000	45000	50000	100000
Noise figure [dB]	4.8	10	20	40	60	80	100
Probability of detection min [-]	0.5	0	0.1	0.9	1	2	10
Probability of false alarm [-]	10^{-6}	10^{-1}	10^{-3}	10^{-5}	10^{-7}	10^{-9}	10^{-10}
Field ov view azimuth [°]	17	0.1	1	10	30	60	180
Field ov view elevation [°]	4.3	0.1	2	10	30	60	180
Antenna efficiency [-]	1	0.1	0.3	0.5	0.8	1.5	2

3.2.5 Virtual Test Drive

Within this step virtual test drives are executed in the simulation framework, based on the virtual test scenarios (TestRun) generated before. To reduce the effort of executing virtual test drives for all of the parameter variations, the *Test Manager* is used. The Test Manager is a tool implemented in IPG Carmaker® and offers the possibility to use variations of parameters within a TestRun instead of creating a separate TestRun for each variation. Therefore, the RADAR sensor parameters have to be set as so called *Named Values* (NValue). This is realised by entering a dollar sign ($) to the parameter field within the vehicle module, followed by a variable name and a default value, in this case the value according to the data-sheet. In fig. 3.15 the GUI of the Test Manager for variation of the speed accuracy is illustrated as an example.

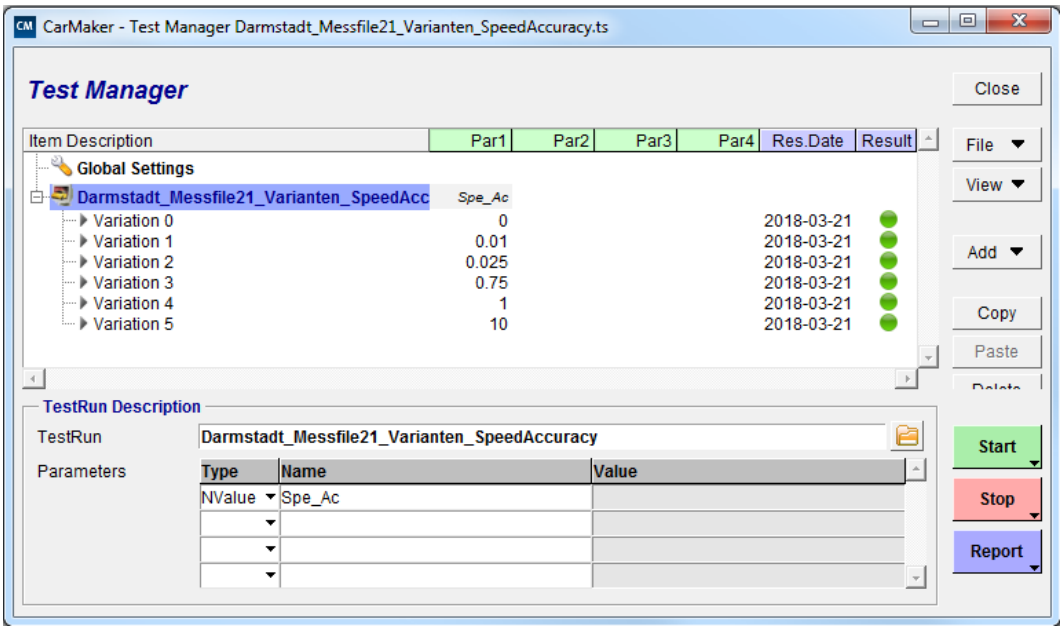


Figure 3.15: IPG CarMaker® Test Manager

During the simulation of a TestRun data from the RADAR sensor model is recorded and stored into a result file, which is than used for further considerations. The RADAR sensor model provides general as well as object specific output signals. In table 3.6 an overview of the output quantities is given, based on [19], [20] wherein more detailed descriptions of the signals can be found. In this table lines with the output signals used for data comparison in the next step are highlighted in light grey.

Table 3.6: Output quantities of the RADAR sensor model

Name	Unit	Description (data type, if not double)
General		
nObj	-	number of detected objects (integer)
RelvTgt	-	object id of the relevant target (integer)
nLanesL	-	number of lanes left
nLanesR	-	number of lanes right
DistToLeftBorder	m	distance to left road border
DistToRightBorder	m	distance to right road border
Object list		
ObjId	-	global ID to identify object
LeftInGroup	-	object id of left traffic object in group, if sensor sees two or more traffic objects as one (integer)
MeasStat	-	measurement status (integer) 0: no object 1: new object 2: (currently not used) 3: object measured
Dist	m	distance from sensor to reference point of the detected object
DistX	m	longitudinal displacement in sensor frame
DistY	m	lateral displacement in sensor frame
DistZ	m	vertical displacement in sensor frame
RelCourseAngle	rad	relative course angle in sensor frame
Vrel	m/s	velocity of reference point of the detected object relative to sensor
VrelX	m/s	relative longitudinal velocity
VrelY	m/s	relative lateral velocity
ArelX	m/s^2	relative longitudinal acceleration
Length	m	length of bounding box
Width	m	width of bounding box

Table continues on next page

Name	Unit	Description (data type, if not double)
LengthClass	-	length class from 0 to 7 (integer) 0:unknown 1: <0.5m 2: <1m 3: <2m 4: <3m 5: <4m 6: <6m 7: exceeds
WidthClass	-	width class from 0 to 7 (integer) 0:unknown 1: <0.5m 2: <1m 3: <2m 4: <3m 5: <4m 6: <6m 7: exceeds
RCS	dBm^2	radar cross section - value
SignalStrength	dBW	signal
SNR	dB	signal-to-noise ratio
DynProp	-	dynamic property (integer) 0: not classified 1: standing (has never moved before) 2: stopped 3: moving 4: oncoming
ProbDetect	-	probability of detection
ProbExist	-	probability of existence (integer)
ProbObst	-	obstacle probability

Important within this context is the fact that the reference point of the traffic objects is always the centre of the rectangle visualizing the object consisting of length and width.

3.2.6 Comparison of Real Test and Simulation

Within this step RADAR sensor output data from real test drive and virtual TestRun are compared with each other, to asses if the RADAR sensor model provides reliable results on the one hand, and on the other hand to evaluate to which extend the different parameters influence the output .

As criteria for the assessment, maximum and average deviation of longitudinal displace-

ment $\Delta\text{Dist}_x\text{max}$ and $\Delta\text{Dist}_x\text{ave}$, maximum and average deviation of relative longitudinal velocity $\Delta\text{Vrel}_x\text{max}$ and $\Delta\text{Vrel}_x\text{ave}$, and the accurate detection of the object within a TestRun have been defined. To evaluate the correct object detection over the simulation cycle, the object properties measurement status, length class, width class, dynamic property, probability of existence and obstacle probability are used, see table 3.6.

The calculation of $\Delta\text{Dist}_x\text{max}$, $\Delta\text{Dist}_x\text{ave}$, $\Delta\text{Vrel}_x\text{max}$ and $\Delta\text{Vrel}_x\text{ave}$ as well as graphical comparison of output quantities from real sensor and RADAR sensor model for qualitative assessment is implemented in MATLAB®. Since the real RADAR sensor detects many different objects during real test drive, recorded data has to be filtered to extract only the relevant data of the target vehicle. For this purpose a researcher from the institute of automotive engineering developed a tool in MATLAB®, see fig. 3.16, to visualise reflection points during the measurement, with the aim to match the relevant data to the target vehicle which is then stored in a MATLAB® (*.mat*) file.

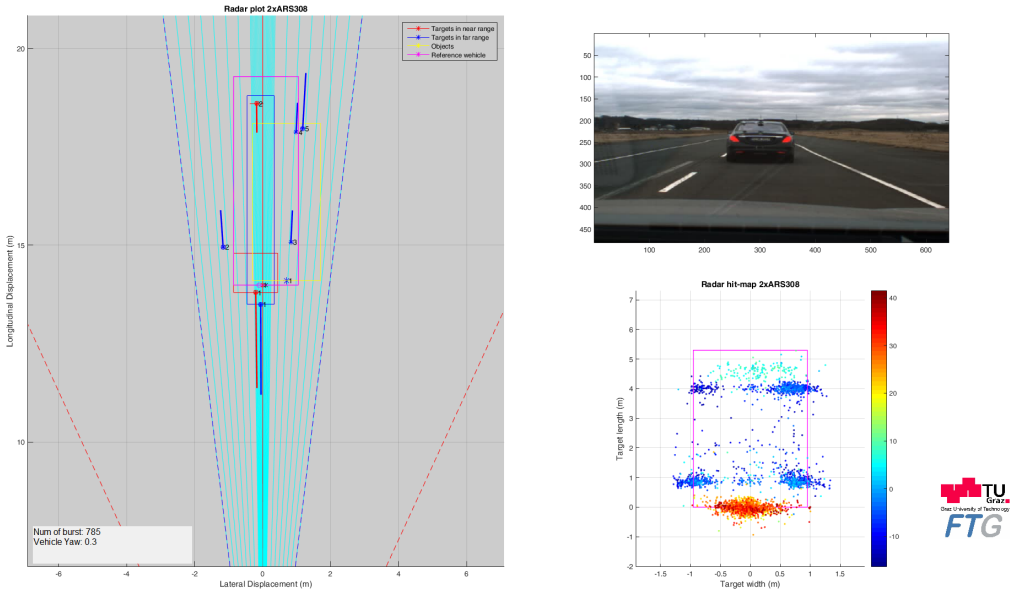


Figure 3.16: Visualisation tool

The output data from the virtual TestRun is imported with the *cm read* function, which allows the import of simulation results from IPG CarMaker® into MATLAB®. To compare data from real test drive and simulation using plots, the data arrays have to be of same size. Therefore, within the MATLAB® script data of the simulation is reduced according to the size of data arrays from the real test drive. Then $\Delta\text{Dist}_x\text{max}$, $\Delta\text{Dist}_x\text{ave}$, $\Delta\text{Vrel}_x\text{max}$ and $\Delta\text{Vrel}_x\text{ave}$ is calculated and graphical comparison of the output quantities is visualised. As an example, in fig. 3.17 measured and simulated longitudinal displacement and reference distance from a random simulation variation is plotted over time.

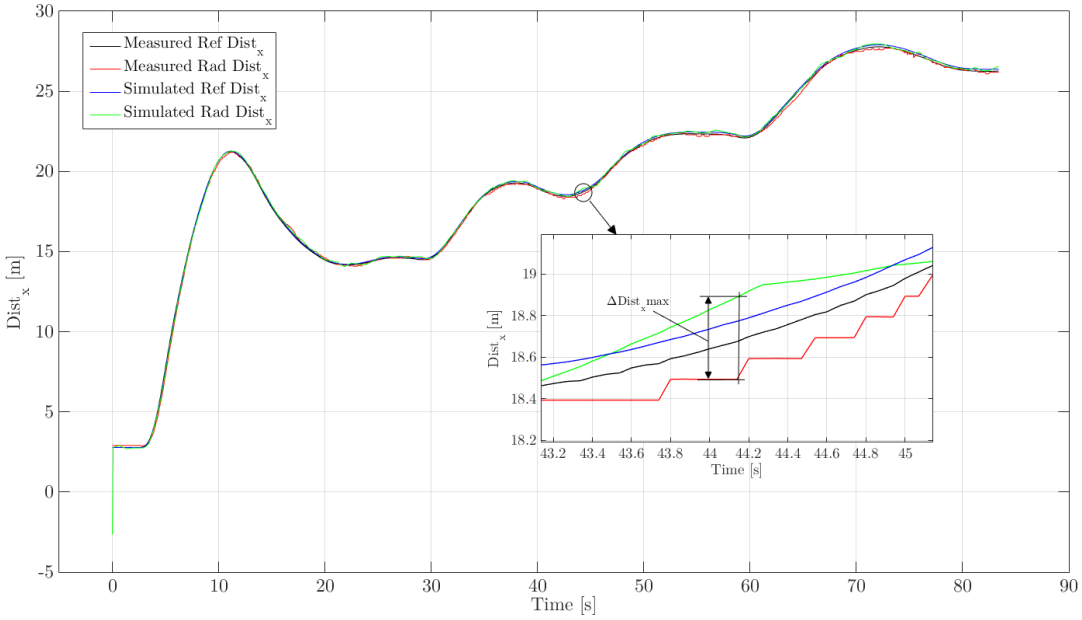


Figure 3.17: Comparison of measured and simulated longitudinal displacement

3.2.7 Results

In the following paragraphs, results for comparison between real test drive and simulation with different parameter variations are listed in tables 3.7-3.27 and discussed in further detail.

According to data-sheet:
In table 3.7 results for parametrizing the sensor model according to the data-sheet of the real sensor hardware are shown, which are used as reference for further evaluation. On consideration of the assessment criteria, the sensor model delivers quite satisfying results for the used manoeuvre.

Table 3.7: Parameters according to data-sheet

$\Delta Dist_x max$ [m]	$\Delta Dist_x ave$ [m]	$\Delta Vrel_x max$ [m/s]	$\Delta Vrel_x ave$ [m/s]	Object detection
0.40	0.13	0.30	0.05	$\surd$

Reference temperature:
Results in table 3.8 show that variation of the reference temperature does not influence outputs of simulated distance and relative velocity, and the target vehicle is detected through the whole duration of the simulation.

Table 3.8: Variation of reference temperature

Reference temperature [°C]	ΔDist$_x$max [m]	ΔDist$_x$ave [m]	ΔVrel$_x$max [m/s]	ΔVrel$_x$ave [m/s]	Object detection
-10	0.40	0.13	0.30	0.05	√
0	0.40	0.13	0.30	0.05	√
10	0.40	0.13	0.30	0.05	√
30	0.40	0.13	0.30	0.05	√
40	0.40	0.13	0.30	0.05	√
100	0.40	0.13	0.30	0.05	√

Rain rate:
Variation of rain rate also has no influence on results within this manoeuvre, as seen in table 3.9. To evaluate the impact of this parameter, considerations of further manoeuvres are necessary.

Table 3.9: Variation of rain rate

Rain rate [mm/h]	ΔDist$_x$max [m]	ΔDist$_x$ave [m]	ΔVrel$_x$max [m/s]	ΔVrel$_x$ave [m/s]	Object detection
5	0.40	0.13	0.30	0.05	√
10	0.40	0.13	0.30	0.05	√
20	0.40	0.13	0.30	0.05	√
40	0.40	0.13	0.30	0.05	√
80	0.40	0.13	0.30	0.05	√
100	0.40	0.13	0.30	0.05	√

Range in fog:
Apart from setting the range in fog to 0 m, variation of this parameter has no impact on the results within the tested manoeuvre, see table 3.10. To asses the impact of this parameter a manoeuvre with higher distance between ego and target vehicle has to be considered.

Table 3.10: Variation of range in fog

Range in fog [m]	ΔDist$_x$max [m]	ΔDist$_x$ave [m]	ΔVrel$_x$max [m/s]	ΔVrel$_x$ave [m/s]	Object detection
0	30.33	22.12	4.06	0.52	×
10	0.40	0.13	0.30	0.05	√
100	0.40	0.13	0.30	0.05	√
500	0.40	0.13	0.30	0.05	√
1500	0.40	0.13	0.30	0.05	√
2000	0.40	0.13	0.30	0.05	√

Distance accuracy:
The distance accuracy is the only parameter which directly influences distance calculation. Results in table 3.11 show that significant deviations only occur at parameter values which are not reasonable and therefore not relevant. Within the range of distance accuracy between 0 and 0.5 m, $\Delta\text{Dist}_x\text{max}$ only differs by 0.14 m.

Table 3.11: Variation of distance accuracy

Distance accuracy [m]	$\Delta\text{Dist}_x\text{max}$ [m]	$\Delta\text{Dist}_x\text{ave}$ [m]	$\Delta\text{Vrel}_x\text{max}$ [m/s]	$\Delta\text{Vrel}_x\text{ave}$ [m/s]	Object detection
0	0.38	0.12	0.30	0.05	√
0.01	0.38	0.12	0.30	0.05	√
0.1	0.38	0.12	0.30	0.05	√
0.5	0.52	0.15	0.30	0.05	√
1	0.88	0.24	0.30	0.05	√
10	7.87	2.15	0.30	0.05	√

Distance resolution:
The parameter distance resolution specifies the minimum radial distance between two objects to be detected as individual ones. Therefore, it is reasonable that variation of this parameter does not influence results within the considered manoeuvre, as shown in table 3.12. To evaluate the impact of this parameter on simulation output, a manoeuvre with at least two target vehicles is needed.

Table 3.12: Variation of Distance resolution

Distance resolution [m]	$\Delta\text{Dist}_x\text{max}$ [m]	$\Delta\text{Dist}_x\text{ave}$ [m]	$\Delta\text{Vrel}_x\text{max}$ [m/s]	$\Delta\text{Vrel}_x\text{ave}$ [m/s]	Object detection
0	0.40	0.13	0.30	0.05	√
0.1	0.40	0.13	0.30	0.05	√
1	0.40	0.13	0.30	0.05	√
2.5	0.40	0.13	0.30	0.05	√
10	0.40	0.13	0.30	0.05	√
100	0.40	0.13	0.30	0.05	√

Azimuth accuracy:
Results in table 3.13 show that variation of azimuth accuracy also has no impact on the output of the sensor model. To evaluate the parameter, a manoeuvre with lateral offset between ego and target vehicle needs to be considered and an additional criteria for assessment of output quantities in lateral direction has to be introduced.

Table 3.13: Variation of Azimuth accuracy

Azimuth accuracy [°]	ΔDist$_x$max [m]	ΔDist$_x$ave [m]	ΔVrel$_x$max [m/s]	ΔVrel$_x$ave [m/s]	Object detection
0	0.40	0.13	0.30	0.05	√
0.01	0.40	0.13	0.30	0.05	√
0.05	0.40	0.13	0.30	0.05	√
0.5	0.40	0.13	0.30	0.05	√
1	0.40	0.13	0.30	0.05	√
10	0.40	0.13	0.30	0.05	√

Azimuth resolution:
The parameter azimuth resolution specifies the minimum angular deviation between two objects to be detected as individual ones. Therefore, it is reasonable that variation of this parameter does not influence results within the considered manoeuvre, as shown in table 3.14. To evaluate the impact of this parameter on simulation output, a manoeuvre with at least two target vehicles is needed.

Table 3.14: Variation of Azimuth resolution

Azimuth resolution [°]	ΔDist$_x$max [m]	ΔDist$_x$ave [m]	ΔVrel$_x$max [m/s]	ΔVrel$_x$ave [m/s]	Object detection
0	0.40	0.13	0.30	0.05	√
0.1	0.40	0.13	0.30	0.05	√
0.5	0.40	0.13	0.30	0.05	√
1.5	0.40	0.13	0.30	0.05	√
10	0.40	0.13	0.30	0.05	√
100	0.40	0.13	0.30	0.05	√

Speed accuracy:
The speed accuracy is the only parameter that directly influences calculation of speed. Results in table 3.15 point out that variation of speed accuracy within the range between 0 and 1 km/h shows no significant impacts on ΔVrel$_x$max and ΔVrel$_x$ave.

Table 3.15: Variation of Speed accuracy

Speed accuracy [km/h]	ΔDist$_x$max [m]	ΔDist$_x$ave [m]	ΔVrel$_x$max [m/s]	ΔVrel$_x$ave [m/s]	Object detection
0	0.40	0.13	0.30	0.05	$\surd$
0.01	0.40	0.13	0.30	0.05	$\surd$
0.025	0.40	0.13	0.30	0.05	$\surd$
0.75	0.40	0.13	0.30	0.06	$\surd$
1	0.40	0.13	0.32	0.07	$\surd$
10	0.40	0.13	2.48	0.55	$\surd$

Speed resolution:
The parameter speed resolution specifies the minimum speed difference between two objects to be detected as individual ones. Therefore, it is reasonable that variation of this parameter does not influence results within the considered manoeuvre, as shown in table 3.16. To evaluate the impact of this parameter on simulation output, a manoeuvre with at least two target vehicles is needed.

Table 3.16: Variation of Speed resolution

Speed resolution [km/h]	ΔDist$_x$max [m]	ΔDist$_x$ave [m]	ΔVrel$_x$max [m/s]	ΔVrel$_x$ave [m/s]	Object detection
0	0.40	0.13	0.30	0.05	$\surd$
0.1	0.40	0.13	0.30	0.05	$\surd$
1.5	0.40	0.13	0.30	0.05	$\surd$
3.5	0.40	0.13	0.30	0.05	$\surd$
10	0.40	0.13	0.30	0.05	$\surd$
100	0.40	0.13	0.30	0.05	$\surd$

Separability:
Same as for distance, azimuth and speed resolution, variation of this parameter does not influence results, see table 3.17, since it only has impact on the ability of the sensor model to differentiate between objects which are located close to each other. For evaluation of this parameter also a manoeuvre with at least two target vehicles is needed.

Table 3.17: Variation of Separability

Separability [-]	ΔDist$_x$max [m]	ΔDist$_x$ave [m]	ΔVrel$_x$max [m/s]	ΔVrel$_x$ave [m/s]	Object detection
0	0.40	0.13	0.30	0.05	$\surd$
0.1	0.40	0.13	0.30	0.05	$\surd$
1	0.40	0.13	0.30	0.05	$\surd$
3	0.40	0.13	0.30	0.05	$\surd$
10	0.40	0.13	0.30	0.05	$\surd$
100	0.40	0.13	0.30	0.05	$\surd$

Transmit frequency:
Table 3.18 shows that variation of the transmit frequency within the range between 0 to 100 GHz has no impact on results of the simulation. Since real RADAR sensors are operated within frequency bands between 24 and 81 GHz, results at higher frequencies are not of great interest.

Table 3.18: Variation of Transmit frequency

Transmit frequency [GHz]	ΔDist$_x$max [m]	ΔDist$_x$ave [m]	ΔVrel$_x$max [m/s]	ΔVrel$_x$ave [m/s]	Object detection
0	0.40	0.13	0.30	0.05	$\surd$
0.1	0.40	0.13	0.30	0.05	$\surd$
1	0.40	0.13	0.30	0.05	$\surd$
10	0.40	0.13	0.30	0.05	$\surd$
100	0.40	0.13	0.30	0.05	$\surd$
1000	30.33	20.45	2.75	0.37	$\times$

Transmit power:
Results in table 3.19 point out that variation of the transmit power does not influence the output quantities of the sensor model within the observed manoeuvre.

Table 3.19: Variation of Transmit power

Transmit power [dBm]	ΔDist$_x$max [m]	ΔDist$_x$ave [m]	ΔVrel$_x$max [m/s]	ΔVrel$_x$ave [m/s]	Object detection
0	0.40	0.13	0.30	0.05	$\surd$
0.1	0.40	0.13	0.30	0.05	$\surd$
1	0.40	0.13	0.30	0.05	$\surd$
9	0.40	0.13	0.30	0.05	$\surd$
12	0.40	0.13	0.30	0.05	$\surd$
100	0.40	0.13	0.30	0.05	$\surd$

System losses:
Results in table 3.20 show that the parameter system losses influences correct detection of the target vehicle. It is difficult to define variations of this parameter since it is not specified in the data-sheet of the used RADAR sensor hardware.

Table 3.20: Variation of System losses

System losses [dB]	ΔDist$_x$max [m]	ΔDist$_x$ave [m]	ΔVrel$_x$max [m/s]	ΔVrel$_x$ave [m/s]	Object detection
1	0.40	0.13	0.30	0.05	$\surd$
5	0.40	0.13	0.30	0.05	$\surd$
10	0.40	0.13	0.30	0.05	$\surd$
20	0.40	0.13	0.30	0.05	$\surd$
50	30.14	0.45	0.56	0.05	$\sim$
100	30.33	22.12	4.06	5.20	$\times$

Noise bandwidth:
Variation of the noise bandwidth within the range between 30000 and 100000 Hz has no impact on simulation results, listed in table 3.21. Same as for system losses, this parameter is not specified in the RADAR sensor data-sheet.

Table 3.21: Variation of Noise bandwidth

Noise bandwidth [Hz]	ΔDist$_x$max [m]	ΔDist$_x$ave [m]	ΔVrel$_x$max [m/s]	ΔVrel$_x$ave [m/s]	Object detection
30000	0.40	0.13	0.30	0.05	$\surd$
35000	0.40	0.13	0.30	0.05	$\surd$
40000	0.40	0.13	0.30	0.05	$\surd$
45000	0.40	0.13	0.30	0.05	$\surd$
50000	0.40	0.13	0.30	0.05	$\surd$
100000	0.40	0.13	0.30	0.05	$\surd$

Noise figure:
Table 3.22 shows that the noise figure does influence the ability of the sensor model to detect the target vehicle. The value of this parameter for the used RADAR sensor is not specified by the producer.

Table 3.22: Variation of Noise figure

Noise figure [dB]	ΔDist$_x$max [m]	ΔDist$_x$ave [m]	ΔVrel$_x$max [m/s]	ΔVrel$_x$ave [m/s]	Object detection
10	0.40	0.13	0.30	0.05	$\surd$
20	0.40	0.13	0.30	0.05	$\surd$
40	0.40	0.13	0.30	0.05	$\surd$
60	30.33	6.14	1.19	0.09	$\sim$
80	30.33	22.12	4.06	0.52	$\times$
100	30.33	22.12	4.06	0.52	$\times$

Probability of detection min:
Results in table 3.23 show that this parameter has to be set within the range between 0 and 1 for accurate object detection of the sensor model. In case of setting it to 1 or higher, the target vehicle is not detected during the complete simulation.

Table 3.23: Variation Probability of detection min

Probability of detection min [-]	ΔDist$_x$max [m]	ΔDist$_x$ave [m]	ΔVrel$_x$max [m/s]	ΔVrel$_x$ave [m/s]	Object detection
0	0.40	0.13	0.30	0.05	$\surd$
0.1	0.40	0.13	0.30	0.05	$\surd$
0.9	0.40	0.13	0.30	0.05	$\surd$
1	30.33	22.12	4.06	0.52	$\times$
2	30.33	22.12	4.06	0.52	$\times$
10	30.33	22.12	4.06	0.52	$\times$

Probability of false alarm:
This parameter shows no effect on the reviewed output data of the sensor model within this manoeuvre, see table 3.24.

Table 3.24: Variation Probability of false alarm

Probability of false alarm [-]	ΔDist$_x$max [m]	ΔDist$_x$ave [m]	ΔVrel$_x$max [m/s]	ΔVrel$_x$ave [m/s]	Object detection
10^{-1}	0.40	0.13	0.30	0.05	$\surd$
10^{-3}	0.40	0.13	0.30	0.05	$\surd$
10^{-5}	0.40	0.13	0.30	0.05	$\surd$
10^{-7}	0.40	0.13	0.30	0.05	$\surd$
10^{-9}	0.40	0.13	0.30	0.05	$\surd$
10^{-10}	0.40	0.13	0.30	0.05	$\surd$

Field of view azimuth, Field of view elevation, Antenna efficiency:
These 3 parameters specify the antenna gain map of the sensor model which is used to consider the characteristic of the antenna. However, results in tables 3.25, 3.26 and 3.27 show that within the considered manoeuvre variation of these parameters does not influence the output of the sensor model.

Table 3.25: Variation Field of view azimuth

Field of view azimuth [°]	ΔDist$_x$max [m]	ΔDist$_x$ave [m]	ΔVrel$_x$max [m/s]	ΔVrel$_x$ave [m/s]	Object detection
0.1	0.40	0.13	0.30	0.05	√
1	0.40	0.13	0.30	0.05	√
10	0.40	0.13	0.30	0.05	√
30	0.40	0.13	0.30	0.05	√
60	0.40	0.13	0.30	0.05	√
180	0.40	0.13	0.30	0.05	√

Table 3.26: Variation Field of view elevation

Field of view elevation [°]	ΔDist$_x$max [m]	ΔDist$_x$ave [m]	ΔVrel$_x$max [m/s]	ΔVrel$_x$ave [m/s]	Object detection
0.1	0.40	0.13	0.30	0.05	√
2	0.40	0.13	0.30	0.05	√
10	0.40	0.13	0.30	0.05	√
30	0.40	0.13	0.30	0.05	√
60	0.40	0.13	0.30	0.05	√
180	0.40	0.13	0.30	0.05	√

Table 3.27: Variation Antenna efficiency

Antenna efficiency [-]	ΔDist$_x$max [m]	ΔDist$_x$ave [m]	ΔVrel$_x$max [m/s]	ΔVrel$_x$ave [m/s]	Object detection
0.1	0.40	0.13	0.30	0.05	√
0.3	0.40	0.13	0.30	0.05	√
0.5	0.40	0.13	0.30	0.05	√
0.8	0.40	0.13	0.30	0.05	√
1.5	0.40	0.13	0.30	0.05	√
2	0.40	0.13	0.30	0.05	√

To summarize previous results out of comparison from real RADAR sensor output and virtual sensor model output, the following statements can be deduced:

- There exist enormous differences of the measured value for the RCS between real test drive and simulation, see fig. 3.18. This results from the fact that only a predefined RCS-map is used within the virtual test drive, since determination of the RCS values and researching the reflection phenomena within RADAR physics is of high complexity. Since the RCS influences the strength of the received signal and therefore the object detection, it is recommended to focus on modelling these phenomena or at least to find a way to consider them in a way that is more linked to reality.

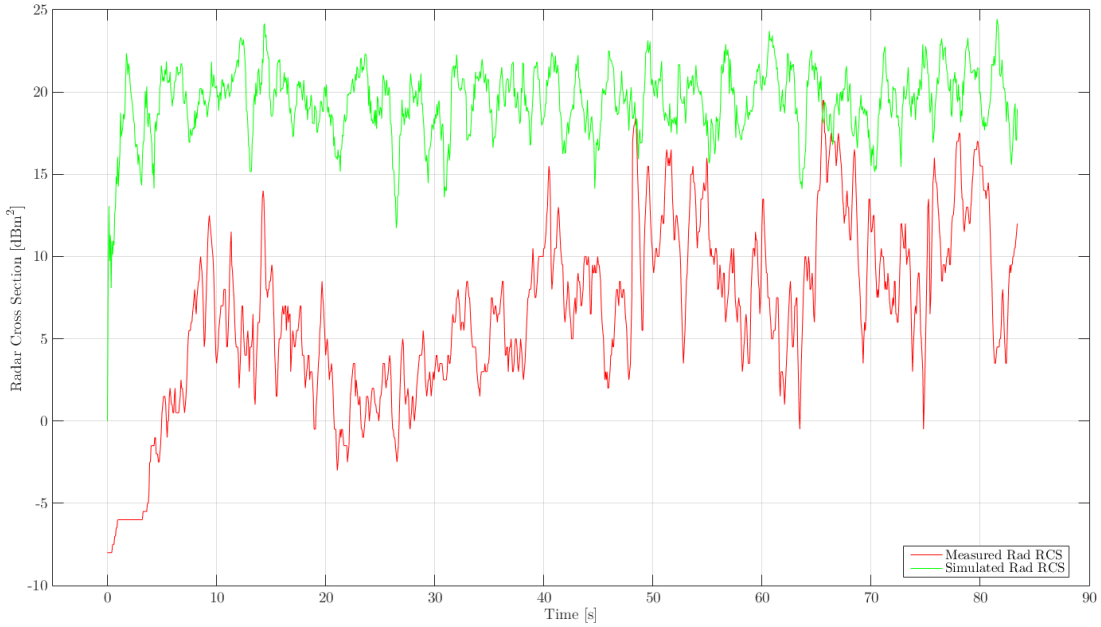


Figure 3.18: Comparison of measured and simulated RCS value

- The RADAR sensor model generates wrong output for the objects *LengthClass*.
- The RADAR sensor model classifies the *Dynamic Property* of the target vehicle mistakenly as *oncoming* in some cases.
- The considered manoeuvre with the ego vehicle following one target vehicle (**Following test**) on a straight lane without lateral offset and small longitudinal displacement is only limited applicable for the evaluation of all parameters.

4 Summary and Outlook

Within this chapter a summary of the outcome of the presented master's thesis is provided and an outlook and recommendations for further steps and improvements are given.

Chapter 1: Introduction
In this chapter the main drivers for the increasing demand of development and integration of *Advanced Driver Assistance Systems* (ADAS) in modern vehicles to improve safety and comfort of passengers were discussed. The set goal from the European Union in 2010, to reduce road fatalities by half until the year 2020, and the fact that the European *New Car Assessment Programme* (NCAP) includes the *Automatic Emergency Brake* (AEB) and *Lane Keeping Assist* (LKA) in their current assessment program has led to the fact that ADAS have become available even in vehicles of the lower cost segment. Therefore, virtual development and validation methods for ADAS and *Automated Driving Systems* (ADS) are needed with the aim to reduce execution of time-consuming, expensive and partly safety critical test drives.

This leads to the prerequisite to use valid environment perception sensor models, which are able to represent the specific characteristics of the real sensor hardware in a sufficient way. The output of these sensor models is the basis for the algorithms of ADAS and ADS functions. Therefore they have to be validated compared to real sensor data before being implemented into a simulation framework.

Chapter 2: State of the Art
In the second chapter the term ADAS was defined and different possible classifications were presented. Due to the growing trend towards ADS an overview of levels of driving automation according to the *Society of Automotive Engineers* (SAE) was given. Furthermore, an overview of used perception sensors for environment recognition with a focus on *Radio Detection and Ranging* (RADAR) sensors, including basic physical principles and methods for distance, velocity and angle measurement was provided. At the end of this chapter existing approaches for RADAR sensor modelling within the virtual development process were described and suggested methods for their validation have been presented.

Chapter 3: Methodology
The first part of this chapter presented the modelling process of the test track at the proving ground from MAGNA STEYR Fahrzeugtechnik AG & CoKG with the goal to build up a model for future virtual testing of camera bases ADAS functions. Various steps of the process, beginning with high accurate GPS measurement of the test track,

over coordinate transformation into a usable coordinate system, calculation of the lane widths to the final modelling of the test track within the software tool IPG CarMaker® were described. The result is a basic model of the test track without considering any static objects of the environment. The model can be used as basis for further modelling steps with more detailed representation of the environment such as trees and buildings or extension with further parts of the proving ground. Basically it was planned to use this model for the parametrization and validation process presented in the thesis. Since in the end other measurements, which were more suitable for this purpose have been available, this plan was rejected.

The second part of this chapter presented a method for the parametrization and validation of a physical *High Fidelity* (HiFi) RADAR sensor model, as implemented in the simulation tool IPG CarMaker®, on the basis of real test drives. The goal was to analyse the impact of available and adjustable parameters within the sensor model on its output. The method was applied by reproducing a simple manoeuvre (**Following test**) of the real test drive in a virtual scenario and comparison of recorded data from the real RADAR sensor hardware during real test drives with synthetically generated data from the virtual sensor model during simulation.

The main outcome of assessment is based on calculation of introduced criteria maximum and average deviation of longitudinal displacement $\Delta\text{Dist}_x\text{max}$ and $\Delta\text{Dist}_x\text{ave}$, maximum and average deviation of relative longitudinal velocity $\Delta\text{Vrel}_x\text{max}$ and $\Delta\text{Vrel}_x\text{ave}$ and graphical comparison of data. It shows that the considered manoeuvre, with the ego vehicle following one target vehicle (**Following test**) on a straight lane without lateral offset and small longitudinal displacement, is only limited applicable for the evaluation of all parameters. Therefore, further manoeuvres with

- at least 2 target vehicles, whereby they are located behind each other and then next to each other ⇒ testing separability (**Separation test**),
- increased longitudinal displacement between ego and target vehicle ⇒ testing influence of distance (**Range test**) and
- lat. offset between ego and target vehicle ⇒ verification of angle measurement (**Lateral offset test**)

are recommended, in order to be able to acquire meaningful statements regarding the impact of all different parameters on the output of the sensor model on the one hand, and on the validity of the HiFi RADAR sensor model on the other hand. In table 4.1 parameters and their impact on the output of the different manoeuvres and thus, their testability within these manoeuvres are illustrated in a matrix.

To make an efficient usage of the presented methodology for a great number of driving manoeuvres possible, further improvements are necessary and recommended:

- Definition of trajectory of ego and target vehicle based on real measurements has to be improved.

- Filtering of data from the real RADAR sensor hardware, recorded during real test drive, to obtain relevant data of the target vehicles and integration of this data into MATLAB® for further processing should be automatised.

Table 4.1: Parameter - Manoeuvre Matrix

Manoeuvre / **Parameter**	**Following (1 Target)**	**Separation (2 Targets)**	**Range (1 Target)**	**Lat. offset (1 Target)**
Reference temperature	√	√	√	√
Rain rate	√	√	√	√
Visual range in fog	√	√	√	√
Distance accuracy	√	√	√	√
Distance resolution	X	√	X	X
Azimuth accuracy	X	√	X	√
Azimuth resolution	X	√	X	X
Speed accuracy	√	√	√	√
Speed resolution	X	√	X	X
Separability	X	√	X	X
Transmit frequency	√	√	√	√
Transmit power	√	√	√	√
System losses	√	√	√	√
Noise bandwidth	√	√	√	√
Noise figure	√	√	√	√
Probability of detection min	√	√	√	√
Probability of false alarm	√	√	√	√
Field of view azimuth	√	√	√	√
Field of view elevation	√	√	√	√
Antenna efficiency	√	√	√	√

List of Figures

List of Tables

Bibliography

[1] IEEE Standard Definitions of Terms for Antennas. *IEEE Std 145-1993*, pages 1–32, 1993.

[2] Leica Geosystems AG. Leica TPS1200 Series - High performance Total Station. Leica TPS1200 Technical Description, 2005.

[3] C. Balanis. *Antenna Theory: Analysis and Design*. John Wiley & Sons, New Jersey, 3rd edition, 2005.

[4] BASELABS. Sensor Models for Virtual ADAS Sensors. Available at `https://www.baselabs.de/sensor-models-for-a-more-realistic-simulation/`. Accessed March 2018.

[5] S. Bernsteiner. *Integration of Advanced Driver Assistance Systems on Full-Vehicle Level - Parametrization of an Adaptive Cruise Control System Based on Test Drives*. PhD thesis, Graz University of Technology, 2015.

[6] S. Bernsteiner, Z. Magosi, D. Lindvai-Soos, and A. Eichberger. Radar Sensor Model for the Virtual Development Process. *ATZelektronik worldwide*, 10(2):46–52, 2015.

[7] Statistisches Bundesamt. Unfallentwicklung auf deutschen Straßen 2015. Available at `https://www.destatis.de/DE/PresseService/Presse/Pressekonferenzen/2016/Unfallentwicklung_2015/Pressebroschuere_unfallentwicklung.pdf?__blob=publicationFile`, 2016. Accessed April 2018.

[8] P. Cao, W. Wachenfeld, and H. Winner. Perception sensor modeling for virtual validation of automated driving. *it - Information Technology*, 57(4):243–251, 2015.

[9] European Commission. Roadmap to a Single European Transport Area - Towards a competitive and resource efficient transport system. *WHITE PAPER*, 144, 2011.

[10] European Commission. Road fatalities in the EU since 2001. Available at `https://ec.europa.eu/transport/road_safety/specialist/statistics_en`, 2017. Accessed April 2018.

[11] Magellan Corporation. Z-Xtreme GPS Receiver Operation and Reference Manual. Available at `https://ashgps.com/mirror/master/Z-Xtreme/Manuals/ZX%20Operations%20and%20Reference%20Manual.pdf`, 2001. Accessed March 2018.

[12] L. Danielsson. *Tracking and radar sensor modelling for automotive safety systems*. PhD thesis, Chalmers University of Technology, 2010.

[13] F. de Ponte Müller. Survey on Ranging Sensors and Cooperative Techniques for Relative Positioning of Vehicles. *Sensors*, 17(2), 2017.

[14] E. Donges. A Conceptual Framework for Active Safety in Road Traffic. *Vehicle System Dynamics*, 32(2-3):113–128, 1999.

[15] F. Fölster. *Erfassung ausgedehnter Objekte durch ein Automobil-Radar*. PhD thesis, Hamburg-Harburg University of Technology, 2006.

[16] Continental Engineering Services GmbH. Datasheet ARS308. Available at `https://www.continental-automotive.com/getattachment/9b6de999-75d4-4786-bb18-8ab64fd0b181/ARS30X-Datasheet-EN.pdf.pdf`, 2009. Accessed January 2018.

[17] Continental Engineering Services GmbH. Technical Documentation ARS308. Standardized ARS Interface - V1.15, 2012.

[18] IDC EDV GmbH. Geosi VERM Data Sheet. Available at `http://www.idc-edv.at/wp-content/plugins/pdfjs-viewer-shortcode/pdfjs/web/viewer.php?file=http://www.idc-edv.at/datenblaetter/GeosiVERM_V18.pdf`, 2017. Accessed March 2018.

[19] IPG Automotive GmbH. Reference Manual. IPG CarMaker Version 6.0.3, 2017.

[20] IPG Automotive GmbH. User's Guide. IPG CarMaker Version 6.0.3, 2017.

[21] M. Grgic. Generic Radar Model for Automotive Applications. Master's thesis, Graz University of Technology, 2015.

[22] O. Günther. *Modellierung und Leakage-Kompensation von 77GHz FMCW-Weitbereichsradar-Transceivern in SiGe-Technologie für Kfz-Anwendungen*. PhD thesis, Erlangen-Nürnberg University, 2008.

[23] A. Hein. *Verarbeitung von SAR-Daten unter besonderer Berücksichtigung interferometrischer Anwendungen*. PhD thesis, Siegen University, 1998.

[24] N. Ingale. ADAS system - Advancement of Driver Assistant System (DAS). Available at `https://www.linkedin.com/pulse/adas-system-advancement-driver-assistant-das-nikhil-ingale`, 2017. Accessed March 2018.

[25] Z. Magosi. Modellbildung und Simulation eines Radarsensors für die virtuelle Entwicklung von Fahrerassistenzsystemen. Master's thesis, Graz University of Technology, 2013.

[26] M. M. Meinecke and H. Rohling. Combination of LFMCW and FSK Modulation Principles for Automotive Radar Systems. German Radar Symposium GRS2000, October 11-12 2000.

[27] R. Mende. *Radarsysteme zur automatischen Abstandsregelung in Automobilen*. PhD thesis, Carolo-Wilhelmina zu Braunschweig University of Technology, 1999.

[28] European New Car Assessment Programme (Euro NCAP). 2020 Roadmap. Euro NCAP, 2015.

[29] M. Nentwig. *Untersuchungen zur Anwendung von computergenerierten Kamerabildern für die Entwicklung und den Test von Fahrerassistenzsystemen.* PhD thesis, Erlangen-Nürnberg University, 2013.

[30] W. L. Oberkampf and T. G. Trucano. Verification and validation benchmarks. *Nuclear Engineering and Design*, 238(3):716–743, 2008.

[31] Society of Automotive Engineers (SAE). Taxonomy and Definitions for Terms Related to Driving Automation Systems for On-Road Motor Vehicles, 2016.

[32] Graz University of Technology. Driving Simulator. Available at `https://www.tugraz.at/institute/ftg/forschung/labor/fahrsimulator/`, 2017. Accessed March 2018.

[33] S. Patole, M. Torlak, D. Wang, and M. Ali. Automotive Radars: A review of signal processing techniques. *IEEE Signal Processing Magazine*, 34:22–35, 2017.

[34] M. Richards, J. Scheer, and W. Holm. *Principles of Modern Radar - Basic Principles.* SciTech Publishing Inc., United States of America, 2010.

[35] H. Rohling. Radartechnik und -signalverarbeitung. Hamburg-Harburg University of Technology - Lecture notes, 2007.

[36] E. Roth, T. Dirndorfer, A. Knoll, K. Neumann-Cosel, T. Ganslmeier, A. Kern, and M. O. Fischer. Analysis and Validation of Perception Sensor Models in an Integrated Vehicle and Environment Simulation. *22nd International Technical Conference on the Enhanced Safety of Vehicles (ESV)*, 2011.

[37] A. Schaermann, A. Rauch, N. Hirsenkorn, T. Hanke, R. Rasshofer, and E. Biebl. Validation of Vehicle Environment Sensor Models. *2017 IEEE Intelligent Vehicles Symposium (IV)*, pages 405–411, 2017.

[38] C. Schmidth. *Hardware-in-the-Loop gestützte Entwicklungsplattform für Fahrerassistenzsysteme - Analyse und Generierung kritischer Verkehrsszenarien.* PhD thesis, Kassel University, 2010.

[39] H. Schreiber. Radartechnik. Graz University of Technology - Lecture notes, 2016.

[40] R. Schubert, N. Mattern, and R. Bours. Simulation von Sensorfehlern zur Evaluierung von Fahrerassistenzsystemen. *ATZ*, 9:38–41, 2014.

[41] M. I. Skolnik. *Introduction to Radar Systems.* McGraw-Hill, Singapore, 2nd edition, 1981.

[42] M. I. Skolnik. *Radar Handbook.* McGraw-Hill, 3rd edition, 2008.

[43] J. Steinbaeck, C. Steger, G. Holweg, and N. Druml. Next generation radar sensors in automotive sensor fusion systems. *2017 Sensor Data Fusion: Trends, Solutions, Applications (SDF)*, pages 1–6, 2017.

[44] Keysight Technologies. A Framework for Understanding: Deriving the Radar Range Equation. Available at `http://literature.cdn.keysight.com/litweb/pdf/5992-1386EN.pdf`, 2017. Accessed December 2017.

[45] Iowa University. The National Advanced Driving Simulator. Available at `https://www.nads-sc.uiowa.edu/sim_nads1.php`, 2014. Accessed March 2018.

[46] T. A. Wheeler, M. Holder, H. Winner, and M. J. Kochenderfer. Deep Stochastic Radar Models. *2017 IEEE Intelligent Vehicles Symposium (IV)*, pages 47–53, 2017.

[47] H. Winner, S. Hakuli, F. Lotz, and C. Singer. *Handbuch Fahrerassistenzsysteme: Grundlagen, Komponenten und Systeme für aktive Sicherheit und Komfort*. Springer Vieweg, Wiesbaden, 3rd edition, 2015.

[48] J. Xu, W. Hong, H. Zhang, G. Wang, Y. Yu, and Z. H. Jiang. An Array Antenna for Both Long- and Medium-Range 77 GHz Automotive Radar Applications. *IEEE Transactions on Antennas and Propagation*, 65(12):7207–7216, 2017.