Technische Universität München
Ingenieurfakultät Bau Geo Umwelt
Fachgebiet für Risikoanalyse und Zuverlässigkeit

Safety Assessment of Environment Perception in Automated Driving Vehicles

Mario Jürgen Berk

Vollständiger Abdruck der von der Ingenieurfakultät Bau Geo Umwelt der Technischen Universität München zur Erlangung des akademischen Grades eines

Doktor-Ingenieurs (Dr.-Ing.)

genehmigten Dissertation.

Vorsitzender: Prof. Dr.-Ing. Fritz Busch

Prüfer der Dissertation:

1. Prof. Dr. Daniel Straub
2. Prof. Dr.-Ing. Markus Lienkamp
3. Prof. Matteo Pozzi, Ph.D.

Die Dissertation wurde am 21.05.2019 bei der Technischen Universität München eingereicht und durch die Ingenieurfakultät Bau Geo Umwelt am 05.09.2019 angenommen.

Acknowledgements

I would like to express my deepest gratitude to all the individuals who contributed to this Ph.D. thesis and supported me throughout the past years. Also, I am very thankful for the funding of this project by AUDI AG, which made this research possible.

Above of all, very special thanks go to Prof. Dr. Daniel Straub, who sparked my interest in reliability analysis and probability theory. Prof. Straub was the most exceptional supervisor, whose personal commitment and effort in supporting this thesis were truly remarkable. Without his ability to provide helpful feedback, this thesis would not have been possible. I would like to thank him very much for this opportunity and for teaching me so much on a professional and personal level.

This research was conducted in the INI.TUM framework as a collaborative project between the Engineering Risk Analysis group of Technische Universität München and the development radar- / laser sensors automated driving, AUDI AG. I would like to express my very great appreciation to all involved persons at AUDI AG. I thank Dr. Boris Buschardt very much for this opportunity and for his confidence shown to me. He provided valuable input and support for this thesis, and a working environment that allowed me to focus on this research. Special thanks go to Hans-Martin Kroll, whose trust was crucial for initiating this project. I thank him very much for teaching me the fundamentals in automated driving and for being my mentor at AUDI AG with invaluable advice. The supervision by Dr. Olaf Schubert was greatly appreciated. His deep analytical mindset and his experience with sensor technology were of great help to this thesis. I am very thankful for the time he spent supervising and advising me.

I like to thank Prof. Dr.-Ing. Fritz Busch for mentoring me during this Ph.D. project at Technische Universität München.

It is thanks to all my colleagues that I felt at home at both, the Engineering Risk Analysis group and at AUDI AG. I am very glad for the familiar working atmosphere at the offices and for the technical discussions I had with my colleagues, which all contributed to shaping this thesis.

Finally, I am particularly grateful for the support and motivation provided by my family and friends. My parents enabled me to follow this path, did everything to help me focus on my thesis and stayed at my side when most needed. To all my friends, who helped me spend some time not thinking about my thesis. And to Ani, who was there for me in difficult times and who encouraged me in the final phase of this work.

Mario Berk,
26. March 2019

Abstract

Automated driving systems (ADS) promise to increase traffic safety. Central challenges in the development of ADSs with higher levels of driving automation (≥ level 3) are to derive safety requirements for its components and to actually validate system safety. Important for ADS safety is the perception provided by the fused data of environment sensors, e.g. radar, camera and lidar. Perception errors can be safety-critical and can in the worst case cause accidents. Existing standards and test procedures do not directly allow to validate sufficient perception reliability, which we define as probability of absence of safety-critical perception errors. Well known in this context is the approval trap for ADSs, i.e. the impracticably large number of test kilometers required to validate the safety of the intended ADS functionality by simply "driving around".

These challenges motivate the development of perception reliability analysis methods to ensure ADS safety. To this end, we derive reliability requirements for individual sensors. This is based on 1.) a stochastic description of perception reliability, 2.) a conceptualization of sensor data fusion in terms of individual sensors with a simple k-out-of-n-model and 3.) statistical dependence models for sensor errors. We additionally devise methods to assess sensor perception reliability through simulation, on proving grounds and in field tests. A challenge in assessing sensor perception reliability in field tests is the need for a reference truth to identify sensor errors, which is not always available with the required specifications. For this reason, we propose an approach to learn sensor perception reliabilities without a reference truth by exploiting sensor redundancy. We find in synthetic case studies that one can theoretically learn sensor perception reliabilities by exploiting sensor redundancy, if an adequate dependence model is selected for errors among different sensors. With an inadequate dependence model, the estimates can however be biased.

This thesis comprehensively studies how to demonstrate perception reliability for ADS safety validation. With the presented safety validation strategy, an empirical demonstration of perception reliability is in principle possible, as long as the models adequately represent the system. Additionally, the proposed approach to exploit sensor redundancies is an opportunity to cover a large number of realistic test situations by learning reliabilities from a fleet of driver controlled vehicles, equipped with standard series sensors. Considerable challenges in demonstrating perception reliability however remain: the difficulty of a systematic definition of safety-relevant perception errors, potentially improving simplifying modeling choices such as the k-out-of-n reliability model for sensor data fusion, the evaluation of adequate dependence models for the fleet learning framework and dealing with system design modifications during development represent future research opportunities.

Zusammenfassung

Durch das automatisierte Fahren verspricht man sich eine erhöhte Verkehrssicherheit. Große Herausforderungen in der Entwicklung des automatisierten Fahrens sind jedoch eine Wahl geeigneter Sicherheitsanforderungen für Komponenten sowie eine Validierung der Systemsicherheit (speziell für ≥ Level 3). Zentral dabei ist die fusionierte Umfeldwahrnehmung von Radar, Kamera und Lidar Sensoren. Sensorische Wahrnehmungsfehler können zu sicherheitskritischem Fehlverhalten und schlimmstenfalls zu Unfällen führen. Etablierte Standards und Testverfahren reichen für einen Nachweis der benötigten Wahrnehmungszuverlässigkeit (Wahrscheinlichkeit der Abwesenheit von sicherheitskritischen Wahrnehmungsfehlern) nur teilweise aus. Ein Beispiel für die Limitierung bestehender Testverfahren ist die Absicherungsfalle, die den kaum zu bewältigenden empirischen Absicherungsaufwand für die Freigabe des automatisierten Fahrens durch „Herumfahren“ bezeichnet.

Diese Herausforderungen motivieren die Entwicklung von Absicherungsmethoden für die Umfeldwahrnehmung. Um die Sicherheit eines automatisierten Fahrsystems zu gewährleisten, leiten wir Anforderungen an einzelne Sensoren ab. Dazu verknüpfen wir 1.) eine stochastische Beschreibung der Wahrnehmungszuverlässigkeit, 2.) eine Konzeptualisierung der Sensordatenfusion durch Einzelsensoren mit einem k-aus-n Modell, und 3.) statistische Modelle für Wahrnehmungsfehler. Zusätzlich entwickeln wir Methoden, um die Wahrnehmungszuverlässigkeit der Sensoren über Simulationen, sowie in Prüfgelände- und Feldtests zu lernen. In Feldtests ist die Erhebung einer Referenzwahrheit zur Identifikation von Wahrnehmungsfehlern jedoch herausfordernd. Daher entwickeln wir einen Ansatz, um die Wahrnehmungszuverlässigkeit ohne Referenzwahrheit unter Ausnutzung von Sensorredundanzen zu lernen. Wir zeigen auf, dass damit die Wahrnehmungszuverlässigkeit theoretisch korrekt gelernt werden kann, solange das verwendete Abhängigkeitsmodell die Sensorfehler adäquat beschreiben kann. Ist dies nicht der Fall, ist die Abschätzung verfälscht.

In dieser Dissertation erarbeiten wir Methoden für einen Nachweis der Wahrnehmungszuverlässigkeit für die Absicherung des automatisierten Fahrens. Mit der entwickelten Strategie ist ein empirischer Nachweis der Wahrnehmungszuverlässigkeit prinzipiell möglich, soweit die verwendeten Modelle das System adäquat beschreiben können. Des Weiteren bietet der Ansatz unter Ausnutzung von Sensorredundanzen die Chance eine hohe Testabdeckung durch eine Fahrzeugflotte zu erreichen. Es sind aber noch nicht alle Herausforderungen gelöst: Eine systematische Definition von sicherheitskritischen Wahrnehmungsfehlern, vereinfachende Annahmen wie das k-aus-n Modell für die Sensordatenfusion, die Validierung von geeigneten Abhängigkeitsmodellen für den flottenbasierten Absicherungsansatz und die Berücksichtigung von Systemänderungen stellen Forschungsmöglichkeiten dar.

Contents

Nomenclature

The symbols used in this work are defined in the text when first introduced. They are additionally summarized in the list below. Following conventions on nomenclature are pointed out:

- Some common symbols for variables (e.g. X) are defined locally, if explanations are relevant for a specific Chapter or Section only. In case of ambiguities, local definitions are highlighted in the list below to clarify the scope of a variable.
- Counter indices (subscripts such as j) may refer locally to different sets.
- Bold faced symbols represent vectors or matrices.
- Random variables are denoted with upper case letters (e.g. X) and their respective realizations with lower case letters (e.g. x).
- The probability of a random variable X taking a value x is denoted with $\Pr(X = x)$. Equivalently used in this Thesis is $p_X(x) = \Pr(X = x)$.
- $p_{X|\boldsymbol{\theta}}(x|\boldsymbol{\theta})$ is the conditional probability mass function of $X = x$ given $\boldsymbol{\theta}$.
- $f_X(x)$ is the probability density function of X and $F_X(x)$ the corresponding cumulative distribution function.
- $f_{X|\boldsymbol{\theta}}(x|\boldsymbol{\theta})$ is the conditional probability density function of X given $\boldsymbol{\theta}$.
- Prior and posterior probability density functions do not carry subscripts, e.g. $f(\boldsymbol{\theta})$ is the prior probability density function of $\boldsymbol{\theta}$ and $f(\boldsymbol{\theta}|\mathbf{x})$ is the posterior probability density function of $\boldsymbol{\theta}$ given $\mathbf{x}$.
- $F(\boldsymbol{\theta}|\mathbf{x})$ is the posterior cumulative distribution function of $\boldsymbol{\theta}$ given $\mathbf{x}$.
- $\mathbb{I}(x = 1)$ is the indicator function which is one if $x = 1$ is true, and zero otherwise.
- $\propto$ is the proportionality symbol.

Roman Symbols

a	Shape parameter of gamma distribution
$a(\hat{s}(s))$	Selected automated driving system action given $\hat{s}(s)$
$a(s)$	Selected automated driving system action given s
a'	Gamma shape parameter in prior distribution
a''	Gamma shape parameter in posterior distribution
$A_{\mathrm{B}}(R)$	Lidar beam cross section at distance R [m^2]
A_{R}	Aperture area of optical receiver [m^2]
A_{T}	Cross sectional area of target [m^2]
b	Base width stereo camera [m] (defined locally in Section 2.1.2)
b	Inverse scale parameter of gamma distribution (defined locally in Section 4.3)
b'	Gamma inverse scale parameter in prior distribution
b''	Gamma inverse scale parameter in posterior distribution
$c(E)$	Consequences of context variable E on perception performance
c_{F_j}	Consequences associated with failure event F_j
c_0	Speed of light [$\mathrm{m/s}$]
d	Disparity (defined locally in Section 2.1.2)
d	Displacement between optical axes of receiver and transmitter [m] (defined locally in Section 5.2)
D, d	Binary event of indicating an object
D, d	Binary object indications in different sensors (detection vector)
D_{drop}	Drop diameter [m]
$D_{i,m}, d_{i,m}$	Binary event of indicating an object in sensor i at point in time m
D_i, d_i	Binary event of indicating an object in sensor i
D_{lens}	Diameter of receiving lense (lidar) [m]
$\mathbf{D}_m, \mathbf{d}_m$	Binary object indications in different sensors at point in time m (detection vector)
E, e	Context variables
$\tilde{\mathbf{e}}$	Future context variables corresponding to future observables $\tilde{\mathbf{x}}$
$\mathbf{e}_j$	Realization of context variables in time interval j
E	The event of a situation in which a hazard can arise (exposure in ISO 26262, locally defined in Section 3.2.1)
E[ ]	Expectation operator
f_0	Constant approximation of transmit frequency [s^{-1}]
f_{length}	Focal length camera [m]
f_{R}	Receive signal frequency [s^{-1}]

f_{T}	Transmit signal frequency [s^{-1}]
F_j	Automated driving system failure event (i.e. accident) of type j
FN	False negative error
FP	False positive error
$g()$	Arbitrary function
g^{-1}	Inverse of $g()$
G_{R}	Receive antenna gain
G_{T}	Transmit antenna gain
H	Spatial impulse response function [m^{-2}]
H_{C}	Spatial impulse response of optical channel [m^{-2}]
H_{T}	Spatial impulse response of targets
i	Identifier (subscript) of a specific individual sensor
I, i	Rainfall intensity (random variable) [mm/h]
K	Number of data points in time interval j (defined locally in Section 5.3)
K, k	Number of sensors indicating an object (defined locally in Section 4.2)
K, k	Number of errors in n redundant sensors (defined locally in Section 6.1)
$L_z(p_{\mathrm{av}}, \rho)$	Likelihood of p_{av} and ρ under observation z
m	Discrete point in time
M	Total number of observations / total number of discrete points in time
n	Number of redundant sensors
$n(R)$	Number of scattering particles in a beam volume element at distance R (defined locally in Section 5.2)
n_{drops}	Number of raindrops in volume element V_{beam}
n_{MCS}	Number of Monte Carlo samples
n_y	Number of occurrences of $Y = y$ in M observations
$\mathbf{n}_{Y=y}$	Vector containing the observed n_{y} for each value of $Y = y$
n_z	Number of observations with $Z = z$ in a test with M observations
N	Number density of scattering particles per unit volume [m^{-3}] (defined locally in Section 5.2)
$N(\mu, \sigma^2)$	Normal probability density function with mean μ and standard deviation σ
O, o	Binary event of an object being present in a specific area of the field of view
O_m, o_m	Binary random variable indicating whether an object is present in a specific area of the field of view at point in time m
p	Generic probability (of e.g. failure or detection event), average of p_m (locally defined in Section 4.2.2)
p	Heuristical existence measure (locally defined in Section 4.1.1)

p_{av} Average probability of sensor errors in a large number of randomly selected points in time

$\hat{p}_{\mathrm{av}}$ Posterior mean of p_{av}

$p_{\mathrm{FN},j}$ Fraction of safety-critical false negative errors in area j

$p_{\mathrm{FP},j}$ Fraction of safety-critical false positive errors in area j

P_m, p_m Generic probability (of e.g. failure or detection event) at point in time m (locally defined in Section 4.2.2)

p_{obj} Probability of an object being present in a specific area of field of view

$p_{\mathrm{per}}(p_{\mathrm{av}}, \rho)$ Probability of the perception module to fail at a randomly selected point in time in function of p_{av} and ρ

$\hat{p}_{\mathrm{per}}$ Posterior mean of p_{per}

$p_{\mathrm{per}_{\mathrm{low}}}$ Lower credible bound on p_{per}

$p_{\mathrm{per}_{\mathrm{up}}}$ Upper credible bound on p_{per}

p_{s} Probability of a critical shock events (all sensors fail deterministically)

$p_{\mathrm{s}_{\mathrm{low}}}$ Lower credible bound on p_{s}

$p_{\mathrm{s}_{\mathrm{up}}}$ Upper credible bound on p_{s}

$p_{\mathrm{TLS}_{\mathrm{per}}}$ Target level of safety on p_{per} (acceptable p_{per})

p_y Probability of $Y = y$

$\mathbf{p}_y$ Vector containing p_{y} for all y

P_0 Peak transmit power [W]

PFA Probability of false alarm

PFA_i Probability of false alarm in sensor i

$\mathrm{PFA}_{\mathrm{lidar}}$ Lidar raw data probability of false alarm

$\mathrm{PFA}_{\mathrm{sensor},m}$ Probability of false alarm in individual sensor at point in time m, valid for all sensors $i = 1, \dots, n$

$\mathrm{PFA}_{\mathrm{sensor}}$ Probability of false alarm in individual sensor, valid for all sensors $i = 1, \dots, n$

POD Probability of detection

POD_i Probability of detection in sensor i

$\mathrm{POD}_{\mathrm{lidar}}$ Lidar raw data probability of detection

$\mathrm{POD}_{\mathrm{sensor},m}$ Probability of detection in individual sensor at point in time m, valid for all sensors $i = 1, \dots, n$

$\mathrm{POD}_{\mathrm{sensor}}$ Probability of detection in individual sensor, valid for all sensors $i = 1, \dots, n$

P_{R} Received power [W]

P_{T} Transmit power [W]

Q_{back} Backscattering efficiency

Q_{ext}	Extinction efficiency
$r_{\mathrm{R},0}$	Radius of reception aperture [m]
r_{R}	Radius of receiving channel's cross section [m]
$r_{\mathrm{T},0}$	Radius of transmission aperture [m]
r_{T}	Radius of transmit beam cross section [m]
R	Distance, target range [m]
$\mathbf{R}$	Covariance matrix
R_0	Distance to a hard target [m]
R_1	Minimum lidar range [m]
R_2	Optical channels of transmitter and receiver overlap completely at R_2 (lidar) [m]
$\mathbf{R}_{\mathrm{U,FN}}$	Correlation matrix of false negative sensor errors in standard normal space
$\mathbf{R}_{\mathrm{U,FN,+}}$	Adapted version of $\mathbf{R}_{\mathrm{U,FN}}$ with different signs
$\mathbf{R}_{\mathrm{U,FP}}$	Correlation matrix of false positive sensor errors in standard normal space
$\mathbf{R}_{\mathrm{U,FP,+}}$	Adapted version of $\mathbf{R}_{\mathrm{U,FP}}$ with different signs
$\widehat{\mathbf{R}}_{\mathrm{U,FN}}$	Correlation matrix containing $\hat{\rho}_{\mathrm{U_{FN},}i,j}$
$\widehat{\mathbf{R}}_{\mathrm{U,FP}}$	Correlation matrix containing $\hat{\rho}_{\mathrm{U_{FP},}i,j}$
s	True state of environment
$\hat{s}(s)$	Perceived environment given s
$s(d_i)$	Sign $(+1, -1)$ in dependence of d_i
S	Severity of damage (ISO 26262)
t	Time / test effort [h]
t_{cloudy}	(Test) time in cloudy weather [h]
t_{crit}	Time until an error becomes safety-critical [s]
t_{cycle}	Time to update the perception (cycle time) [s]
t_{rain}	(Test) time in rainy weather [h]
t_{snow}	(Test) time in snowy weather [h]
t_{sun}	(Test) time in sunny weather [h]
t_{t}	Detection threshld
t_{ToF}	Time-of-flight [s]
T	Atmospheric transmission factor (defined locally in Section 2.1.3)
T	Temperature (defined locally in Section 5.3)
TN	True negative detection event
TP	True positive detection event
T_{standard}	Standardized temperature
$u_{\mathrm{PFA},i}$	Inverse of standard normal cumulative distribution function with argument PFA_i
$u_{\mathrm{POD},i}$	Inverse of standard normal cumulative distribution function with argument POD_i

$\mathbf{U}$ Vector of standard normal random variables describing sensor object indications D_i

U_{c} Standard normal distributed auxiliary random variable

$U_{\mathrm{FN},i}$ Standard normal random variable describing true positive and false negative detections in sensor i

$U_{\mathrm{FP},i}$ Standard normal random variable describing false positive and true negative detections in sensor i

U_i Standard normal random variable describing individual sensor object indications D_i

v_{ego} Velocity of ego-vehicle $[\mathrm{m/s}]$

v_{rel} Relative velocity between transmitter and observer $[\mathrm{m/s}]$

$V_{\mathrm{beam}}(R)$ Volume element of beam at distance R $[\mathrm{m}^3]$

X, x Number of failure events in time interval t (defined locally in Section 4.3)

$\tilde{\mathbf{x}}$ Future observable data

$\mathbf{x}_{j,k}$ Data point k in time interval j (defined locally in Section 5.3)

$\mathbf{x}_j$ Combined data in time interval j (defined locally in Section 5.3)

$\mathbf{x}_{\mathrm{labeled}}$ Labeled data of binary object indications in different sensors

$\mathbf{x}_m$ Hidden true state of an object at point in time m (defined locally in Section 2.2.2)

X, x Generic data (defined locally in Section 5.3)

$\mathbf{X}, \widehat{\mathbf{X}}$ True and estimated object states (defined locally in Section 4.1.2)

X_i, x_i Binary variable for error occurrence in sensor i (defined locally in Section 6.1)

y_m Observation Y at point in time m

Y Variable identifying a specific combination of **D**

Y_m Variable identifying a specific combination of $\mathbf{D}_m$ at point in time m

Y_{n} Maximum deviation of state quantity in n observations

Z, z Observation without reference truth when not distinguishing between error types and different sensors (locally defined in Section 6.1)

z Collection of observations without reference truth when not distinguishing between error types and different sensors over time (locally defined in Section 6.1)

$\mathbf{z}_{1:m-1}$ History of sensor observations up to $m-1$ (defined locally in Section 2.2.2)

z_m Observation without reference truth when not distinguishing between error types and different sensors at point in time m (defined locally in Section 6.1)

$\mathbf{z}_m$ Noisy sensor observation of hidden true state (defined locally in Section 2.2.2)

$\tilde{\mathbf{z}}_m$ One step ahead future sensor state observations (defined locally in Section 2.2.2)

Greek Symbols

$\alpha(r)$	Extinction coefficient at distance r (defined locally in Section 5.2)
α	Parameter of beta distribution
β	Parameter of beta distribution
β_0	Constant factor influencing σ_j
β_1	Weight for temperature influence on σ_j
$\boldsymbol{\beta}$	Weights for the influence of $\mathbf{e}_j$ on $\boldsymbol{\theta}_j$ (defined locally in Section 5.3)
$\beta(R)$	Backscattering coefficient at distance R (defined locally in Section 5.2)
γ	Credibility level of complying with a target level of safety
γ_T	Divergence of transmit beam [°]
γ_R	Divergence of receive channel [°]
$\gamma(a, b \cdot \lambda)$	Incomplete gamma function with arguments a and $b \cdot \lambda$
Γ	Reflectivity of a hard target (defined locally in Section 5.2)
$\Gamma(a)$	Gamma function with argument a
δ	Dirac function
$\delta_{k,n}$	Kronecker delta, which is $\delta_{k,n} = 1$ if $k = n$ and $\delta_{k,n} = 0$ otherwise
Δf	Shift between instantaneous transmit and receive signal frequencies [s^{-1}]
$\Delta f_{\text{Doppler}}$	Doppler frequency shift [s^{-1}]
Δt	Duration of one trial in which a perception error can occur (measurement cycle) [s]
$\Delta x_{j,k}$	Object position deviation k in time interval j (defined locally in Section 5.3)
ΔX	Object position deviation (defined locally in Section 5.3)
$\Delta\mathbf{X}$	Deviation between estimated and true state (defined locally in Section 4.1.2)
ζ	Crossover function
η_T	Optical efficiency transmitter
η_R	Optical efficiency receiver
$\boldsymbol{\theta}$	Generic model parameter
$\widehat{\boldsymbol{\theta}}$	Global mean of model parameters over time intervals j
$\boldsymbol{\theta}_{\text{dep}}$	Model parameters for dependent sensor errors with Gaussian copula
$\boldsymbol{\theta}_{\text{dep,DS}}$	Model parameter under dependent sensor errors with low rank Gaussian copula
$\boldsymbol{\theta}_{\text{indep}}$	Model parameter under statistically independent sensor errors
$\boldsymbol{\theta}_{\text{indep,MAP}}$	Maximum a posteriori point estimate of $\boldsymbol{\theta}_{\text{indep}}$
$\boldsymbol{\theta}_j$	Model parameter in time interval j
λ	Generic failure rate [1/h]
$\bar{\lambda}$	Average of generic failure rate [1/h]
$\hat{\lambda}$	Posterior mean generic failure rate [1/h]

λ_{act}	Automated driving system actuation rate of failures [1/h]
λ_{Fail}	An item's failure rate [1/h]
λ_{F_j}	Rate of automated driving system failure events (i.e. accident) F_j of type j [1/h]
λ_{FN}	Rate of false negative errors [1/h]
$\lambda_{\mathrm{FN}_{\mathrm{crit},j}}$	Rate of safety-critical false negative errors of perception module in area j [1/h]
λ_{FN_i}	False negative error rate of individual sensor i [1/h]
$\lambda_{\mathrm{FN},j}$	Rate of false negative errors of perception module in area j [1/h]
λ_{FP}	Rate of false positive errors [1/h]
$\lambda_{\mathrm{FP}_{\mathrm{crit},j}}$	Rate of safety-critical false positive errors of perception module in area j [1/h]
λ_{FP_i}	False positive error rate of individual sensor i [1/h]
$\lambda_{\mathrm{FP},j}$	Rate of false positive errors of perception module in area j [1/h]
λ_{funct}	Rate of failures of automated driving function module [1/h]
λ_{per}	Rate of failures of perception module [1/h]
$\lambda_{\mathrm{sensor}}$	Generic sensor error rate [1/h]
λ_{sys}	Rate of automated driving system failures (i.e. fatal accidents) [1/h]
λ_{TLS}	Generic target level of safety (acceptable rate) [1/h]
$\lambda_{\mathrm{TLS}_{\mathrm{FN,crit}}}$	Acceptable safety-critical false negative error rate of perception module [1/h]
$\lambda_{\mathrm{TLS}_{\mathrm{FN},i}}$	Acceptable false negative error rate of individual sensor i [1/h]
$\lambda_{\mathrm{TLS}_{\mathrm{FN,sensor}}}$	Acceptable rate of false negative sensor errors, valid for all sensors $i = 1, \dots, n$ [1/h]
$\lambda_{\mathrm{TLS}_{\mathrm{FP,crit}}}$	Acceptable safety-critical false positive error rate of perception module [1/h]
$\lambda_{\mathrm{TLS}_{\mathrm{FP},i}}$	Acceptable false positive error rate of individual sensor i [1/h]
$\lambda_{\mathrm{TLS}_{\mathrm{FP,sensor}}}$	Acceptable rate of false positive sensor errors, valid for all sensors $i = 1, \dots, n$ [1/h]
$\lambda_{\mathrm{TLS}_{\mathrm{per}}}$	Acceptable failure rate of perception module [1/h]
$\lambda_{\mathrm{TLS}_{\mathrm{sys}}}$	Acceptable rate of automated driving system failures [1/h]
$\lambda_{\mathrm{U}_{\mathrm{FN},i}}$	Correlation factor determining the correlation coefficient $\rho_{\mathrm{U}_{\mathrm{FN},i,j}}$ if the correlation matrix is of Dunne-Sobel class
$\lambda_{\mathrm{U}_{\mathrm{FP},i}}$	Correlation factor determining the correlation coefficient $\rho_{\mathrm{U}_{\mathrm{FP},i,j}}$ if the correlation matrix is of the Dunnet-Sobel class
$\boldsymbol{\lambda}_{\mathrm{U}_{\mathrm{FP}}}$	Vector containing the correlation factors $\lambda_{\mathrm{U}_{\mathrm{FP},i}}$ for all i
$\boldsymbol{\lambda}_{\mathrm{U}_{\mathrm{FN}}}$	Vector containing the correlation factors $\lambda_{\mathrm{U}_{\mathrm{FN},i}}$ for all i
λ_{wave}	Wavelength [m]
$\mu(t)$	Mean number of failures in time interval t

μ_j	Mean of object position deviations in time interval j
μ_N	Average number of rain drops in a unit volume $[\mathrm{m}^{-3}]$
μ_μ	Mean of μ_j over time intervals j
ρ	Target reflectance
ρ	Correlation coefficient
ρ	Sensor error correlation coefficient (defined locally in Section 6.1)
$\hat{\rho}$	Posterior mean of ρ
ρ_{FN}	Pairwise correlation coefficient of false negative errors valid for all pairs of sensors
$\rho_{\mathrm{FN},i,j}$	Pairwise correlation coefficient of false negative errors in sensors i and j
ρ_{FP}	Pairwise correlation coefficient of false positive errors valid for all pairs of sensors
$\rho_{\mathrm{FP},i,j}$	Pairwise correlation coefficient of false positive errors in sensors i and j
$\rho_{\mathrm{U}_{\mathrm{FN},i,j}}$	Correlation coefficient among the random variables $U_{\mathrm{FN},i}$ and $U_{\mathrm{FN},j}$
$\rho_{\mathrm{U}_{\mathrm{FN},+,i,j}}$	Correlation coefficient among the random variables $U_{\mathrm{FN},i}$ and $U_{\mathrm{FN},j}$ with adapted sign
$\rho_{\mathrm{U}_{\mathrm{FP},i,j}}$	Correlation coefficient among the random variables $U_{\mathrm{FP},i}$ and $U_{\mathrm{FP},j}$
$\rho_{\mathrm{U}_{\mathrm{FP},+,i,j}}$	Correlation coefficient among the random variables $U_{\mathrm{FP},i}$ and $U_{\mathrm{FP},j}$ with adapted sign
$\hat{\rho}_{\mathrm{U}_{\mathrm{FN},i,j}}$	Posterior mean of $\rho_{\mathrm{U}_{\mathrm{FN},i,j}}$
$\hat{\rho}_{\mathrm{U}_{\mathrm{FP},i,j}}$	Posterior mean of $\rho_{\mathrm{U}_{\mathrm{FP},i,j}}$
$\sigma_{\mathrm{back},i}$	Backscattering cross section of particle i $[\mathrm{m}^2]$
$\sigma_{\mathrm{ext},i}$	Extinction cross section of particle i $[\mathrm{m}^2]$
$\overline{\sigma}_{\mathrm{ext}}$	Mean extinction cross section of scattering particles $[\mathrm{m}^2]$
σ_j	Standard deviation of object position deviations in time interval j
σ_{RCS}	Radar cross section $[\mathrm{m}^2]$
$\sigma_{\Delta X}$	Standard deviation of the deviation between estimated and true state
σ_μ	Standard deviation of μ_j over time intervals j
τ_p	Pulse width [s]
τ_φ	Hyperparameter for $\varphi_{\sigma,j}$ (precision)
φ	Standard normal probability density function
$\boldsymbol{\varphi}$	Collection of all random effect parameters (defined locally in Section 5.3)
φ_2	Bivariate standard normal probability density function
$\boldsymbol{\varphi}_j$	Random effect parameters in time interval j (defined locally in Section 5.3)
$\varphi_\mathrm{n}(\mathbf{u}, \mathbf{R})$	n-dimensional multivariate correlated standard normal probability density function with argument $\mathbf{u}$ and covariance matrix $\mathbf{R}$
$\varphi_{\sigma,j}$	Random effect for variability in σ_j

Φ	Standard normal cumulative distribution function
$\boldsymbol{\phi}$	Hyperparameter (defined locally in Section 5.3)
Φ^{-1}	Inverse standard normal cumulative distribution function
Φ_2	Bivariate standard normal cumulative distribution function
Φ_n	n dimensional multivariate correlated standard normal cumulative distribution function
ϕ_R	Geometric property to calculate circle-circle intersection of transmitter and receiver channel [°]
ϕ_T	Geometric property to calculate circle-circle intersection of transmitter and receiver channel [°]
$\boldsymbol{\phi}_\mu$	Hyperparameter for mean μ_j
$\boldsymbol{\phi}_\sigma$	Hyperparameter with influence on standard deviation σ_j

Abbreviations

ADAS	Advanced driver assistance systems
ADS	Automated driving system
AEB	Automatic emergency brake
ALARP	As low as reasonable possible
APD	Avalanche photo diode
ASIL	automotive safety integrity level
AUC	Area under the ROC curve
BASt	Bundesanstalt für Straßenwesen
CDF	Cumulative distribution function
CMOS	Complementary metal-oxide-semiconductor
CNN	Convolutional neural network
DGPS	Differential global positioning system
DIN	Deutsches Institut für Normung
E/E	Electrical and/or electronic systems
EM	Expectation maximization algorithm
ETA	Event tree analysis
FBD	Functional block diagram
FMCW	Frequency modulated continuous wave
FMEA	Failure mode and effects analysis
FN	False negative
FOV	Field of view
FP	False positive
FTA	Fault tree analysis
GAMAB	Globalement au moins aussi bon
GPS	Global positioning system
HAZOP	Hazard and operability study
HiL	Hardware-in-the-loop
HMM	Hidden Markov Model
IEC	International Electrotechnical Commission
ISO	International Organization for Standardization
JIPDA	Joint Integrated Probabilistic Data Association
LSS	Limit state surface
MAP	Maximum a posteriori estimate
MCMC	Markov Chain Monte Carlo
MCS	Monte Carlo Simulation

MEM	Minimum endogenous mortality
MiL	Model-in-the-loop
OEM	Original equipment manufacturer
PDF	Probability density function
PIN	Positive intrinsic negative
PMF	Probability mass function
RBD	Reliability block diagram
ROC	Receiver operating characteristic
SAE	Society of Automotive Engineers
SIL	Safety integrity level
SiL	Software-in-the-loop
SOTIF	Safety of the Intended Functionality
SSD	Single shot detector
TLS	Target level of safety
TN	True negative
ToF	Time-of-flight
TP	True positive
ViL	Vehicle-in-the-loop
VSL	Value of a statistical life
XiL	X-in-the-loop

Publications

This thesis is based on and in parts taken from our following publications:

M. Berk, H.-M. Kroll, O. Schubert, B. Buschardt, and D. Straub, “Zuverlässigkeitsanalyse umfelderfassender Sensorik: Eine stochastische Methodik zur Berücksichtigung von Umgebungseinflüssen am Beispiel von LiDAR Sensoren,” in *Fahrerassistenz und automatisiertes Fahren: 32. VDI-VW-Gemeinschaftstagung*, Wolfsburg, 2016, pp. 455–475.

M. Berk, H.-M. Kroll, O. Schubert, B. Buschardt, and D. Straub, “Bayesian Test Design for Reliability Assessments of Safety-Relevant Environment Sensors Considering Dependent Failures,” in *SAE Technical Paper 2017-01-0050*, 2017.

M. Berk, O. Schubert, H.-M. Kroll, B. Buschardt, and D. Straub, “Reliability assessment of safety-critical sensor information: Does one need a reference truth?” *IEEE Transactions on Reliability*, 2019.

M. Berk, O. Schubert, H.-M. Kroll, B. Buschardt, and D. Straub, “Absicherung der Umfeldwahrnehmung von hoch- und vollautomatisierten Fahrzeugen,” in *Fahrerassistenzsysteme und automatisiertes Fahren: 34. VDI/VW-Gemeinschaftstagung*, Wolfsburg, 2018, pp. 165–184.

M. Berk, O. Schubert, H.-M. Kroll, B. Buschardt, and D. Straub, “Exploiting Redundancy for Reliability Analysis of Sensor Perception in Automated Driving Vehicles,” *Accepted by IEEE Transactions on Intelligent Transportation Systems*, 2019.

M. Berk, M. Dura; J.R. Vargas Rivero, O. Schubert, H.-M. Kroll, B. Buschardt, and D. Straub, “A Stochastic Physical Simulation Framework to Quantify the Effect of Rainfall on Automotive Lidar,” in *SAE Technical Paper 2019-01-0134*, 2019.

M. Berk, O. Schubert, H.-M. Kroll, B. Buschardt, and D. Straub, “Assessing the Safety of Environment Perception in Automated Driving Vehicles,” *Submitted to SAE International Journal of Transportation Safety*, 2019.

1 Introduction

1.1 Background and Motivation

In the Living Machine (1935), science-fiction author David H. Keller envisions automated driving: "Old people began to cross the continent in their own cars. Young people found the driverless car admirable for petting. The blind for the first time were safe. Parents found they could more safely send their children to school in the new car than in the old cars with a chauffeur." [1]. Automated driving is not only a science-fiction motif but is researched since the 1950s [2–13]. Almost every original equipment manufacturer (OEM) has developed *automated driving system*[1] (ADS) prototypes and has announced the release of automated driving functionalities [15–28], which are enabled by environment perceiving sensors such as radar, lidar and cameras [29]. The old vision of automated driving with its "promise of salvation" [2] seems close to become reality.

In The Living Machine there is however a twist in the story: "Cars, without control, coursed the public highways, chasing pedestrians, killing little children, smashing fences. […] Traffic was paralyzed. The nation became panic-stricken. Schools were closed." [1]. The dystopia of automated driving vehicles coming to life to hunt people does not seem realistic[2], but the science-fiction pattern of the "wonderful and scary" ([30], authors translation) is reflected in emotions of society towards automated driving [31] and persists also in the discussion about the introduction of ADSs. ADSs are expected to increase safety, efficiency and mobility as well as to reduce land use and congestion [3, 31–38]. However, large challenges for a market introduction of ADSs are to develop safe ADSs and to demonstrate that they are actually safe enough [33, 35, 39–45].

Such a safety demonstration is a prerequisite for authorities, society, end-users, regulatory bodies, the insurance industry and OEMs to accept that ADSs make safety-relevant decisions with implications on human life [39, 44]. Therefore, criteria for rational decision making about ADS safety are required [39]. An important aspect for the safety of ADSs is the reliability of its environment perception, provided by radar, lidar and camera sensors, because perception errors can be safety-critical. In the context of ADS safety, this thesis studies how to describe, assess and demonstrate the safety-critical reliability of environment perception.

[1] This thesis uses *automated driving* instead of *autonomous driving* because the latter inadequately implies the connotation of an agent with "…capacity for self-governance" [14], e.g. by deciding whether, when and where to drive. In contrast to *autonomous driving*, *automated driving* should make clear that such decisions are made by a user. See [14] for a discussion on terminology.

[2] To give a correct account of The Living Machine: the gasoline industry was jealous of the automotive industry's success with automated driving vehicles and as revenge decided to purposely mix cocaine into gas. The cocaine made the automated driving vehicles act without control. [1]

1.2 Relevance

1.2.1 Levels of Driving Automation

To discuss the safety of automated driving, one has to differentiate the capabilities of ADSs. The capabilities of ADSs are classified by the Society of Automotive Engineers (SAE) J3016 [14] and the German Bundesanstalt für Straßenwesen (BASt) [38] in terms of different levels of driving automation. The definitions according to SAE and BASt are corresponding, except for the terminology of levels 3-4 and an additional level 5 in J3016 [45]. Figure 1.1 summarizes the different levels of driving automation, for their exact definition we refer to [14, 38].

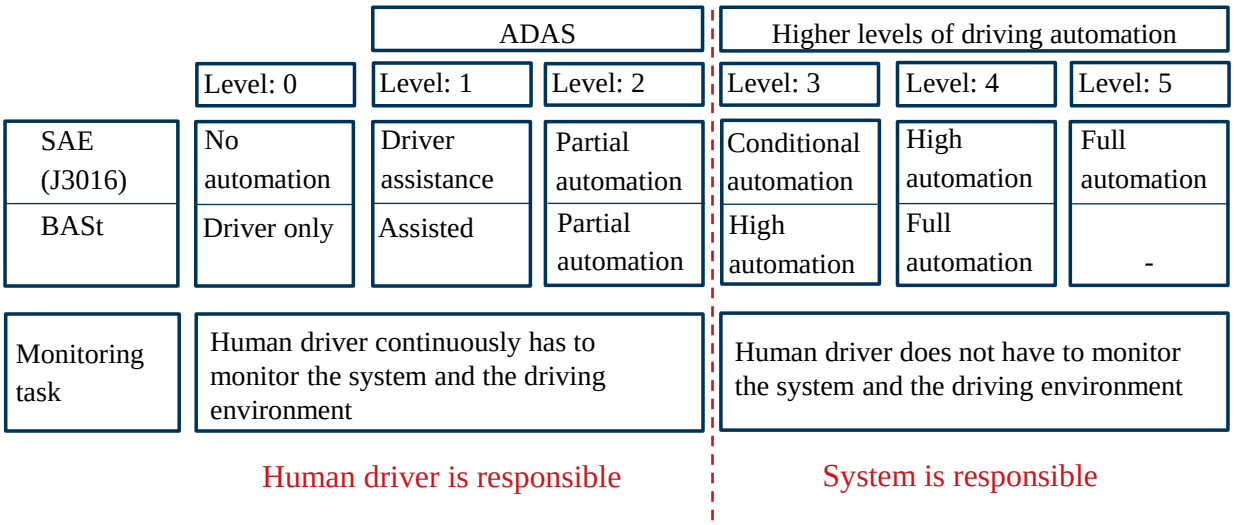

Figure 1.1 Levels of driving automation according to SAE J3016 [14] and German Bundesanstalt für Straßenwesen (BASt) [38]. Starting with level 3, the system is fully responsible for monitoring the system itself and the driving environment. In this thesis, levels 1-2 are summarized as advanced driver assistance systems (ADAS) and levels 3-5 as higher levels of driving automation[3].

Currently, the most advanced commercially available automated driving functionalities classify as level 2 systems (partial automation). A level 2 system is able to take over longitudinal and lateral vehicle control in specific driving situations, with the restriction that the driver must continuously monitor the system and the environment. In case of an error or inadequate behavior of the system, the driver is responsible for overriding the *advanced driver assistance system's* (ADAS's) action.

As indicated in Figure 1.1, a paradigm change occurs between a level 2 and a level 3 system because the human driver in level 3 does not have to monitor the system and the environment, when the system is engaged. Hence, the human driver is not responsible to react (immediately) in case of a system failure or error. The restriction of a level 3 system is that the driver has to be

[3] It is pointed out that the term *automated driving system* (ADS) in [14] refers to a system with driving automation levels 3-5, while *driving automation system* refers to levels 1-5. To avoid confusion, this thesis generally uses the term *automated driving system (*ADS*)* for levels 1-5, the term *higher levels of driving automation* for levels 3-5 and *advanced driver assistance systems* (ADAS) for levels 1-2.

receptive to take over control after an adequate time frame, whenever the system detects a failure or a situation it cannot handle. This restriction is not applicable to level 4 anymore, the system is expected to automatically reach a minimal risk condition[4], when it detects a system failure or a situation it cannot handle. While levels 3-4 are restricted to specific driving domains (e.g. highway), a level 5 system can handle any driving domain.

The human driver not being responsible for monitoring the ADS and its environment has profound implications on legal matters and liability[5] [33, 45, 49], on ethical questions related to ADS actions [50–52], on ADS design, on ADS safety and performance requirements, and on system safety validation and testing procedures [35, 40, 44].

1.2.2 Higher Levels of Driving Automation: Implications on Testing and Safety

In the following, it is assumed that it is legal for a human driver to not monitor the ADS and its environment (see e.g. [49] for a discussion on legal matters). The implications of *higher levels of driving automation* on safety, testing and validation under this assumption are explained with the three layer hierarchy of the driving task according to Donges [53]:

- Navigation: selection of an adequate driving route. Time span in the orders of minutes to hours.
- Vehicle guidance: selection of target driving variables such as the desired lane and velocity. Time span in the orders of seconds to minutes.
- Vehicle stabilization: aligning the current vehicle state with desired driving variables and stabilizing the motion of the vehicle. Time span in the orders of milliseconds to seconds.

A human driver is implicitly assumed to have the required perceptive, cognitive and sensory-motoric abilities to handle the navigation, guidance and stabilization of a vehicle. This assumption is commonly verified in a driver license test. Based on this assumption, traditional safety concepts for the vehicle-driver system aim at demonstrating that the vehicle components have sufficiently low failure rates and that a driver is safely able to control the vehicle (controllability). [44]

For instance, ISO 26262 based on a V-Model process is applied to design functional safe electrical and/or electronic (E/E) systems, i.e. to ensure the hardware and software of E/E systems have sufficiently low failure rates [54]. Controllability is demonstrated in exemplary test cases. If the

[4] The minimal risk condition, sometimes also termed the safe state, is for instance achieved by parking on the hard shoulder of a highway [46].

[5] Under the UN Vienna Convention on Road Traffic, the legal status of ADS with *higher levels of driving automation* was not clear: ”Every driver shall at all times be able to control his vehicle or to guide his animals.“ [47]. In 2016, the convention was amended, trying to clarify the legal status of ADS with *higher levels of driving automation* [48]. With this amendment, an ADS is acceptable, if the system is designed such that the driver is able to override the ADS’s actions and stop the ADS functionality at all times [48].

test driver is able to control the vehicle in these exemplary situations, it is assumed that another driver with a driver license is also able to control the vehicle in other relevant but not tested situations. Hence an integral part of the traditional automotive safety concept for the driver-vehicle system are the abilities of the drivers. [44]

For ADAS (levels 1-2), this safety concept does not change fundamentally [44]. Ultimately, the driver is responsible and hence the task is still to demonstrate controllability and sufficiently low component failure rates [44, 55, 56]. For example, the Code of Practice for the Design and Evaluation of ADAS summarized the state of the art in assessing controllability[6] of level 1-2 systems [40, 56]. An additional challenge in an ADAS safety validation compared to driver only systems is to provide a safe possibility for the driver to override the system's actions and to ensure that the automation does not lead to a decreased situation awareness of the driver, which impairs controllability [44].

With *higher levels of driving automation*, the testing for driver controllability is not applicable anymore because the human driver does not have to monitor the system and the environment. Per definition, the system has to be able to control and monitor the vehicle as well as its environment (in specific situations, depending on the driving automation level and when engaged). In contrast to a human driver, one cannot simply assume the required perceptive, cognitive and sensory-motoric abilities for an ADS to handle the navigation, guidance and stabilization of the vehicle. These abilities have to be tested and validated. With Donges' three layer model, Figure 1.2 illustrates which parts of the driver-vehicle system had to be tested traditionally (blue shaded area), and which parts have to be tested additionally in an ADS with *higher levels of driving automation* (red shaded area). [44]

The navigation task is not directly deemed to be safety relevant and is therefore not included in Figure 1.2. One of the main points in Figure 1.2 is that one has to validate the reliability of the system's perception and cognition to demonstrate the safety of an ADS with *higher levels of driving automation*. In ADSs, the perception is provided by e.g. radar, lidar and camera sensors, which gather information about the environment [29]. The information provided by the different sensors is typically combined in a sensor data fusion [57–60]. The fused sensor information is then the basis for the decision making of an ADS.

Applying existing standards such as ISO 26262, e.g. to demonstrate the functional safety of an ADS's environment perception, is not necessarily sufficient to ensure acceptable safety for *higher levels of driving automation* [40, 54, 61]. A failure in ISO 26262 is defined as the "termination of an element to perform a function as required" [62]. But even if environment perceiving sensors

[6] The Code of Practice defines controllability as the: "likelihood that the driver can cope with driving situations including ADAS-assisted driving, system limits and system failures" [56].

"perform as required", deficient information due to the inherently uncertain environment perception could lead to an inadequate ADS behavior, and in the worst case to an accident [40, 54, 63, 64]. This type of failure is as a distinction from classical (hardware / software) system failure events termed functional deficiency in [54, 64]. ISO 26262 reaches its limits in demonstrating a sufficiently low risk of functional deficiencies, because it is probably impossible to explicitly specify and verify requirements for all potential situations an ADS and its sensors have to handle. [40, 54, 61].

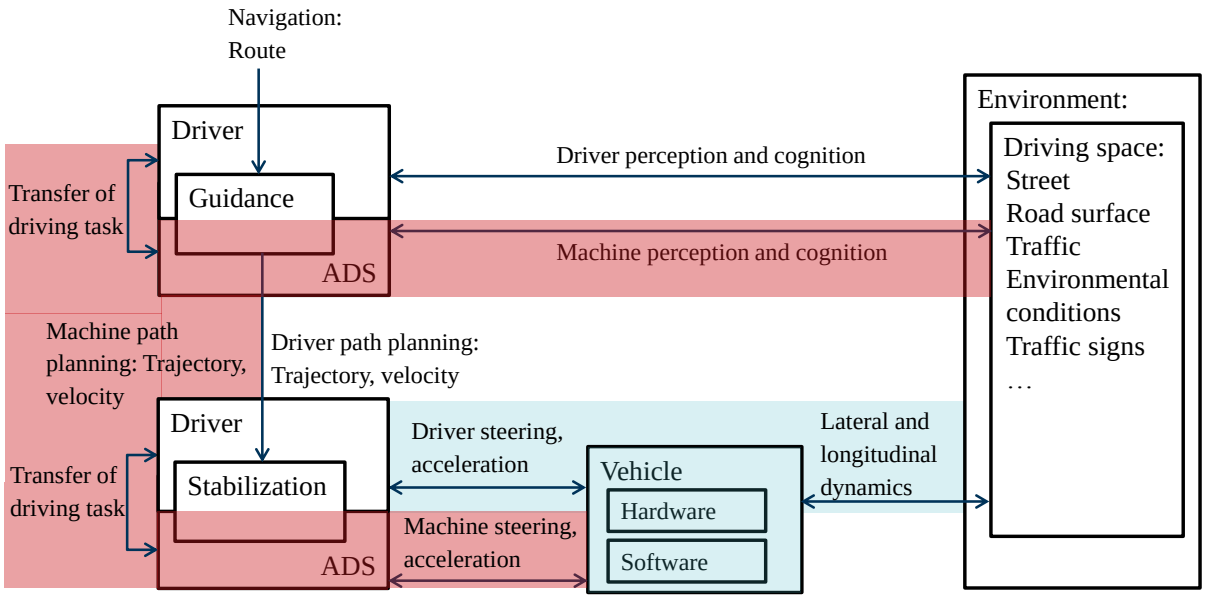

Figure 1.2 Validation and testing scope of an ADS with higher levels of driving automation (red and blue area) compared to traditional test concepts (blue area only). The safety relevance of transferring the driving task depends on the level of driving automation. In analogy to Fig. 21.6 in [44].

To preclude an unacceptable risk of functional deficiencies, one hence has to validate the reliability of the perception, provided by radar, lidar and camera sensors, with respect to the intended automated driving functionality [39]. To clarify that this task is not (entirely) identical to the demonstration of functional safety, the term *Safety of the Intended Functionality* (SOTIF) is used.

Hand in hand with the necessity of validating the SOTIF of ADSs goes the challenge that classical automotive testing procedures are not directly applicable to validate the safety of ADSs and their environment perception. Scenario based test methods by specifying tests in catalogues have the limitation that an ADS has to be able to control a near infinite number of situations [40, 44], field tests ("driving to safety" [41]) seem infeasible due to the large number of required test kilometers[7] [35, 41, 44, 65] and sufficiently realistic and comprehensive simulation methods for an exclusively simulation based safety demonstration do not yet exist [44, 66]. [39]

[7] The unmanageable amount of required test kilometers is referred to as the „approval trap" [35] for ADS.

1.2.3 Novelty of Challenge

One might think that the automation in other domains such as commercial aviation or railways leads to corresponding challenges. This is not the case. The safety concept of automated aviation systems and typically also of automated railway systems includes a supervision of the automated system by trained human operators. Hence these systems classify in analogy to the levels of driving automation as level 2. In contrast to automated road traffic, the automated aviation and railway systems are additionally externally controlled, which in case of railways is further supported by the infrastructure. Another crucial difference is that the traffic space in aviation and railways is closed, while the road traffic space is open. The open road traffic space leads to a high variability of traffic participants and objects that have to be handled by the ADS. Due to the open road traffic space, the road traffic flow cannot be planned a priori, which is in contrast to aviation and railways. This underlines the novelty of the challenge of demonstrating the safety of *higher levels of driving automation*, which depends on environment perception. [40, 44, 46]

The novelty of the challenge together with the previous discussion on implications of *higher levels of driving automation* underline the importance of validating an ADS's safety. According to [40, 63], with *higher levels of driving automation*, no methods to sufficiently address the safety validation of ADSs and their environment *perception reliability* are known.

1.3 Research Objectives and Scope of Thesis

Motivated by the discussed challenges, the subject of this thesis is the validation of *perception reliability* in the context of validating ADS safety. We define *perception reliability* as *probability of absence of safety-critical perception errors* at the output of sensor data fusion. An ADS's *perception reliability* depends on the reliability of the perception provided by individual radar, lidar and camera sensors, which we define as *sensor perception reliability*. In reference to [60], we interpret *environment perception* to answer questions in three uncertainty domains:

- Existence uncertainty: is there a relevant object in a specific area of the ADS's surroundings?
- Classification uncertainty: if there is a relevant object, what type of object is it? E.g. car, truck, pedestrian, lane marking, traffic sign, red traffic light, infrastructure, etc..
- State uncertainty: if there is a relevant object, what are its position, dimensions, velocity, acceleration, etc.?

With these definitions, the objectives of this thesis are to:

- comprehensively structure the task of validating ADS *perception reliability* in the context of an ADS's safety validation.
- identify and discuss challenges associated with validating ADS *perception reliability*.
- evaluate the applicability of existing automotive standards, safety validation frameworks and testing procedures on validating ADS *perception reliability*.
- describe *perception reliability* with suitable reliability metrics.
- devise general strategies for validating *perception reliability* in light of identified challenges.
- develop statistical methods to assess and estimate *sensor perception reliability*.

Within the scope of this thesis is *sensor perception reliability* and how errors in individual sensors can conceptually be connected to the *perception reliability* at the output of sensor data fusion. Not in scope of this thesis is an explicit modeling of sensor data fusion and an investigation of perception errors due to sensor data fusion, such as a wrong association of the information provided by individual sensors [59].

Also not in scope is the (reliability of the) interpretation of a perceived situation, which requires to comprehend the situation's meaning by relating the different elements in the situation to each other and to project a situation into the future [31, 56, 67]. To clarify the scope of this thesis, we exclude questions related to the situation interpretation such as for instance:

- Does a pedestrian try to cross the street?
- Where will a pedestrian be located in 3 s from now?
- Is a vehicle in front of the ego-vehicle (ADS) trying to change its lane?
- Will a vehicle be decelerating in 2s from now?
- What is an oncoming vehicle going to do at an intersection?
- Is the ego-vehicle allowed to turn left at an intersection at a specific point in time?

1.4 Structure of Thesis

We review the technical background of environment perception for ADSs in Chapter 2, including different sensor technologies, how an ADS represents its environment and how the data of different sensors is aggregated in a sensor data fusion. The reader familiar with environment perception for ADSs can skip Chapter 2.

Thereafter, in Chapter 3, we formally derive and define the task of validating *perception reliability*. We further identify challenges associated with validating and assessing *(sensor) perception reliability*. In light of these challenges, we assess the applicability of existing standards, safety frameworks and testing procedures on validating *perception reliability,* which highlights in detail the necessity of this research.

In Chapter 4, we describe *perception reliability* with suitable metrics, we derive requirements for *sensor perception reliability* from requirements on *perception reliability* and we estimate the test effort required to validate a required level of *perception reliability.*

Next, in Chapter 5, we present different methods to assess *(sensor) perception reliability*. These include a semi-quantitative analysis method, an exemplary virtual simulation, tests on proving grounds and field tests.

In Chapter 6, we investigate and develop statistical learning methods to estimate *sensor perception reliabilities* solely by exploiting sensor redundancies. Such an approach would facilitate the validation of *perception reliability* in Shadow-Mode [68], based on big data generated by a fleet of end-user vehicles.

Finally, we summarize our contributions, provide conclusions and give an outlook in Chapter 7.

2 Automated Driving and Environment Perception

The functional principle of an *automated driving system* (ADS) can be described with the classical robot control paradigm *sense, plan, act* [42, 69, 70]. Figure 2.1 presents a generic functional block diagram, detailing the *sense, plan, act* paradigm for ADSs.

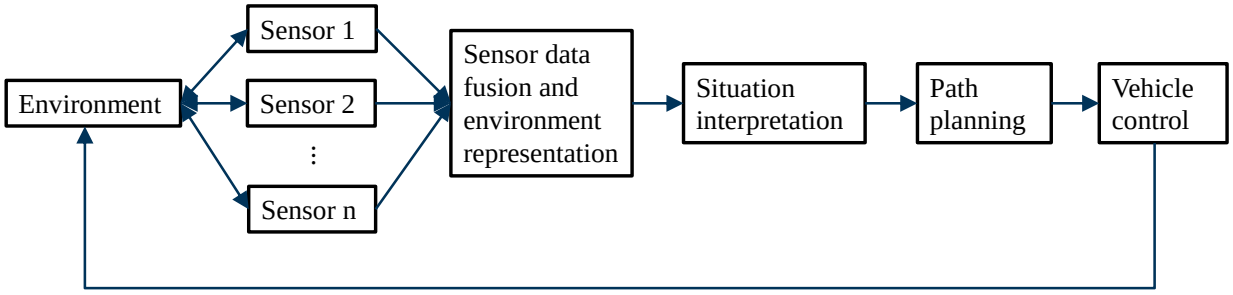

Figure 2.1 Generic functional block diagram of an automated driving system. In analogy to Fig 20.1 in [60].

Information about the environment is gathered by multiple environment perceiving sensors such as radar, camera and lidar sensors (described in Section 2.1) [29]. This information is combined in a sensor data fusion [57–60] to derive an abstract representation of the ego-vehicle's surroundings, which is called vehicle environment model (described in Section 2.2) [67]. Based on the vehicle environment model, the situation is interpreted by relating the different elements in the environment model to each other [31, 60, 71]. A path together with desired driving parameters (longitudinal and lateral acceleration, velocities…) is derived for the given situation, e.g. by minimizing a cost function for a given driving goal [31, 51]. The vehicle control (actuators for steering and acceleration) then execute the planned path by controlling the ego-vehicle's driving parameters [31]. The whole process is repeated and updated cyclically, when the sensors provide new information [60].

Figure 2.1 is a generic representation of a single automated driving functionality, the actual functional architecture of an ADS might differ. For instance, multiple fusion modules could be implemented for reasons of redundancy, or because they provide information to different automated driving functionalities. Also not indicated in Figure 2.1 is the fact that an ADS might use digital maps for localization [31, 60] and that the ADS makes use of information about the ego-vehicle's state, such as its velocity and acceleration [72].

Any of the ADS functionalities illustrated in Figure 2.1 could fail in a safety-critical way or make a safety-critical error, which in the worst case can lead to an accident. This highlights the safety-relevance of the environment perception, the path planning together with situation interpretation and the actuation.

2.1 Environment Perceiving Sensors

Our focus is on *perception reliability*. This Section describes the automotive sensing technology that enables the environment perception for ADSs. The aim is to provide an overview over the main physical principles of the different sensing technologies, to broadly discuss their signal and data processing algorithms and to discuss potential sensing errors together with the physical mechanisms that cause these errors. The purpose is not to provide a complete review of all technical details and of the latest sensor technology, but to support an understanding of the sensor technology as a basis for describing and assessing *(sensor) perception reliability*.

To this end, radar sensors are described in Section 2.1.1, camera sensors in Section 2.1.2 and lidar sensors in Section 2.1.3. Ultrasonic sensors are because of their small range (< 6 m, see e.g. [73]) mostly restricted to automated driving with small velocities, for instance in automated parking functionalities. The small velocities have limited safety-relevance, therefore ultrasonic sensors are not described here. For further information on ultrasonic sensors, we refer to [74, 75].

The strengths and weaknesses of different sensors are thereafter briefly summarized in Section 2.1.4. Finally, a generic representation of the sensors' signal and data processing is given in Section 2.1.5, which provides an abstract basis for describing *sensor perception reliability*.

2.1.1 Radar

This Section presents automotive radar technology in reference to [76] if not otherwise indicated.

The main purpose of radar (radio detection and ranging) sensors is to detect objects in the environment and to measure their distances and relative velocities. Radar technology goes back to military applications in the Second World War. The first commercial automotive implementation of radar sensors was in the context of Adaptive Cruise Control (ACC) in 1998. Thereafter, more radar use cases such as Automatic Emergency Braking (AEB) and Lane Change Assist (LCA) followed.

Object Detection

A radar transmits electromagnetic radiation, which is reflected back to the sensor by objects[8] in the environment. The radar's receiving antenna captures the reflected signal. The frequency bands utilized in the automotive domain include 21.65-26.65 GHz, 24.0-24.25 GHz and 76-77 GHz.

[8] In radar literature, relevant objects are termed targets, which goes back to its military origin. In this work the targets are objects in the environment of an ADS.

An object is for instance detected by the radar, if the signal intensity in the frequency domain exceeds a detection threshold [40, 76]. The received power P_{R} at the radar is described with the radar range equation [77], which helps to identify factors with influence on radar detection capabilities:

$$P_{\mathrm{R}} = \frac{P_{\mathrm{T}} \cdot G_{\mathrm{T}} \cdot G_{\mathrm{R}} \cdot \sigma_{\mathrm{RCS}} \cdot \lambda_{\mathrm{wave}}^2}{(4\pi)^3 \cdot R^4} \tag{2.1}$$

P_{T} is the peak transmit power. The gain of the transmit antenna G_{T} is the power density ratio of the antenna's directed radiation to an isotropic radiator with identical transmit power and quantifies the directivity of the radar. Additionally, G_{T} accounts for transmit losses. The gain of the receiving antenna G_{R} is generated by the receiver's aperture. G_{R} depends on the radiation's wavelength λ and the receiver losses. σ_{RCS} is an object's radar cross section, which determines how much of the incident radiation is reflected back towards the radar. σ_{RCS} is influenced by the size, shape, relative position [40], orientation [40] and the (surface) material of an object. The received power decreases with the distance to an object R to the power of four. Eq. (2.1) allows to estimate a radar's maximum range. Atmospheric losses are neglected in Eq. (2.1). [76, 77]

Distance and Relative Velocity Measurement

The distance and relative velocity measurements of radars are enabled by signal modulation, i.e. by encoding information into the transmit signal. By demodulating the received signal and comparing its properties to the transmit signal, the distance to and the relative velocity of an object are derived. A variety of modulation techniques are applied in the automotive domain, these include:

- Pulse modulation (Pulse Doppler method)
- Frequency-Shift-Keying (FSK)
- Linear Frequency Modulation Shift Keying (FMSK)
- Frequency Modulated Continuous Wave (FMCW)
- Chirp Sequence Modulation

It is not in the scope of this thesis to review all modulation techniques. Instead, the principle behind a Frequency Modulated Continuous Wave (FMCW) radar is presented to exemplarily explain how distances and relative velocities can be measured. For further information on radar signal processing it is referred to [76, 78–80].

A FMCW radar modulates the signal by linearly varying the instantaneous transmit signal frequency $f_{\mathrm{T}}(t)$ with a rate df_{T}/dt. As illustrated in Figure 2.2, the frequency of the received

signal $f_{\mathrm{R}}(t)$ at time t is shifted compared to $f_{\mathrm{T}}(t)$[9]: a) because of the time-of-flight (ToF) t_{ToF} in combination with the varying transmit frequency; and b) because of the Doppler frequency shift $\Delta f_{\mathrm{Doppler}}$.

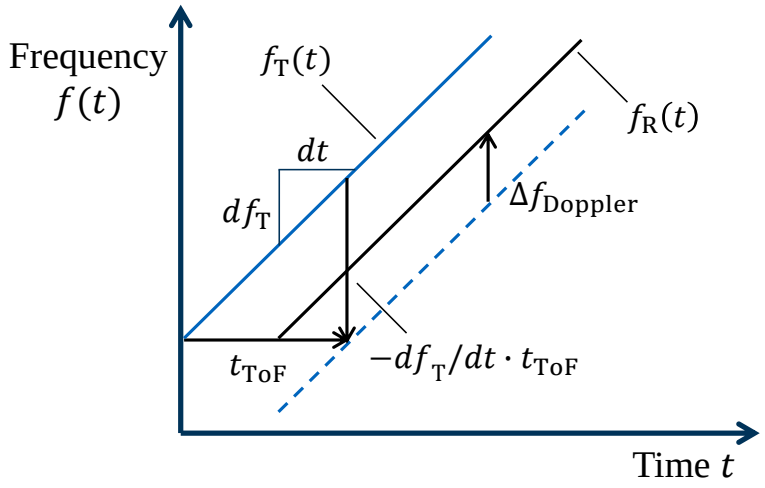

Figure 2.2 Relationship between transmit (solid blue line, $f_{\mathrm{T}}(t)$) and receive (solid black line, $f_{\mathrm{R}}(t)$) signal frequencies of a Frequency Modulated Continuous Wave (FMCW) radar. The illustrated Doppler Frequency shift $\Delta f_{\mathrm{Doppler}}$ represents the case in which sensor and object are approaching each other. The dashed line is the received signal's frequency in the stationary case. In analogy to Fig. 14 in [76] and to Bild 1 in [81].

t_{ToF} is the time needed for the signal to travel to an object at distance R and back to the sensor:

$$t_{\mathrm{ToF}} = \frac{2 \cdot R}{c_0} \tag{2.2}$$

with c_0 the velocity of light. The Doppler Effect describes the frequency shift $\Delta f_{\mathrm{Doppler}}$ a wave experiences due to the relative movement of an observer and a transmitter:

$$\Delta f_{\mathrm{Doppler}} = -\frac{2 \cdot v_{\mathrm{rel}} \cdot f_0}{c_0} \tag{2.3}$$

where v_{rel} is the relative velocity between observer and transmitter and f_0 is the frequency of the transmitted wave. f_0 is often approximated with a constant frequency within the utilized frequency band leading to a negligible error.

[9] The notation for the frequencies $f_{\mathrm{T}}(t)$ and $f_{\mathrm{R}}(t)$ is not to be confused with the notation of a probability density function $f_X(x)$ used in the remaining parts of this thesis.

Combining both effects a) and b) results in the frequency shift between the instantaneous transmit and receive signal frequencies $\Delta f = f_{\mathrm{T}}(t) - f_{\mathrm{R}}(t)$:

$$\Delta f = \frac{df_{\mathrm{T}}}{dt} \cdot t_{\mathrm{ToF}} - \Delta f_{\mathrm{Doppler}} = \frac{df_{\mathrm{T}}}{dt} \cdot \frac{2 \cdot R}{c_0} + \frac{2 \cdot v_{\mathrm{rel}} \cdot f_0}{c_0} \tag{2.4}$$

which is a line in the R-v_{rel} space. In combination with a second frequency ramp, typically with negative df_{T}/dt, two lines are obtained in R-v_{rel} space. The intersection of the two lines determines the distance R and the relative velocity v_{rel} to an object. Potential ambiguities due to the detection of multiple objects are resolved with additional frequency ramps of different df_{T}/dt. In practice, Δf is evaluated by mixing[10] transmit and receive signals and transforming the mixed signal into the frequency domain with a Fourier transformation.

Angular Discrimination

A variety of methods are employed to obtain measurements in different directions in space. Angular discrimination in the far field is easier to achieve in the azimuth[11], as automotive sensor size limitations restrict the discrimination in elevation[12]. A simple solution for the angular discrimination is mechanical scanning by rotating the radar's antenna. Another option is the use of multibeam antennas, which evaluate an object's angular direction by exploiting known antenna characteristics[13]. Planar antenna arrays with known spatial separation of the individual antennas use the dependence of the received signal phases at each antenna on signal direction for angular discrimination. Further used are the Monopulse method and Dual-Sensor concepts.

The distance, velocity and angular measurements of a radar are aggregated cyclically and constitute the sensor's raw data. The raw data is clustered and is the input to tracking algorithms, which associate the raw data to relevant objects in the environment and filter their state estimates (position, velocity, acceleration, etc.).

Strengths, Limitations and Perception Errors

The main advantages of radars compared to other sensors are their direct relative velocity measurement capability and their comparatively strong robustness towards weather influences. An important disadvantage is their limited angular discretization due to restrictions on acceptable sensor sizes. Object classification is another limitation of radars because, amongst others, the high

[10] Mixing is the process of signal multiplication in high-frequency technology. [76]
[11] Azimuth is the angle of a direction in the horizontal sensor plane w.r.t. the sensor's orientation.
[12] Elevation is the angle of a direction in the vertical sensor plane w.r.t. the sensor's orientation.
[13] The antenna characteristic describes the relative strength of the electromagnetic field emitted by an antenna in dependence of the angular direction. [76, 82]

variability of the radar cross section within one object makes object classification based on the radar cross section difficult. Automotive long range radars have a range of up to 250 m.

Despite their robustness towards weather influences, strong rainfall with drop sizes in the magnitude of the radar's wavelength can lead to strong atmospheric attenuation and larger signal to noise ratio. Water on the radome leads to a refraction of the beam, potentially causing errors in angular measurements [76, 83]. Spray water next to vehicles could cause false positive detections. An important effect is the multipath propagation of a radar beam, for instance due to reflections on the road surface [40, 59]. Multipath propagation can lead to signal interference, detection errors and angular measurement errors. As apparent in Eq. (2.1), the radar cross section and with it the properties of the objects are an important influencing factor for a radar's detection performance [40]. These exemplary factors and influences can cause perception errors.

2.1.2 Camera

Cameras are well suited for object detection, object classification and angular discrimination (e.g. estimating the orientation or contour of an object). In case of stereo camera systems, they also provide distance measurements. In the automotive domain, cameras were first implemented as rear view cameras to assist drivers. Later, a more advanced camera use case was automatic high beam control. Currently, cameras are enabling a variety of ADAS by providing, amongst others, object, traffic sign, lane and free space detection. [84]

Measurement Principle

A 3d scene is optically projected in a camera through a lens on a 2d image sensor, which is discretized into pixels [84, 85]. Photons hitting the pixels generate an electrical charge proportional to the light intensity [84]. With the generated electrical charge, the 2d image is digitized [84]. Through the projection of a 3d scene on a 2d image sensor, information about one dimension is lost [85]. Angular information is contained in the 2d image.

Predominating are complementary metal-oxide-semiconductor (CMOS) image sensors [84]. CMOS image sensors transform and digitize the electrical charge at the individual pixel level [84]. Unlike radars, lidars and ultrasonic sensors, which all actively emit a signal, cameras are passive sensors making use of available electromagnetic radiation in the environment. Most commonly, the image sensors are sensitive towards the visible light spectrum [84].

Often no color filter is implemented in the image sensors due to the involved cost, i.e. the sensors are monochromatic and provide a greyscale image [84–86]. An alternative are infrared cameras, which are deployed as active sensors with an infrared headlight, or alternatively as passive sensors generating a thermal image [84, 86]. Due to their comparatively large cost, infrared cameras are

not implemented frequently [84, 87]. The raw output of an automotive camera is a greyscale, infrared, or alternatively, a rgb color[14] image with frame rates in the order of 30 Hz [84, 85].

Distance Measurements

It is possible to derive distance information using a monocamera, e.g. with known camera height above the ground and by assuming to drive over a plane [85]. The associated distance measurement accuracy is however unacceptably large, preventing safety-critical applications such as Automatic Emergency Braking (AEB) [84–86]. To obtain distance measurements with acceptable accuracy, stereo camera systems are used instead [85, 86]. Stereopsis[15] is based on two images of the same scene, recorded simultaneously by cameras from different positions with known calibration parameters [85]. In an axis-parallel stereo system, the depth R[16] of a pixel is reconstructed from the known focal length f_{length} and known distance between the two cameras (base width) b [84–86]:

$$R = \frac{f_{\text{length}} \cdot b}{d} \tag{2.5}$$

where d is the disparity of a specific corresponding pixel in the images of two cameras, defined as the displacement in pixels in the image coordinate systems [84, 85]. f_{length} is the distance between the image sensor and the camera's optical center[17] [85]. In the general case of stereo cameras not being axis-parallel, a transformation (rectification) is applied to virtually align the cameras' coordinate systems [85]. To evaluate the disparity d, the correspondence between pixels in the two images has to be found, for which various methods have been developed [85]. Ultimately, a stereo camera system provides for each relevant pixel a 3d coordinate together with the associated greyscale or rgb information [86]. The tracking of corresponding pixels in subsequent images additionally enables to estimate the motion vector (velocity, acceleration) for each pixel [86].

Object Detection and Classification

Object detection and classification is based on temporally stable groups of features. "Features are locally constraint[sic], expressive parts of an image which allow for a symbolic or empiric description of the image properties or the object." [85]. Features are for instance edges and corners, which have the property of a considerable variation in grayscale or color. Hence, edges and corners

[14] rgb color images have three channels: red, green and blue. For each channel, the pixels take values between 0 and 255 (in case of 8-bit encoding) to represent the light intensities in the respective wavelength-spectra. A grayscale image only has a single channel, which takes values between 0 and 255 (8-bit), representing the light intensity.
[15] Stereopsis is also the basis for depth perception by humans. See e.g. https://en.wikipedia.org/wiki/Stereopsis.
[16] To be consistent with the notation for the distance in this thesis, the depth is denoted with R for range.
[17] The optical center lies in the aperture, i.e. in a camera pinhole model the optical center is the pinhole [85].

can be identified through e.g. the grayscale gradient. Object detection might additionally be enhanced with the disparity and the motion vector of the pixels. [85]

Camera based object detection and classification is mostly approached with Machine Learning, i.e. a classifier is trained with a labeled set of images with known object classes. Widely used was the Viola–Jones object detection framework [88], e.g. for vehicles [89] and pedestrians [90]. The algorithm uses a cascade of Ada-Boost classifiers [91] to detect objects on the basis of Haar-like features[18] [88].

Currently, object detection and classification is approached with Convolutional Neural Networks (CNN), a class of deep learning algorithms[19] [93, 95]. Relevant features for object detection and classification are learned by the algorithm from labeled images within the convolutional layers of the CNN [93, 95]. Once trained, feature patterns present in unseen images activate the CNN's neurons. The activation in the final layer of neurons in a CNN is typically passed through a softmax function to obtain a probability estimate of object existence, or of the object classes, respectively [93, 95]. A detection is indicated by the camera if the object existence probability exceeds a detection threshold (and if additional criteria are met, e.g. the object detection is validated by subsequent images). The detected objects are then in a final processing step tracked, for instance with a Kalman filter [85].

An exemplary CNN architecture for real-time applications in embedded systems is the single shot detector (SSD) [96]. Other network architectures such as the Faster R-CNN [97] or the R-FCN [98] have larger computational cost compared to the SSD, which is an important factor for deployment in ADSs[20].

Strengths, Limitations and Perception Errors

Compared with other environment perceiving sensors, cameras have the advantage of providing the most extensive content of information and they are in analogy to human vision [84–86]. A disadvantage is their sensitivity towards environmental conditions (e.g. ambient light, weather, spray water, etc.) [40, 59, 84, 86]. Physical limitations of a camera are related to its resolution and

[18] A Haar-like features is essentially the difference in summed pixel intensities over adjacent rectangular regions in an image [88].

[19] Deep learning refers to training a deep neural network, which has millions of model parameters, and goes back to a seminal paper by Geoffrey Hinton et al. in 2006 [92]. Prior to [92], training a deep neural network "was widely considered impossible" [93]. Deep Convolutional Neural Networks are the state of the art in computer vision since Alex Krizhevsky et al. [94] won the ImageNet Large-Scale Visual Recognition Challenge in 2012 with their AlexNet. See also: https://beamandrew.github.io/deeplearning/2017/02/23/deep_learning_101_part1.html for a history on deep learning.

[20] Methods on Deep Learning develop fast and are quickly outdated. The state of the art performance in object recognition is typically evaluated with a benchmark suite, i.e. a labeled data set of images which allows to compare different algorithms. For instance, the KITTI Vision Benchmark Suite [99] for computer vision in ADSs is used to assess the performance of different algorithms and hence provides information on the state of the art.

optics [84]. For instance, to discriminate objects, a minimum number of pixels have to provide information about an object [84]. Another limitation is that the distance measurement accuracy of stereo cameras drops with R^2, leading to errors in the order of multiple meters for $R > 50$ m [84–86]. Also, misalignment of a stereo camera system leads to errors [86].

Important factors with influence on a camera's performance are ambient light conditions e.g. insufficient light during the night, large background radiation, scattered light, a low sun [40, 59, 84, 86], and weather influences such as strong rainfall, spray water, snowfall or fog [86]. Imperfections in the optics lead to unwanted reflections and light scattering, causing reduced performance [100] as cited in [84]. Another error source are projection errors due to deficiencies of the optics [84].

Of special importance for the performance of a camera are also the learned features (detection patterns) in the trained CNN themselves [40, 59]. Depending, amongst others, on whether the activation in the final layer of the CNN exceeds a threshold, an object is detected or not. Wrongly not exceeding the detection threshold causes false negative errors, e.g. because relevant features of an object have not been learned from the training set. Additionally, patterns in an image that lead to a high activation but do not contain a relevant object cause false positive errors. Ultimately, a slight variation in the orientation, location and contrast of an object in a camera image could lead to a variation in the activation that makes the difference between object detection or not, see also [40].

2.1.3 Lidar

Lidar (light detection and ranging) sensors complement radar and camera sensors [101]. Their main tasks are object detection and distance measurements [102]. With limitations, lidars are also used for object classification [102]. At present, lidars are a relatively novel alternative in the automotive series production, but they have been widely utilized in research projects, e.g. in [12, 13]. The following presentation of lidar technology is in reference to [102] if not otherwise indicated.

Similar to radars, lidars are active sensors but use an optical measurement principle. A lidar's laser diode emits electromagnetic radiation in the ultra violet, the visible, or the infrared spectrum. Common in automotive applications are wavelengths between 850 – 1000 nm (infrared spectrum).

Distance and Relative Velocity Measurements

The Time-of-Flight (ToF) principle for distance measurements is most widely used in automotive lidars. The ToF principle is illustrated schematically in Figure 2.3. As sketched in Figure 2.3, at time t_0 a laser pulse is emitted. The laser pulse propagates through the atmosphere, hits an object

and is scattered back to the laser source at time t_1. The backscattered pulse is detected by the sensor at time t_2. From the ToF $t_{\mathrm{ToF}} = t_2 - t_1$ and the known light velocity c_0, the distance R is derived with Eq. (2.2) [102].

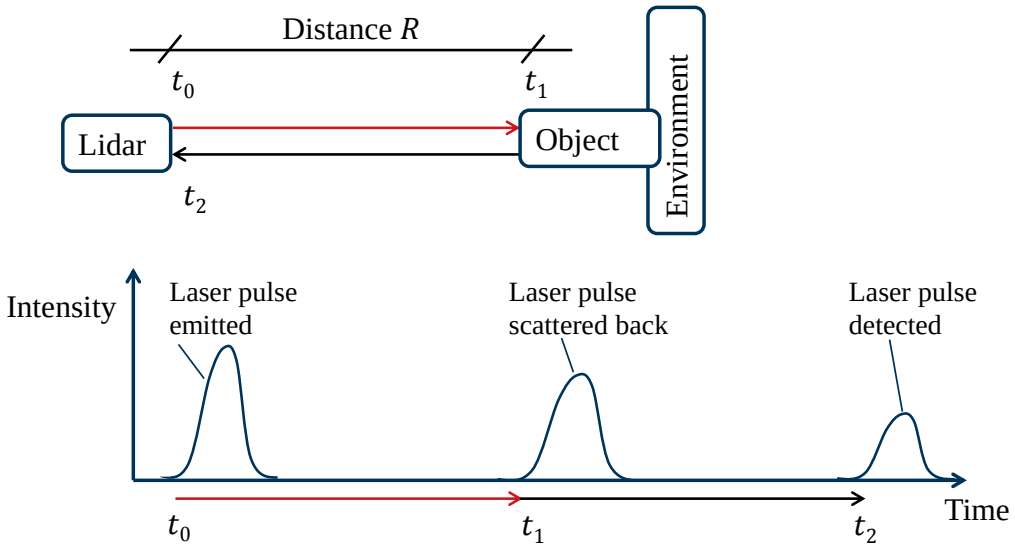

Figure 2.3 Sketch of the Time-of-Flight (ToF) measurement principle.

With a lidar, relative velocity measurements are obtained indirectly by differentiating subsequent range measurements over time. Measuring the Doppler frequency shift to derive relative velocity information is too expensive in the infrared spectrum for automotive applications.

Object Detection

To detect the backscattered laser pulse, its intensity at the sensor's receiver has to exceed a detection threshold [40]. The receiver is typically either a PIN-diode (positive intrinsic negative) or an APD (avalanche photo diode). The power P_{R} at the receiver is described with the lidar equation, which is in analogy to Eq. (2.1) [103]:

$$P_{\mathrm{R}} = \frac{T^2 \cdot \rho \cdot \eta_{\mathrm{T}} \cdot \eta_{\mathrm{R}} \cdot D_{\mathrm{lens}}^2 \cdot P_{\mathrm{T}}}{16 \cdot R^2} \tag{2.6}$$

where T is the atmospheric transmission factor, ρ the reflectance of an object. η_{T} and η_{R} are the optical efficiencies of the transmitter and receiver, respectively. D_{lens} is the diameter of the receiving lens, P_{T} the transmitted power and R the distance to an object. Eq. (2.6) assumes that the beam cross section is completely overlapping with an objects cross section.

While propagating through the atmosphere, part of a laser pulse's energy is diffusely scattered and absorbed by particles. Only the remaining energy of the laser pulse is transmitted. Together, the scattering and absorption constitute the atmospheric attenuation. The atmospheric transmission T factor accounts for attenuation, it describes the part of the energy that is transmitted. [102, 104]

Angular Discrimination

Above, the measurement principle is described in terms of a single laser beam, resulting in a single distance measurement. Scanning mechanisms that deflect the laser beam are used to obtain distance measurements in different directions in space. Alternatively, multibeam sensors allow for an angular discretization of the measurements.

The distance measurements in the different spatial directions are aggregated and cyclically repeated in measurement cycles. The resulting lidar raw data is for each measurement cycle a 3d point cloud of the driving environment. Based on this data, algorithms detect and track relevant objects such as traffic participants (see Sections 2.1.5 and 2.2.2).

Strengths, Limitations and Perception Errors

The advantage of lidar sensors is their high distance measurement accuracy [60]. To a large degree, the uncertainty in the distance measurements is independent from the actual distance to an object [60]. Compared to a camera, the disadvantage of a lidar is that the object's contours cannot be identified as accurately due the lidar's limited angular resolution [60]. Further, little information about an object's texture is obtained [60]. Compared to a radar, the disadvantage of lidars is their high sensitivity to environmental influences such as snowfall. The main physical limitations of lidars are that they potentially cannot detect objects with low reflectivity at the respective wavelength (i.e. high absorption or high transmission) and total reflection (i.e. the laser pulse is not scattered back to the sensor but deflected in another direction) [102]. The (current) maximum range of an automotive lidar is around 150 m [102].

Based on the described physical measurement principle and based on Eq. (2.6), examples for factors influencing the performance of a lidar are: background illumination (e.g. low sun, scattered light) [102], particles in the atmosphere (e.g. rainfall, fog, snowfall, spray water, dust) [40, 59, 102, 104, 105], properties of the target (surface material and color [106], relative position and orientation [59]) and potential dirt, snow or ice on the sensor cover [101, 102]. These factors must be considered when evaluating the reliability of lidar perception.

2.1.4 Summarizing Sensor Strengths and Weaknesses

The strengths and weaknesses of different sensor technologies are qualitatively summarized in Table 2.1. For a robust[21] environment perception, the combination of different sensing technologies is necessary because the advantages of one help to mitigate the disadvantages of

[21] Terms like robust, safe and reliable are often used interchangeably. Even though these concepts are related, they are technically not identical. As a distinction, a clear definition of robust is given: "Robustness: strength, or the ability of elements, systems, and other units of analysis to withstand a given level of stress or demand without suffering degradation or loss of function" [107].

another [58]. Perception is therefore provided by multiple sensors to achieve sufficient *perception reliability*.

Table 2.1 Qualitative comparison of automotive environment sensors. (++: strong advantage / highest performance, + advantage / high performance; 0: neutral / medium performance, -: disadvantage / poor performance; -- strong disadvantage / worst performance). Partly based on [108].

	Radar	Camera	Lidar
Field of View	+	+	++
Range	++	+	++
Velocity measurements	++	0	+
Angular resolution	-	++	+
Adverse weather	0	-	-
Ambient light	0	--	-
Object classification	-	++	+

2.1.5 Generic Sensor Data Processing

Based on the review of the sensing technologies, a generic environment sensor model is presented in Figure 2.4. With this model, the data processing of different environment perceiving sensors is generalized to describe how an ADS represents its environment. Despite their different physical measurement principles, on an abstract level, each of the sensing technologies can be modeled with Figure 2.4 [40, 109].

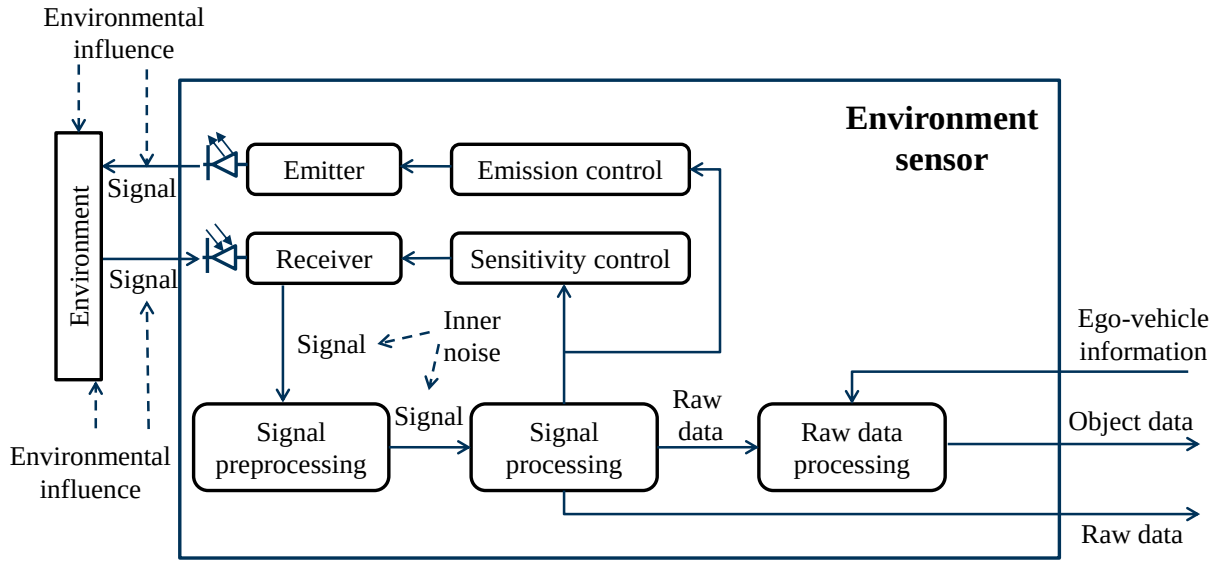

Figure 2.4 Generic environment sensor model and signal processing. Based on a combination of Abbildung 2.1 in [109] and Bild 8 in [40].

Active sensors such as radars and lidars emit a signal, which is reflected back to the sensor by objects in the environment. The receiver captures the reflected signal. A camera deviates slightly from Figure 2.4 because it does not actively emit a signal but uses a signal already present in the environment [40].

As discussed in Section 2.1, environmental influences act as external performance influencing factors, or in other words, as external noise [40, 58]. The received signal is preprocessed, including e.g. filtering, amplification and digitization [58, 76, 102]. In these processing steps, inner noise sources [40] apply. Examples include dark current noise [84], photon shot noise [84], phase noise of oscillators [76] and thermal noise [40].

From the preprocessed signal, the raw data is derived [58]. In case of radars and lidars, raw data evaluation is triggered if the signal exceeds a detection threshold [40]. The different sensing technologies provide different types of raw data: a radar for instance provides distance, relative velocity and angular measurements within a fixed discretization. A lidar provides a 3d point cloud of the environment and a camera an image, i.e. light intensities discretized in pixels.

Typically, in the next processing step, sensor specific algorithms[22] detect objects from raw data with help of features. Features for object detection include properties of the underlying signals such as frequencies, phases and intensities as well as the raw data and information extracted from the raw data such as gradients, feature maps (CNN), stereo images and signal spectra. Clustering is used to generate object hypotheses from regions with similar features. With the input features of an object hypothesis, the object detection is a binary classification problem. This classification is either approached with decision rules set up by experts or by learning a discriminating decision boundary in the feature space from labeled data with statistical methods or machine learning. For instance, in a CNN, the discriminating decision boundary is implicit in the trained weights of its neurons (see Section 2.1.2). [59, 110]

The detected objects of individual sensors are usually described in terms of an object based vehicle environment model [58], which is outlined next. Potentially both, the raw and the object data, are transferred to the vehicle bus and serve as an input to the sensor data fusion.

[22] In practice, it is often not possible to obtain detailed descriptions of the implemented algorithms, as these are proprietary.

2.2 Environment Representation

By combining information provided by multiple sensors, a fused vehicle environment model is generated with sensor data fusion [58, 67]. The content of the environment model (together with digital maps or any other offline information) is ultimately the basis for the decision making of an ADS [31, 51, 60, 67] and is therefore central for system safety. To assess *perception reliability*, it is crucial how a vehicle's surrounding is represented and how this representation is set up with the information provided by individual sensors. Therefore, Section 2.2.1 describes how an ADS represents its surroundings in an environment model and Section 2.2.2 outlines how the information of different sensors is combined in a sensor data fusion to generate a fused environment model.

2.2.1 The Vehicle Environment Model: How an ADS Sees the World

A vehicle environment model is an abstract representation of a vehicle's surroundings. It ideally contains accurate[23] state and semantic information about all relevant static and dynamic objects in the environment over time and space [60, 67]. Relevant static objects are for instance road boundaries, obstacles (drivable and non-drivable), infrastructure, lane markings, traffic signs and traffic lights. Exemplary dynamic objects are cars, trucks, pedestrians, animals, bicycles and motorcycles [60]. Which types of objects and information are relevant depends on the specific automated driving functionality [60].

Two environment modeling approaches are distinguished: an object based and a grid based environment representation [60, 67]. In an object based approach, each relevant dynamic and static object is associated with a dynamic time-discrete state space model [60, 67]. An object's dimension is typically represented by a 2d or 3d bounding box (i.e. a rectangle or a cuboid)[24] in the object based modeling approach [60, 67, 106]. It is common to project all objects into a plane [67, 106], that is, the bounding boxes often live in a 2d space. The dynamic object models are time-discrete because the sensing data is provided at discrete points in time [67]. Exemplarily, Table 2.2 lists common quantities to describe the objects' states[25]. The required and used state quantities also depend on the implemented dynamics model [67]. In practice, the objects' state estimates are at each discrete point in time stored in object lists. This modeling approach is well suited for dynamic objects, as it allows to efficiently filter state estimates.

[23] It is here on purpose not defined what accurate exactly means, as this definition itself is a challenge.
[24] Representing the objects with 2d or 3d bounding boxes is state of the art. In the future, it is likely that more detailed object representations will prevail, see e.g. [111].
[25] A stationary quantity (e.g. the dimensions of a vehicle) is commonly referred to as parameter and a temporally variable quantity (e.g. a vehicle's velocity) is termed state [59]. Simplifying, this distinction is not made and both stationary and temporally variable quantities of an object are referred to as states.

Table 2.2 Exemplary object state variables in a vehicle environment model. See e.g. [60].

Description	Object State
Position	2d coordinates of bounding box reference point
Dimension	Width, length, (height) of bounding box
Velocity	2d velocity vector
Acceleration	2d acceleration vector
Orientation	Yaw angle
Rotation	Yaw rate

A grid based environment representation discretizes a vehicle's environment into spatially fixed 2d or 3d grid-cells. A cell is marked as either occupied or not occupied, conditional on the distance measurements provided by environment sensors, e.g. through Bayesian inference. Methods to deal with moving objects in the grid based approach have been developed. Due to remaining difficulties of representing moving objects in grid cells, an occupancy grid map is best suited for the static environment. An important output of occupancy grid maps is the drivable free space. [60, 67]

It is possible to combine a grid based environment representation for static objects with an object based representation of dynamic objects to exploit the advantages of both modeling approaches [67]. This thesis concentrates on the object based vehicle environment representation because it is the dominating modeling approach in ADSs, often even for static objects.

2.2.2 Sensor Data Fusion

"Data fusion is the process by which data from a multitude of sensors is used to yield an optimal estimate of a specified state vector pertaining to the observed system." [112]. The idea behind sensor data fusion is to combine the strengths of different sensors, or more generally, of different sources of information in order to increase the accuracy of state estimates and in order to reduce the influence of sensing errors [58, 110]. Sensor data fusion is here described as a basis for its conceptualization in a *perception reliability* assessment.

In context of sensor data fusion, it is helpful to distinguish between complementary and redundant sensors. Complementary sensors provide different types of information (e.g. a camera provides object classification and a radar distance measurements [58]), whereas redundant sensors provide the same type of information within an overlapping field of view (FOV) (e.g. two sensors obtain distance measurements for the same object) [58]. Sensors with non-overlapping FOV are complementary [58] and not redundant. Complementary sensors allow to capture all relevant aspects of an ADS's environment, while redundant sensors typically increase *perception reliability*. For instance, by fusing multiple redundant sensors in contrast to using only a single sensor, the position estimate of an object in the environment is improved and the probability of not detecting an object at all is decreased [58].

It is distinguished between a decentralized, raw data or feature based sensor data fusion and a centralized, object data based sensor data fusion. [58, 110]. In the latter, object detection is performed by individual sensors as illustrated in Figure 2.4. Detected objects of individual sensors are the input to sensor data fusion. The centralized fusion architecture is predominantly used in practice because of reduced requirements on bandwidth for data transmission and because of its comparative simplicity.

Object Tracking

The main task of a centralized sensor data fusion in ADSs is object tracking[26], which is the process of associating object detections in different sensors to object tracks[27] in the environment model and of filtering the objects' state estimates [58, 110].

The Bayes filter is ubiquitous for filtering object states, and hence is central to sensor data fusion. Bayes filtering is a recursive implementation of Bayes rule [114] for inference in hidden Markov Models (HMMs) [115]. Figure 2.5 illustrates an HMM for Bayes filtering in terms of a Bayesian network[28] [115, 116]. $\mathbf{x}_m$ is an uncertain hidden true state of an object at a discrete point in time m and is populated e.g. with the variables in Table 2.2. In reality, one does not observe $\mathbf{x}_m$ but only the corresponding noisy sensor observation $\mathbf{z}_m$. It is assumed that sensor observations at different time steps are conditionally independent given the hidden states $\mathbf{x}_m$ [115].

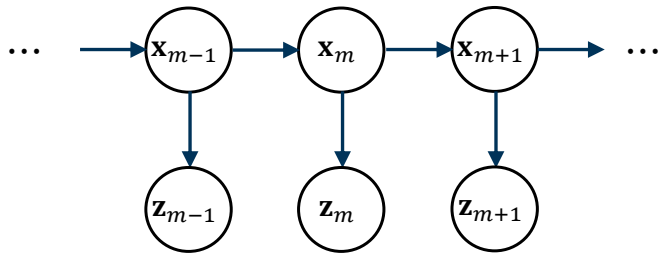

Figure 2.5 Hidden Markov Model (HMM) underlying the Bayes filter [115, 116]. $\mathbf{x}_m$ is the hidden true state of an object at discrete point in time m and $\mathbf{z}_m$ is the corresponding noisy sensor observation.

The Bayes filter proceeds at each discrete point in time m with two steps: the prediction of the states in the next time step and the update of the states with a new sensor observation. The mathematical details of the Bayes filter are well-known and not repeated here, instead we refer to [57, 59].

[26] The state of the art in environment perception, object detection and sensor data fusion is developing. In the future, it might be that sensor data fusion additionally detects objects based on raw data of multiple sensors (raw data fusion), see e.g. [113].
[27] An object track is essentially the state history of a given object.
[28] A Bayesian network represents a joint probability model with a directed acyclic graph by exploiting conditional independence assumptions [115, 116].

The Kalman filter, a specific implementation of the Bayes filter assuming a linear dynamics model and assuming that the observation and process noise follow a normal distribution, is a common choice in practice [57, 67]. With the assumptions underlying the Kalman filter, the solution to the Bayes filter is traceable in analytic form, rendering a cheap implementation [57, 67, 115]. The most common alternatives with less restrictive assumptions but larger computational cost are the unscented Kalman filter, the extended Kalman filter and the particle filter [67, 72].

With a Bayes filter, the estimated state of a tracked object is updated whenever a sensor provides new observations that are associated to the respective track. The incorporation of information from multiple sensors provided at different points in time is based on a sequential [110] application of the Bayes filter for each respective sensor. An extension of the Bayes filter to account for multiple sensors providing information at the same point in time is straightforward with its formulation in the joint space of all sensors [59].

An important task of sensor data fusion and a prerequisite for state filtering is the association of sensor object detections to different object tracks. Based on the model in Figure 2.5, the posterior predictive distribution $f(\tilde{\mathbf{z}}_m|\mathbf{z}_{1:m-1})$ of the one step ahead future sensor observation $\tilde{\mathbf{z}}_m$ given the history of measurements $\mathbf{z}_{1:m-1}$ is derived at prediction time. $f(\tilde{\mathbf{z}}_m|\mathbf{z}_{1:m-1})$ allows to judge whether a certain observation is likely due to a specific track and hence, enables object association. For instance, a validation gate [58, 117] is specified for each track with $f(\tilde{\mathbf{z}}_m|\mathbf{z}_{1:m-1})$, e.g. defined by the 99.9 % probability region of $\tilde{\mathbf{z}}_m$ in the state parameters' hyperspace. Sensor observations that fall within such a validation gate are associated to a track [58, 102]. In case of ambiguities, i.e. if a sensor observation can be associated to multiple tracks, the ambiguity is dissolved by associating the sensor observation $\mathbf{z}_m$ to the track that maximizes its probability conditional on the track being the correct one.

In case of a Kalman filter, the described association procedure is identical to a Maximum Likelihood nearest centroid /nearest neighbor classifier with the Mahalanobis distance as the distance metric [67, 115, 117]. More generally, the procedure is a Bayesian model selection with a uniform prior on different tracks [114]. A problem is however that multiple observations can be associated to one track, violating the common assumption of perfect object discrimination under which each track only generates a single observation per sensor at one time step [59, 67]. To account for this constraint, the problem is extended to express the likelihood of joint association hypotheses under the one observation to one track constraint [59, 67]. This results in a global nearest-neighbor association algorithm, in which the likelihood of all sensor observations at one time step is maximized jointly and not individually [59, 67].

Sensor observations that cannot be associated (e.g. they do not fall within a validation gate) are candidates for new tracks [59]. If a new candidate track fulfills certain heuristics such as a successful association to observations for a certain number of time steps and complies with an upper bound on state estimation uncertainties, a new track is born [59, 102, 106]. Likewise, a track dies if no sensor observations are associated to it for several time steps, or if other heuristics apply, e.g. the state estimation uncertainty exceeds an upper bound [59, 102]. Hence, object existence uncertainty on the fusion level is commonly considered by means of heuristics such as the described multi cycle validation [59], which reduces false positive detection rates.

The disadvantage of associating observations to tracks with a nearest-neighbor approach is that the decision is deterministic [67, 117]. The Joint Integrated Probabilistic Data Association (JIPDA) is an alternative probabilistic association /fusion algorithm [67, 118]. It takes association uncertainties into account by updating each track with a weighted combination of all sensor observations, where the weights depend on the association probabilities of a specific track to the different sensor observations [67, 117, 118]. Due to computational cost and unknown sensor performance parameters necessary to tune the algorithm, JIPDA is not widely implemented in practice.

Each tracked object is finally classified (e.g. car, pedestrian, truck, bicycle, etc.) based on features gained throughout the data processing [58]. This classification problem is solved with a suitable classifier, e.g. trained with supervised learning [110, 115, 119].

Grid Based Fusion

The described tracking of objects is in the context of the object based environment representation. In the grid based environment representation, the fusion of different sensor observations is a straightforward application of Bayes rule to obtain posterior grid occupancy probabilities based on the likelihood of an occupied grid cell, given sensor range measurements [67].

Errors in Sensor Data Fusion

The main source of perception errors due to sensor data fusion are errors in the association of observations to tracks. The consequences include wrong object state estimates (i.e. track deviations) potentially leading to the loss of a track [117], track multiplication [59] or track coalescence [59]. The latter describes a false positive track, which is initiated due to sensors indicating an object when none is present, being associated with the observations arising from an object existing in reality. This might cause false positive and true positive tracks swapping roles, in some cases in an alternating pattern. A heuristic to prevent track coalescence is to delete one of the two tracks if they are too close to each other to represent two real objects [59].

Aside of wrong data association, e.g. caused by the association algorithm, a wrong association could also arise due to the properties of underlying sensing technologies. For instance, a lidar might detect the rear view reflectors of a truck, while a radar might detect the truck's axle [58]. In combination, the fusion might not be able to associate these detections leading to potential perception errors. Another error source is the misalignment of different sensors, which on its own might lead to negligible deviations in individual sensors but could lead to association errors in the fusion [58]. The tracking of correctly associated objects with the Bayes filter is not deemed to be a significant source of perception errors, as long as underlying model assumptions are adequate.

3 The Safety of Automated Driving Systems

An environment sensor is inherently imperfect due to external influences, inner noise, as well as assumptions and potential inadequacies in data processing [40, 58], as follows from Chapter 2. Additional to and in combination with perception errors of individual sensors, sensor data fusion causes perception errors, for instance due to association errors [59]. Because perception errors can cause inadequate ADS behavior and in the worst case can lead to accidents, *perception reliability* has to be validated.

We therefore start this Chapter in Section 3.1 by formalizing the task of validating *perception reliability* in the context of ADS safety. With this formalization, it is in Section 3.2 evaluated if established automotive safety concepts are transferable to develop a reliable environment perception, and if these concepts achieve sufficient safety for an ADS with *higher levels of driving automation*. Section 3.3 additionally discusses if established automotive testing and safety validation procedures allow to demonstrate sufficient *perception reliability*.

3.1 Demonstrating Environment Perception Reliability

The central purpose behind a *perception reliability* demonstration is to validate ADS safety. Safety is typically a legal prerequisite for the market introduction of products [38, 120][29]. To discuss the task of demonstrating *perception reliability*, safety is defined in reference to ISO 26262[30] as "the absence of unreasonable risk" [62]. Unreasonable risk itself is defined as the "risk judged to be unacceptable in a certain context according to valid societal moral concepts" [62]. Hence, the goal of a safety validation according to this definition is to demonstrate that the ADS complies with an acceptable risk [54].

In light of these definitions, the question is, what constitutes an acceptable risk for ADSs? According to [31, 43] this question remains unsolved.

Defining risk and reliability targets for ADSs and their components is therefore a challenge itself. To put the task of validating ADS safety and a demonstration of its *perception reliability* into perspective, and as a foundation for developing *perception reliability* targets, we next present how

[29] The German Product Safety Act for instance states that a product "[…] may only be made available on the market if it […] does not put at risk the safety and health of persons or other legal goods […]" [120].
[30] A similar definition of safety is given in DIN 31000: "Safety is a situation in which the risk does not exceed the acceptable risk". ([121] authors translation, original: "Sicherheit ist eine Sachlage, bei der das Risiko nicht größer als das Grenzrisiko ist.")

to derive rational risk criteria for ADSs that are acceptable for society and OEMs. In other words: how safe is safe enough?

3.1.1 Risk Acceptance for Automated Driving Systems

This Section is partly based on and taken from our publication in [39].

Common ways of expressing the risk due to a system are in terms of the expected value of negative consequences per calendar time interval or per unit time of exposure [122]. The latter has the advantage of taking the actual exposure into account. Risk is measured as[31] [123, 124]:

$$\text{Risk} = \sum_{j} \lambda_{\text{F}_j} \cdot c_{\text{F}_j} \tag{3.1}$$

where λ_{F_j} is the rate of system failure events F_j (i.e. accidents) of type j and c_{F_j} are the associated consequences. The consequences are for instance monetary, injuries or the loss of life. λ_{F_j} has the unit $[1/\text{h}]$. An alternative is to express λ_{F_j} with unit $[1/\text{km}]$, as is common in road vehicle accidents statistics [33, 35, 41].

In an ideal world, the risk of an ADS is zero. However, an absolutely risk free technical transportation system is probably impossible to build. Therefore, risk acceptance criteria are required. Three fundamental ethical principles for defining risk acceptance criteria are distinguished [122, 125, 126] to discuss the acceptable risk "[…] according to valid societal moral concepts" [62]:

- Equity: A risk is acceptable if it is below an upper limit, which should not be exceeded for any member of society.
- Utility: Risk acceptability is derived based on societal cost and benefits of a technical system.
- Technology: A risk is acceptable if it is as low as that of a reference system.

The minimum endogenous mortality[32] (MEM) criterion derived with the equity principle and used in the railway standard EN 50126 [125, 127, 128] is an example of a risk acceptance criteria. The MEM criterion imposes that the risk of dying due to a technical system must not significantly increase the risk of dying in comparison with natural causes such as diseases [125]. In western countries, children between 5 and 15 years have the MEM rate per person [125], which is around

[31] The definition of risk goes back to Blaise Pascal (1623 – 1662): „Risk should be proportional to the probability of occurrence as well as to the extent of damage."
32 "The endogenous mortality rate is the rate of deaths due to internal causes of a given population at a given time." [125].

$2 \cdot 10^{-4}\ \mathrm{yr}^{-1}$ [127]. An acceptable risk for the most exposed persons of $\approx 10^{-5}\ \mathrm{yr}^{-1}$ is derived by imposing that a technical system is only allowed to increase the MEM rate by 5% [125]. Considering a person who drives in the order of 1000 h per year, the MEM criterion would lead to an acceptable fatality risk of $10^{-8}\ \mathrm{h}^{-1}$.

With the utility principle, the acceptable risk is defined through expert judgment, analyzing risks and corresponding benefits that have been accepted by society in the past (revealed preferences [129]) or formal cost-benefit and decision analyses[33] [125]. Note that one could also interpret revealed preferences to follow the technology principle.

Well-known is the "as low as reasonable possible" (ALARP) principle [125, 126], illustrated in Figure 3.1. It defines three risk regions of individual fatality risk for the most exposed persons. In the broadly acceptable region, the risk is negligible and can be accepted. The broadly acceptable region prescribes an individual fatality risk of the most exposed person of $\leq 10^{-6}\ \mathrm{yr}^{-1}$. In the tolerable region between $10^{-6}\ \mathrm{yr}^{-1}$ and $10^{-3}\ \mathrm{yr}^{-1}$, the risk has to be reduced ALARP, which depends on cost and benefits (utility principle) [125]. In the unacceptable region $\geq 10^{-3}\ \mathrm{yr}^{-1}$, the risk cannot be accepted (equity principle) [125]. Assuming an exposure of 1000 h driving per year, a broadly acceptable risk of $10^{-9}\ \mathrm{h}^{-1}$ is derived.

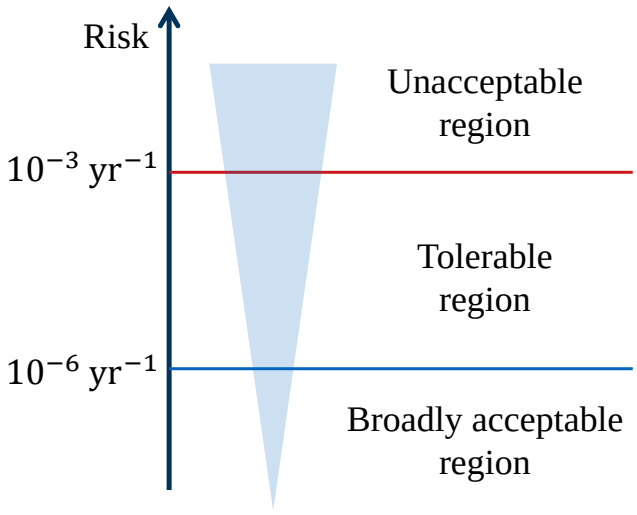

Figure 3.1 Risk acceptability according to the ALARP principle. In reference to Figure 4.3 in [125].

[33] Formal analyses require to express the risk of loss of life with monetary values, which is controversial and often raises ethical concerns. Nevertheless, to make rational decisions accounting for benefits and risks of a new system or policy, the value of a statistical life (VSL) is used in formal analyses [130]. VSL is defined as the amount of money the society is willing to pay for an infinitesimal reduction in the risk of loss of live [131]. Hence, VSL does not value life but is a measure for the accepted societal cost to prevent a statistical fatality. This quantity is implicit in (past) decisions made by society and can be assessed empirically. For instance, the US Environmental Protection Agency (EPA) quantifies $\mathrm{VSL} = 7.4 \cdot 10^6$ \$ (2006) and recommends this value updated to the year of the analysis as a basis for cost benefit analyses [132]. In [133], the EPA quantifies $\mathrm{VSL} = 10.3 \cdot 10^6$ \$ in terms of 2013\$.

The globalement au moins aussi bon (GAMAB: globally as good as existing systems) is an example for the technology principle [125]. The ethics commission on automated driving of the German Federal Ministry of Transport and Digital Infrastructure states [52]: “The licensing of automated systems is not justifiable unless it promises to produce at least a diminution in harm compared with human driving, in other words a positive balance of risks.” Hence, the ethics commission follows the technology principle with the human driver as a reference.

A similar argument that ADSs should be at least as good as human drivers in terms of accident rates is made in [35, 41, 44, 66]. Current reference values for accident rates of human drivers involving fatalities are $\approx 6.8 \cdot 10^{-9}\,\mathrm{km}^{-1}$ in the US (all road types) [41] or $\approx 1.5 \cdot 10^{-9}\,\mathrm{km}^{-1}$ (German highways) [134]. With the assumption of an average velocity of 100 km/h [135], this results in a risk of $\approx 6.8 \cdot 10^{-7}\,\mathrm{h}^{-1}$ and $\approx 1.5 \cdot 10^{-7}\,\mathrm{h}^{-1}$, respectively.

Due to the development in automotive safety systems, the reference risk of human controlled road traffic is however a non-stationary target [40]. Figure 3.2 for instance depicts the road vehicle risk in the US over time. The cited reference risk values of human driving should therefore not be interpreted as fixed.

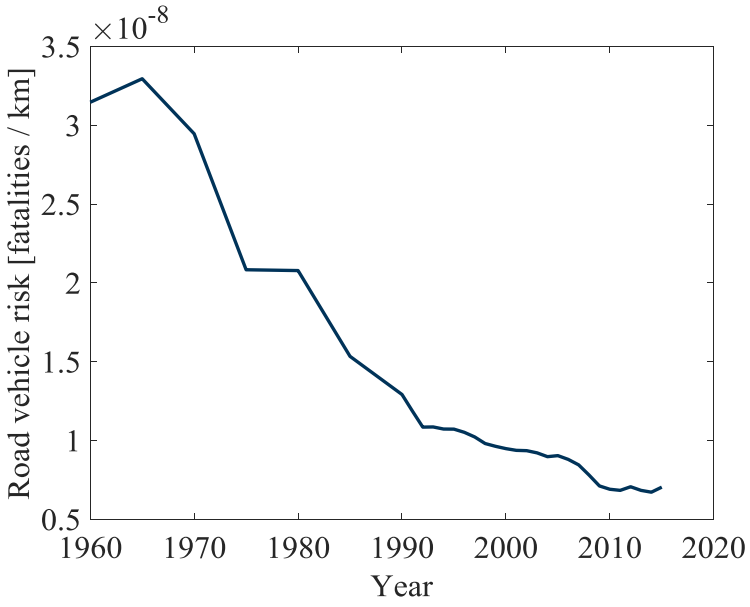

Figure 3.2 The risk of human driving in the US over time, in terms of fatalities per vehicle-km travelled. Data from [136].

Other reference systems are aviation with acceptable failure rates of $10^{-9}\,\mathrm{h}^{-1}$ (for catastrophic failures) [137] and automotive E/E systems with automotive safety integrity level D (ASIL D) having an acceptable (hardware) failure rate of $10^{-8}\,\mathrm{h}^{-1}$ [138].

Aside of ethical risk criteria, it is important for products such as ADSs to consider legal requirements [38] and the risk acceptance therein. For instance, according to German Product Liability Act (German: Produkthaftungsgesetz) [139], a producer is hold liable in case of damages caused by a failure of its products. The liability obligation is excluded, if the product is designed according to the [40, 43, 140] "[…] state of scientific and technical knowledge at the time when the producer put the product into circulation […]" [139].

The state of scientific and technical knowledge is not explicitly defined. Implicitly, the state of scientific and technical knowledge could however be interpreted to impose the technology risk acceptance criteria on a new product. The Product Liability Act also implies that, if under the state of scientific and technical knowledge, risks cannot be avoided, the decision to release the product should be based on assessing the risks and benefits associated with the product [140], i.e. the utility principle applies. Another possible interpretation is that one follows the state of scientific and technical knowledge, if the product is designed according to accepted standards and norms such as ISO 26262 [40, 54, 141]. These standards and norms in turn might be related to risk acceptance criteria, as for example the MEM criterion in the railway standard EN 50126 [127]. Therefore, ethical risk acceptance criteria are at least indirectly considered by German law.

An acceptable risk should be the basis for a formal ADS safety validation and compliance with the acceptable risk should be demonstrated [40]. While the given examples on risk acceptance criteria provide a guidance, a universally established risk acceptance criteria does not exist [125]. Society and its representatives, the authorities, must make a decision on acceptable risk, since ultimately, the risk of a technical system has to be accepted by the public.

It is further pointed out, ultimately, it is not the true but the *perceived* risk of a technical system that determines risk acceptance by society. Public risk perception is influenced by a number of factors and cognitive biases [122, 125, 129, 142–144]. Amongst others, public risk perception and acceptance is influenced by its context, the benefits of accepting a risk, the number of exposed members of society, familiarity, immediacy of effects, availability of occurrences, anchoring, personal control, voluntariness and potential for catastrophic consequences [122, 125, 129, 142–144]. Hence, even if an ADS complies with the discussed risk acceptance criteria, the *perceived* risk could prevent societal acceptance. These factors and biases therefore should be considered in the discussion of acceptable risk for ADSs because in case of fatal accidents, the particular OEM and the technology in general face serious reputational challenges.

3.1.2 The Approval Trap

How much testing is required to demonstrate the discussed risk acceptance targets? We exemplary address this question in this Section, which is partly taken from our publication in [39].

We note that with respect to fatalities, the risk acceptance criteria following Section 3.1.1 can be directly taken as the acceptable ADS failure rate. The acceptable ADS failure rate is in the following termed target level of safety (TLS) and denoted with $\lambda_{\mathrm{TLS_{sys}}}$. Let $\lambda_{\mathrm{F}_j} = \lambda_{\mathrm{sys}}$ be the ADS's rate of fatal accidents. The system should only be released if:

$$\lambda_{\mathrm{sys}} \leq \lambda_{\mathrm{TLS_{sys}}} \tag{3.2}$$

The goal of the ADS safety validation is to demonstrate Eq. (3.2). Other aspects such as monetary consequences of the risk in Eq. (3.1) are neglected in this work but have to be addressed in practice.

As is well known, empirically demonstrating compliance of an ADS with an acceptable risk (i.e. $\lambda_{\mathrm{sys}} \leq \lambda_{\mathrm{TLS_{sys}}}$) by means of field tests alone ("driving to safety" [41]) is hardly possible because of the large amount of required testing [35, 41, 65, 66]. [35] coined the term "the approval trap" for ADSs to describe this issue. To make matters worse, in principle every modification in the design of a specific ADS would require a new series of tests.

In Figure 2.1, similar to [35], we demonstrate the approval trap considering a hypothetical TLS of $\lambda_{\mathrm{TLS_{sys}}} = 10^{-9}\ \mathrm{h}^{-1}$ with an approach we developed in [65].

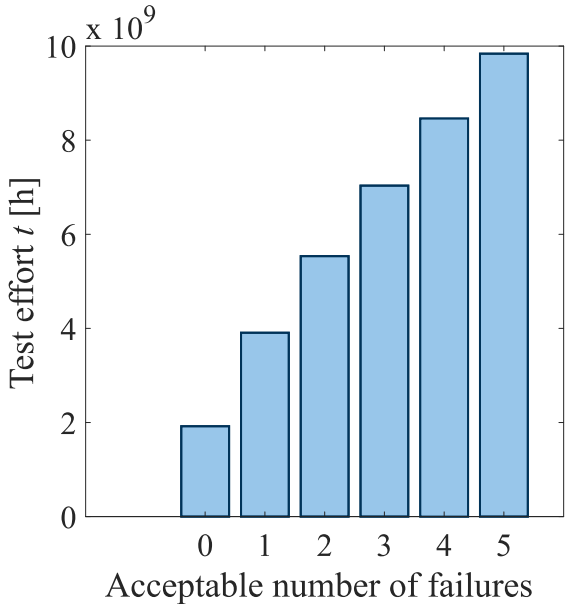

Figure 3.3 Exemplary illustration of the required number of hours to empirically demonstrate compliance with $\lambda_{\mathrm{TLS}} = 10^{-9}\ \mathrm{h}^{-1}$ in function of the acceptable number of failures in the test. Based on [65], adapted from [39].

Illustrated in Figure 2.1 are the required number of hours to empirically demonstrate $\lambda_{sys} \leq \lambda_{TLS_{sys}}$ with a credibility of 95 %, over the corresponding acceptable number of failures (i.e. severe accidents) in the test. The calculations underlying Figure 2.1 [65] are presented in Section 4.3, here we report the results only. As illustrated, at least $2/\lambda_{TLS_{sys}}$ failure free testing is required (equivalent to ≈ 230000 yr) to demonstrate compliance with the TLS according to [65]. It is concluded that the required test effort for an empirical safety validation of ADSs is hardly manageable. Driving to safety is therefore not directly possible without additional measures.

To address the approval trap, [66] propose a so called brave introduction of ADSs. The idea is to first derive an upper confidence limit on an ADS's accident rate λ_{sys} from a limited field test. Based on the upper confidence limit on λ_{sys}, the number of kilometers in a given year that all ADSs combined are allowed to drive are then derived such that the absolute risk for society is unlikely to be increased significantly[34]. At the end of the year, the upper confidence limit on λ_{sys} is updated with new data from the field and the approach is iteratively repeated until enough kilometers are driven to demonstrate Eq. (3.2). This approach however might contradict the equity risk acceptance criterion (Section 3.1.1), as the risk for the drivers of bravely introduced but potentially unsafe ADSs could exceed an upper limit.

Another approach to overcome the approval trap is to formally prove the safety of an ADS's actions [42]. The idea of formally proving the safety of (robot) control algorithms has its origin in robotics [145–149]. [42] prove the safety of an ADS by restricting its path planning such that the ADS can only chose a safe action. An action is considered safe, if it cannot cause an accident of the ADS's blame[35]. The prerequisite for this formal safety validation is however that the probability of perception errors is sufficiently low, otherwise the claim of a provably safe path planning is flawed[36]. Therefore, validating *perception reliability* e.g. by means of field tests is a requirement for provably safe ADS / robot control.

[34] [66] argue that the risk of ADSs for society is negligible, if the number of accidents due to ADSs are within the observed variability (i.e. standard deviation) of the number of human accidents per year.

[35] Blame in turn is formally defined by considering the driving dynamics of the vehicles and bounds on e.g. reaction time and maximum deceleration. With this formalism, the ego-vehicle is for instance not to blame for an accident if it kept a safe distance to other vehicles prior to an accident, which allows to brake in time to prevent an accident. If then an accident happened while ego-vehicle decelerated adequately, it is argued that it cannot be the blame of the ego-vehicle. [42]

[36] Additionally, as is pointed out in [44], the models underlying the formal prove of safety have to be validated themselves. In the words of Mitch et al: "we […] prove that collisions can never occur (as long as the robot system fits to the model)." [145].

3.1.3 System Reliability Theory and System Decomposition

This Section is partly taken from our publication in [39].

If one cannot perform a system safety validation by driving to safety, how can one estimate and demonstrate system safety? System reliability theory and hazard analysis techniques [150–152] have been developed exactly for this purpose: to estimate and predict the probability of failure of technical systems and to design systems with acceptably low failure rates, without an empirical safety validation of the full system.

System reliability theory describes a system in terms of its (sub-)functionalities (or components) and estimates the system failure probability based on the failure probabilities and dependencies of the (sub-)functionalities. On the level of (sub-)functionalities, the failure probabilities are typically easier to assess than on the system level. For an ADS, this implies that one needs to adequately describe the system functionalities by means of a system decomposition.

As described in Chapter 2, the generic ADS functionalities are *sense*, *plan* and *act* [42, 69, 70]. A high level reliability block diagram (RBD) therefore is as illustrated in Figure 3.4, corresponding to a series system with three components. Perception represents the environment sensors and the generation of an environment model with sensor data fusion, Function includes e.g. the situation interpretation and path planning. Actuation executes the planned path.

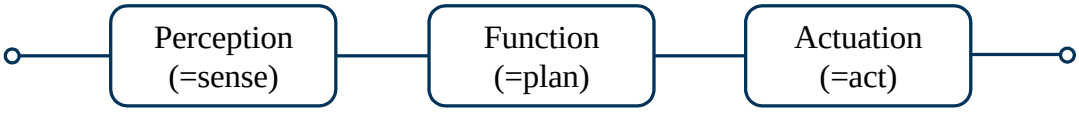

Figure 3.4 High level system reliability block diagram (RBD) for ADSs. Taken from [39].

If at least one of the functionalities in Figure 3.4 fails in a safety-critical way (i.e. makes a safety-critical error that causes an accident), then the ADS fails. The failure rate of the series system in Figure 3.4 is approximated by [153, 154]:

$$\lambda_{sys} \approx \lambda_{per} + \lambda_{funct} + \lambda_{act} \tag{3.3}$$

where λ_{sys} is the ADS failure rate (fatal accident rate), λ_{per} the rate of safety-critical failures of the perception module, λ_{funct} the rate of safety-critical failures of the automated driving function module, and λ_{act} the rate of safety-critical failures of the actuation. Eq. (2.2) is an upper bound on λ_{sys} because it assumes mutually exclusive failures between the different functionalities. This is a conservative approximation, i.e. λ_{sys} can only be overestimated for known λ_{per}, λ_{funct} and λ_{act}. Accounting for statistical dependencies in failures among the three modules is an opportunity to obtain a lower, more accurate estimate of λ_{sys} [153, 154].

The task of validating ADS safety is formalized by Eq. (3.2). In combination with Eq. (3.3), the safety validation of an ADS is thus equivalent to demonstrating:

$$\lambda_{\text{per}} + \lambda_{\text{funct}} + \lambda_{\text{act}} \leq \lambda_{\text{TLS}_{\text{sys}}} \tag{3.4}$$

With this system decomposition, safety validation is based on individually considering λ_{per}, λ_{funct} and λ_{act}. An important prerequisite underlying such a system decomposition is that safety-critical errors in the perception module can be separated from errors in the function module.

λ_{per} is related to the *perception reliability* as defined in Section 1.3. Demonstrating *perception reliability* is equivalent to demonstrating an acceptable λ_{per}. In this thesis, we focus on estimating λ_{per} of the perception module, λ_{funct} and λ_{act} are not further considered. Safe path planning can for instance be approached by formally proving the safety of control algorithms [42] together with simulations and, in the automotive domain, ISO 26262 is employed to design sufficiently safe E/E systems (e.g. actuation).

3.1.4 Challenges in Demonstrating Perception Reliability

This Section discusses challenges associated with estimating λ_{per} and is partly taken from our publication in [39].

The Approval Trap for Environment Perception

As follows from Eq. (3.4), the TLS for the perception module has to be even stricter than for the ADS itself. Hence, the approval trap also applies to validating the *perception reliability* of an ADS [65].

Perception Error Definition

To assess λ_{per}, one first needs to define what constitutes a safety-critical perception error. This is hardly possible without considering e.g. the path planning of the automated driving functionality and without considering the actual driving situation, because a prerequisite for an ADS to react adequately is situational awareness through a situation interpretation [67, 155].

For instance, a false positive detection, i.e. a wrong indication of an object when there is none, might be safety critical if the ADS reacts by applying maximal deceleration and another vehicle in the back of the ADS cannot brake in time to prevent an accident. The same situation, but without another vehicle following the ADS, or with the ADS choosing a different action, might be unpleasant but not necessarily safety-critical. Similarly, a vehicle changing its lane, which is wrongly interpreted to not change its lane due to an error in the estimated lateral velocity, could cause a safety-critical situation. Depending on the implemented situation interpretation, the same

estimation error in the velocity of another vehicle could have no consequence at all in a different situation. Another example is an error in the position estimate of another vehicle. Even a small error in the position estimate of a vehicle could lead to a wrong situation interpretation, for instance by assigning the vehicle to the wrong lane while large position errors of objects that are not safety-critical are irrelevant.

To borrow the formalism from [42], let s be the true state describing the environment and $a(s)$ the action the ADS's path planning module selects given s. The action $a(s)$ is executed by the actors. $\hat{s}(s)$ is introduced, which is the perceived environment provided by the perception module given s. $\hat{s}(s)$ is a probabilistic function. The selected action under the perceived environment is $a(\hat{s}(s))$. A theoretical definition of a safety-critical perception error is: if the action $a(\hat{s}(s))$ leads to an accident, but $a(s)$ would not, then a safety-critical perception error has occurred. As an example, consider the case that the ADS does not detect an obstacle and crashes into it. If the ADS would have been able to prevent the crash if it detected the obstacle, then a perception error has occurred.

In practice, it is challenging to comprehensively derive a catalogue of perception errors because one cannot specify the infinite number of combinations of states in e.g. actual situations s, the resulting perception $\hat{s}(s)$ and of the interpretation of the perception combined with the planned paths (selected actions) a. The safety-criticality of perception errors might therefore only be revealed at test time.

In reality, the outcome of an action (accident or no accident) further depends on multiple time steps, i.e. on how a driving situation evolves, and the actions are adapted constantly in a control loop to the changing perception $\hat{s}(s)$. For simplicity, we do not include a time parameter here but point out that time has to be included in a definition of a safety-critical perception errors.

Generating a Reference Truth

To assess λ_{per} with standard methods, one has to compare the perceived environment $\hat{s}(s)$ with a reference truth (ground truth) s to identify perception errors [156, 157]. The reference truth is either derived from suitable reference sensors (optionally) in combination with human data labeling, directly from the implemented online sensor data fusion [57, 58, 110, 118], or from automatic offline labeling algorithms [158].

Employing reference sensors requires considerable efforts, as the reference sensors must have an extremely high performance. A reference truth derived from online sensor data fusion is not error free. Offline automatic data labeling with offline fusion has larger accuracy than online sensor data fusion, but is still imperfect. These errors, and the fact that labeling is typically deterministic, are relevant when trying to estimate small error probabilities. Moreover, setting up a reference truth requires in part human data labeling because, at present, human data labeling is less error prone

compared with automatic data labeling. The post processing of the reference truth data is thus time consuming and only a limited amount can be handled.

Time Variable Perception Performance

Another challenge lies in the influence of uncertain and variable environment conditions on the perception performance [65]. Exemplary influencing conditions are as discussed in Section 2.1 the weather, background illumination, dirt, the properties of objects and traffic participants themselves as well as their relative orientation and location [104, 159, 160]. Let the conditions with influence on perception performance be described by a vector of context variables $\mathbf{E}$ and their realizations with $\mathbf{e}$.

Due to $\mathbf{e}$, λ_{per} is not a constant but varies with time t, i.e. $\lambda_{\text{per}}(t)$ [161]. Instead of considering λ_{per} as a function of time, one can formulate it in function of $\mathbf{e}$, i.e. $\lambda_{\text{per}}(\mathbf{e})$. As an example, Figure 3.5 schematically illustrates $\lambda_{\text{per}}(\mathbf{e})$ with the example of dry and snowy weather.

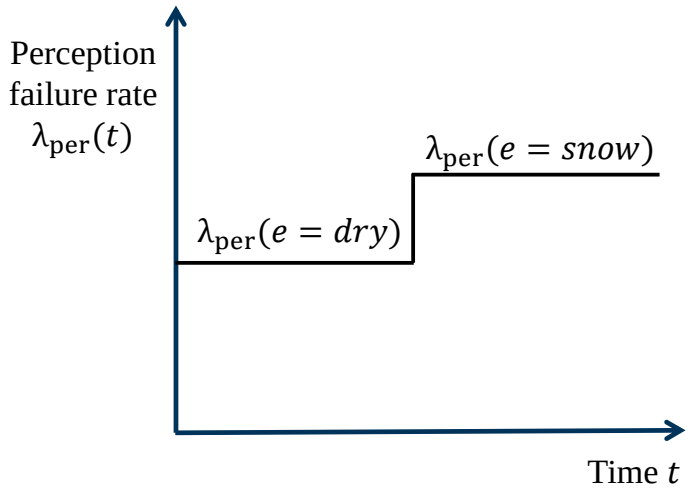

Figure 3.5 Schematic illustration of the temporal variability in the perception error rate λ_{per} due to context variables e. In the graph, e is the weather with two states: dry and snow.

A high variation of driving situations and the complexity of environment perception leads to a large number of relevant context variables [40]. The challenge therefore lies in identifying relevant context variables $\mathbf{E}$ and quantifying their influence on perception performance, i.e. $\lambda_{\text{per}}(\mathbf{e})$.

Representativeness of Testing Perception Reliability

The dependence of perception performance on context variables $\mathbf{E}$ is associated to the challenge of conducting a representative test [40, 54]. In very long representative field tests, the influence of $\mathbf{E}$ is automatically accounted for, since the observed failure rate is averaged over all $\mathbf{e}$.

In such a test, the time variable perception error rate $\lambda_{\text{per}}(t)$ is related to $\lambda_{\text{per}}(\mathbf{e})$:

$$\lambda_{\text{per}} = \int_{\mathbf{e}} \lambda_{\text{per}}(\mathbf{e}) \cdot f_{\mathbf{E}}(\mathbf{e}) d\mathbf{e} = \frac{1}{T} \int_{T=0}^{\infty} \lambda_{\text{per}}(t) dt \tag{3.5}$$

where $f_{\mathbf{E}}(\mathbf{e})$ is a probability density function (PDF) describing the frequency of the randomly occurring context variables **E**. If a test is representative and long (RHS), then the occurrence frequency of the context variables $f_{\mathbf{E}}(\mathbf{e})$ is accounted for (LHS) and one estimates the global average perception error rate λ_{per}.

It follows from the equality in Eq. (3.5) that a test is representative, if it is in line with $f_{\mathbf{E}}(\mathbf{e})$. If not in accordance with $f_{\mathbf{E}}(\mathbf{e})$, the estimate of λ_{per} is biased. A biased estimate of λ_{per} is for example obtained if one exclusively tests the environment perception in favorable weather conditions, e.g. in Figure 3.5 one would only test in $e = dry$.

In practice, it is challenging to make sure a test is representative, for instance because it is challenging to know all context variables **E** with their distribution and because each geographical region is associated with a different distribution $f_{\mathbf{E}}(\mathbf{e})$. Hence, even if a test is representative, say, for German roads, it might not be for Chinese roads because different countries have different distributions of e.g. infrastructure elements, traffic participants or different climate.

Statistical Perception Error Dependence

Due to common context variables **e**, identical sensor types, similar physical measurement principles and similar processing algorithms, a certain degree of statistical dependence among perception errors in different sensors is expected [156, 157]. This error dependence has to be accounted for in order to correctly estimate the joint frequencies of perception errors in multiple sensors [65].

A second type of statistical error dependence are temporal dependencies, as the common influencing factors in **e** such as snowfall are present for a certain time interval [65]. Correctly representing all dependencies in statistical models for perception errors is a challenge.

Changing the System during Development: Reliability Growth?

During the ADS development, the system is modified frequently, e.g. by software updates or by changing the position of a sensor in the vehicle. Ideally, one would expect a reliability growth associated with software updates, i.e. a decrease in λ_{per} with each update. This is however not certain. It is therefore not straightforward to account for these modifications in reliability modeling, estimation and testing. Strictly, each modification would render previous estimates of

λ_{per} as obsolete. Devising methods to model the influence of these changes and updates on λ_{per} without requiring to repeat all testing is challenging.

Potentially, methods from software reliability analysis such as regression testing [162], change impact analysis [163–165] and reliability growth modeling [151] can help to address challenges involved with system updates. Regression tests aim at verifying with optimized test strategies that modifications do not impair performance and interfere with functionalities, e.g. by selecting only tests relevant for the changes made and by prioritizing tests according to their relevance for detecting faults [162]. Change impact analysis deals with identifying elements impacted by a (software) change and with estimating the effect of change [163–165]. Reliability growth models account for increasing reliability over time by eliminating faults in a system [151].

3.2 Established Safety Concepts

Eq. (3.4) formalizes the task of demonstrating *perception reliability* for an ADS's safety validation. As pointed out by [66], in principle there is a difference between *developing* a system that is safe and (formally) *validating* that a safe system has been developed. In practice a clear separation between developing a safe system and validating system safety is however not always possible, this Section therefore discusses in light of the challenges identified in Section 3.1.4 if established standards and safety concepts enable the validation and development of a reliable environment perception for ADS with *higher levels of driving automation*. Test methods to validate system safety are in focus of Section 3.3.

Particularly, Section 3.2.1 reviews ISO 26262 for functional safety and Section 3.2.2 reviews the Code of Practice for the Design and Evaluation of ADAS [40, 43, 56, 63].

3.2.1 Functional Safety: ISO 26262 and the V-Model

IEC 61508 is the generic standard for the development of functional safe electrical, electronic and programmable electronic systems [166]. ISO 26262 is the automotive specific adaption of IEC 61508 with requirements for the development of safety-relevant electrical and/or electronic (E/E) systems within road vehicles [40, 54, 62]. The scope of ISO 26262 is functional safety, defined as the "absence of unreasonable risk due to hazards caused by malfunctioning behavior of E/E systems" [62]. Malfunctioning behavior is the "failure or unintended behavior of an item with respect to its design intent" [62].

The structure and content of ISO 26262 follows the V-Model process, which is illustrated in Figure 3.6 [40, 54, 62, 63]. The development of a product (left branch in Figure 3.6) with the V-Modell progresses from the specification of a system's functionalities over the design on the system level

to the detailed specification, design and implementation of its components [40, 44, 62]. The integration, verification and validation[37] (right branch in Figure 3.6) progresses from the components to the fully integrated system [40, 44, 62].

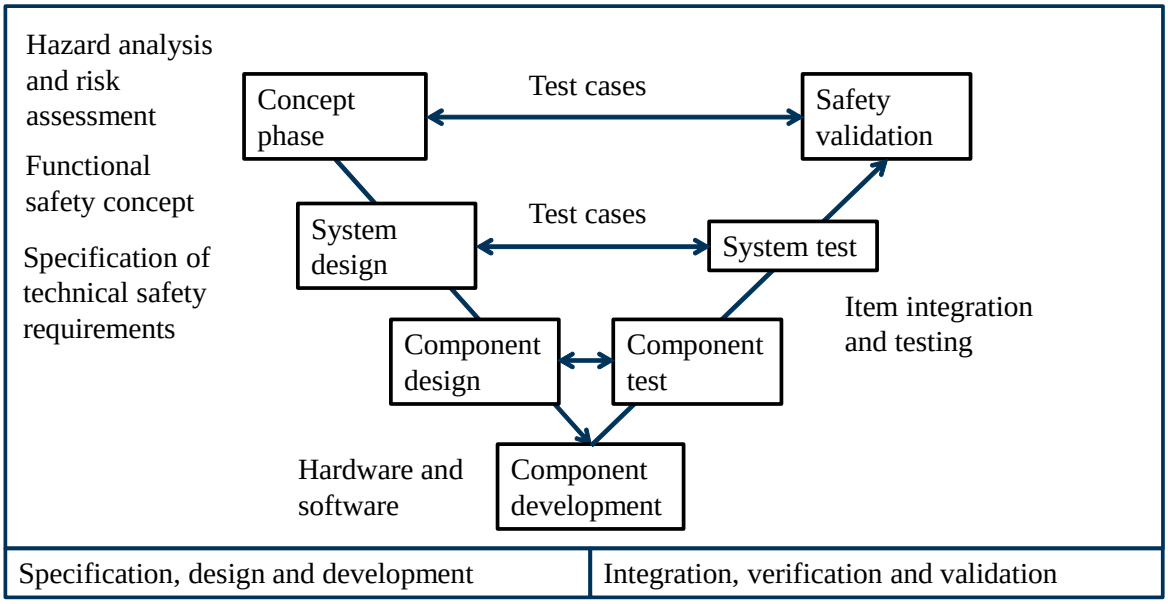

Figure 3.6 Simplified illustration of the V-Modell process in ISO 26262 [62]. In reference to Bild 6 and Bild 7 in [40].

A summary of ISO 26262 is given in the following to discuss its applicability for developing a reliable environment perception.

Achieving Functional Safety: Preliminary Hazard Analysis and Risk Assessment

In the concept phase, a hazard analysis and risk assessment is performed based on an initial description of the system's functionalities (the item[38] definition) [40, 43, 54, 167]. Hazards due to potential malfunction of an item are systematically identified through expert judgment, checklists, failure modes and effects analysis (FMEA), field experience or other suitable methods [167, 168]. The risk associated with identified hazards is then assessed by evaluating the exposure, controllability and severity of the hazardous events [167]. The risk assessment is often further supported by FMEA, event tree analysis (ETA) or fault tree analysis (FTA) [43, 63, 167, 169].

[37] Verification and validation are defined here in reference to [56], which gives a definition of these terms.
Verification: "Assuring, e.g. by testing, that a component, a sub-system, a system or a process is working as required and specified." [56].
Validation: "The process of evaluating a system or component during or at the end of the development process to determine whether it satisfies the expectations." [56].
[38] An item is a "system or array of systems to implement a function at the vehicle level, to which ISO 26262 is applied" [62].

Reverse engineering the risk assessment of ISO 26262 leads to the following expression of risk associated with a single hazard, as utilized in ISO 26262[39]:

$$\text{Risk} = \Pr(E) \cdot \lambda_{\text{Fail}} \cdot [1 - \Pr(C|E, Fail)] \cdot S \tag{3.6}$$

where $\Pr(E)$[40] is the exposure probability towards situations E in which the hazardous event could occur, λ_{Fail} is the item's rate of failure that could lead to the hazard, $\Pr(C|E, Fail)$ is the controllability[41], i.e. the probability that the harm (accident) can be averted by drivers or other persons, conditional on a failure of the item in a potentially hazardous situation. S is the severity of the damage. [167]

Eq. (3.6) is not directly evaluated nor reported in ISO 26262. Instead, the exposure probability is evaluated in five discrete categories (from incredible to high probability), the controllability in four discrete categories (from controllable in general to difficult to control or uncontrollable) and the severity in four discrete categories (from no injuries to life-threatening and fatal injuries) [167]. With these categories, the preliminary risk due to a hazard is classified with a lookup table in terms of the automotive safety integrity level (ASIL) [167]. It is distinguished between five risk classes to qualitatively estimate the preliminary risk (ASIL A-D and QM: quality management) [167]. The item's failure rate λ_{Fail} is not included in the initial risk assessment [167].

The approach in ISO 26262 to achieve an acceptable risk is explained with Figure 3.7, in reference to a similar discussion of IEC 61508's safety integrity level (SIL) in [125, 171]. A high preliminary risk (ASIL D) implies a large required risk reduction during the development, leading to strict requirements for the development of an item. A low preliminary risk (e.g. ASIL A) in contrast implies a small required risk reduction, leading to less strict requirements for the development of an item. The estimated preliminary risk expressed as an ASIL therefore determines the required risk reduction measures during the development of the product to achieve an acceptable risk, thereby implicitly imposing requirements on λ_{Fail}[42]. The acceptable risk is not explicitly defined in ISO 26262 but is claimed to be achieved by the ASIL specific requirements on the specification of safety goals, on the product development and on testing.

[39] An equivalent expression of risk on the basis of ISO 26262 is reported in [170].
[40] In ISO 26262, the exposure is denoted simply with E and the controllability with C [167]. The notation deviates here from [167] to be in line with notation used in probability theory.
[41] Controllability in ISO 26262 is the: "ability to avoid a specified harm or damage through the timely reactions of the persons involved, possibly with support from external measures" [62].
[42] One is able to explicitly derive the acceptable item failure rate by rearranging Eq. (3.6) for λ_{Fail} and inserting the acceptable risk (if it is defined), the exposure, the controllability and the severity. Such an explicit approach is for instance proposed in [170] to derive acceptable error rates for functional deficiencies of ADAS.

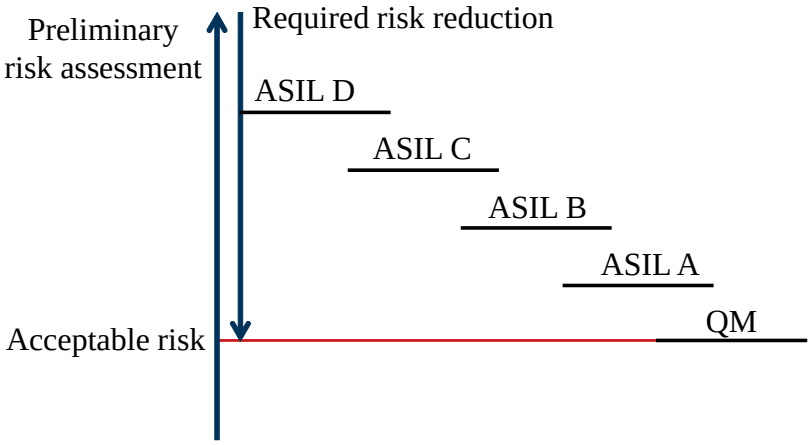

Figure 3.7 Achieving an acceptable risk level in ISO 26262: First, a hazard analysis and risk assessment is performed to determine the automotive safety integrity level (ASIL). The ASIL defines requirements on the development of the system, thereby being a measure for the required risk reduction to achieve an acceptable risk. Based on Figure 3.7 in [125], Fig. 3 in [54] and Fig. 2 in [171].

Specification, Design and Development

Safety goals are defined for each hazardous event on basis of the hazard analysis and risk assessment. A functional safety concept is derived from the safety goals, specifying functional safety requirements to avoid unreasonable risk. The functional safety requirements inherit the ASIL of the corresponding hazards. Technical safety requirements necessary to realize the functional safety requirements are formulated, e.g. by specifying "the measures relating to the detection, indication and control of faults in the system itself" [172] and "the measures that enable the system to achieve or maintain a safe state" [172]. Finally, the system is designed. During the system design, safety and non-safety related requirements are allocated to the architectural elements of the system. Based on the specified requirements, the system's hardware and software elements are designed and implemented. [167, 172]

Integration, Verification and Validation

After developing the hardware and software, the elements are integrated and tested at each integration stage to verify that the design complies with the specified requirements and that the design is correctly implemented (right branch in Figure 3.6). On the level of the integrated item in the vehicle, it is validated by tests that the safety goals are appropriate for functional safety and that the safety goals are being complied with. The safety validation includes for example an evaluation of controllability against corresponding assumptions made in the concept phase [173] and an evaluation of the effectiveness of controlling failures with safety measures. [172]

Test cases are specified for the verification and safety validation. The test cases are for instance derived from requirements, the expected functional behavior, experience and expert knowledge,

boundary values, interactions at interfaces, the statistical distribution of environmental conditions or field experience. The specified test cases are associated with pass/fail acceptance criteria and are executed with a variety of reproducible test methods such as tests under laboratory conditions, tests on proving grounds and simulation methods. Moreover, field tests in real-life conditions are conducted. In addition to testing, ISO 26262 allows to use FMEA, FTA, RBDs, Markov models and ETA for system safety validation. [138, 172, 174, 175]

Applicability to the Development of a Reliable Environment Perception

The functional safety concept in ISO 26262 aims at an item behaving as specified. This is achieved by ensuring sufficiently rare hardware failures, sufficiently rare software failures and providing for driver controllability. As discussed in Section 1.2.2, this is the traditional safety concept for a driver-vehicle system and to a large degree for ADAS [44]. The environment perception can however be deficient without hardware and software failures [40, 54, 63, 64], potentially leading to an inadequate system behavior (functional deficiencies) [64, 170]. For instance, a false positive perception error (indicating an object when none is present) is due to the physical measurement principle of a sensor in combination with implemented algorithms (e.g. a lidar detects snowfall as an object, see Section 2.1).

The nominal performance (i.e. the *Safety of the Intended Functionality* SOTIF) of a system is not addressed in ISO 26262 [54, 62][43]. Therefore, ISO 26262 does not comprehensively cover perception errors with its failure definition. One could argue that the scope of ISO 26262 is easily extended to cover perception errors by adapting its failure definition. But is it possible to apply the processes and methods of ISO 26262 to the development of a reliable environment perception, and in the wider sense, to the development of safe ADSs?

It is possible to conduct a hazard analysis to identify hazards of perception errors. Also, (preliminarily) estimating the risk of perception errors is possible with common methods such as FMEA, ETA or FTA. Limitations of the (preliminary) hazard analysis and risk assessment are potential inconsistencies and subjectivity [168, 169]. Therefore, the hazard analysis and risk assessment is predominantly a design tool, i.e. its purpose is *developing* a safe system (see Figure 3.7) and not formally *validating* that a safe system has been developed. Combining the hazard analysis and risk assessment with systematic and formal modeling approaches is a way to mitigate these limitations [168, 169]. However, due to the complexity of the problem, with a large number of combinations of context variables with influence on perception errors and the difficulty of specifying perception errors (see Section 3.1.4), it is not guaranteed that the estimated risk is accurate and comprehensively covers all potential perception errors.

[43] Currently, a standard addressing SOTIF for ADAS is being drafted (ISO PAS 21448).

Including driver controllability in the hazard analysis and risk assessment as well as in the design of the system [44] is by definition not possible for ADSs with *higher levels of driving automation*, when the system is engaged[44]. This means $\Pr(C|E,Fail) = 0$ in Eq. (3.6)[45], leading to stricter requirements (i.e. a higher ASIL). As a consequence, it is more difficult to achieve safety for *higher levels of driving automation*.

The functional safety concept achieves an acceptable risk by controlling λ_{Fail} with adequate safety measures (see Figure 3.7) [171]. Exemplary safety measures are fault detection, failure mitigation, transition to a safe state and fault tolerance mechanisms [167]. These measures are challenging to apply to the development of a reliable environment perception [40], because perception errors are difficult to observe. While hardware failures (e.g. no power) or software errors (e.g. a value is out of range) are often observable and hence can be dealt with by adequate safety measures [63], a perception error requires a reference truth to be identified[46] (see Section 3.1.4). Potential exceptions that to some degree enable the detection of perception errors are heuristics, plausibility checks and sensor redundancies [59], [176] as cited in [40]. It has to be studied how and whether these measures lead to an acceptable risk, which is not clear due to the lack of certainty in detecting perception errors [40]. Furthermore, it is possible to estimate the accuracy of perception online in the ADS, e.g. with the uncertainty of the state estimates in the Bayes filter, or with an estimate of the a posteriori probability of track existence in the JIPDA-Filter, see Section 2.2.2 [60]. These estimates allow to optimize the perception but do not allow to predict or detect perception errors [60], which is intuitively understood in case of a false negative detection (wrongly not indicating an existing object).

Another issue is that a comprehensive system specification is the prerequisite for verifying compliance with requirements in test cases [54]. It does not seem possible to comprehensively specify environment perception requirements for all possible situations and context variables due to the involved complexities [40, 63]. Limitations of the test case approach for system verification and validation are further discussed in Section 3.3.1.

To conclude, ISO 26262 provides an initial basis for the SOTIF of ADS because the methods in ISO 26262 are partly transferrable to the development of a reliable environment perception. In light of the challenges involved with specifying requirements, verifying the requirements and applying safety measures to the environment perception [40], however, applying the methods in

[44] An exception is a level 3 functionality which passes the control to the diver within an adequate time frame in case a system limitation is detected [14, 49].
[45] Another exception is that other drivers or vehicles could be able to control a hazardous situation caused by an ADS. Then, $\Pr(C|E,Fail) \neq 0$.
[46] This applies to (stochastic) perception errors due to the inherent uncertainties of a sensor. Systematic sensing errors could be addressed with adequate safety measures. For example, a blindness detection could identify that a sensor is blinded by dirt on its cover and initiate a cleaning program. Another example is that one can detect if the ambient light is insufficient for cameras.

ISO 26262 is not sufficient to claim the absence of unreasonable risk due to perception errors, and in the wider sense, an acceptable risk for ADS.

3.2.2 Code of Practice for the Design and Evaluation of ADAS

ISO 26262 is formulated generically, without being specific on its processes and methods. For instance, no detailed instructions are given on how to evaluate the controllability or how to develop an ADAS with environment perception. To provide more specific guidance on these topics, the Code of Practice for the Design and Evaluation of ADAS documented the state of the art[47] for the introduction of safe ADAS [40, 43, 56, 63]. For instance, the Code of Practice describes procedures and provides checklists for the development and safety validation of ADAS such as Adaptive Cruise Control or Lane Keeping Assist [44, 56]. Additionally, the Code of Practice describes system reliability analysis methods such as a hazard and operability study (HAZOP), FMEA and FTA [56].

The Code of Practice's central premise is that "[…] an ADAS is considered safe, as long as the driver is able to control the vehicle." [56]. Based on this premise, the focus of the Code of Practice is to design for and demonstrate driver controllability of the ADAS [40, 44, 56, 63], defined as the: "likelihood that the driver can cope with driving situations including ADAS-assisted driving, system limits and system failures" [56]. The development process in the Code of Practice is illustrated in Figure 3.8. In practice, this process is realized in analogy to the V-Modell (see Figure 3.6) [44].

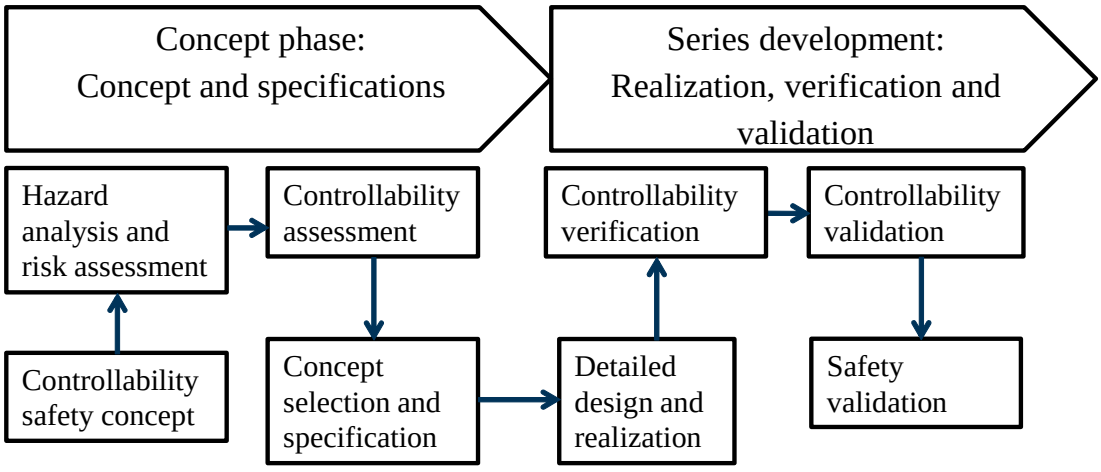

Figure 3.8 Safety process for the development of ADAS according to the Code of Practice. Adapted from Figure 2 in [56].

[47] It is not necessarily given that the Code of Practice published in 2009 still documents the state of the art due to technological and scientific progress.

This safety process is summarized in the following to highlight how the safety of ADASs is achieved and validated, and to discuss its applicability to ADSs with *higher levels of driving automation* and their environment perception.

ADAS Development and Safety Process

In the concept phase, a draft is set up which contains concepts on the ADAS functionality, its use case (i.e. the domain and the scenarios of usage) and the human machine interaction (i.e. a sketch on how to operate the system) with a controllability safety concept. Next, a preliminary hazard analysis and risk assessment is performed to identify hazardous situations and hazards within the ADAS function. The hazard analysis and risk assessment is essentially identical to the approach in ISO 26262 with special focus on controllability. From the ADAS draft and the hazard analysis and risk assessment, controllability concepts detailing the human machine interactions are derived. Criteria are specified to evaluate and compare different concepts, taking into account the controllability requirements derived from the risk assessment. The best among potentially multiple concepts is selected and is incorporated into a detailed system specification. [56]

In the series development, the system including its human machine interaction is designed in detail based on the specifications. The hazard analysis and risk assessment is updated with the detailed design. It is then verified if requirements on controllability are met by the realization of the design. Ultimately the system's safety is validated with a final proof of controllability. To this end, relevant driving scenarios are identified and collected in a list as test scenarios. The identification of relevant scenarios is informed by the hazard analysis and risk assessment [55]. The evaluation of controllability is restricted to the most relevant scenarios under the assumption that similar situations have the same controllability. Selected test scenarios are associated with specified pass/fail criteria and their controllability is assessed by experts, in field tests, on proving grounds and in (driving) simulators [55, 64]. [56]

Additional to the evaluation of the selected test scenarios, an extensive field test of potentially millions of test kilometer supports the final demonstration of controllability and system safety validation [64]. For further information on controllability evaluation it is referred to [63, 64, 168, 177].

Perception Errors in Light of the Code of Practice

Under the Code of Practice, perception errors are not safety-relevant as long as the driver is able to control the consequences. Consider for example an Automatic Emergency Brake (AEB) functionality. False negative (FN) errors have little safety relevance for an AEB system (but reduce its effectiveness). A FN error has a large controllability, as it is the driver's regular duty to control the vehicle in reaction to obstacles and other traffic participants. A false positive (FP) perception

error, however, could lead to an inadequate emergency brake, which is safety relevant because it has a small controllability for following vehicles. This asymmetry in safety relevance of FN and FP errors allows to optimize the perception w.r.t. a reduction of FP errors, at the cost of an increased number of FN errors [44]. An acceptably low FP error rate is then in practice validated with a field test in real traffic [64]. In combination with a design for controllability of FP errors (e.g. limitation of maximum deceleration), the system's safety can be validated with a manageable number of test kilometers [44, 55, 64]. In [64], it is for instance described that a final safety validation of an AEB system was based on 2 Mio km field tests in real traffic.

Applicability to Higher Levels of Driving Automation

As discussed in Section 1.2.1, with *higher levels of driving automation*, the system instead of the human driver is responsible for monitoring and operating the vehicle and for monitoring the environment. Therefore, the abilities of the human driver cannot be part of the safety concept for a level 3-5 ADS. The Code of Practice's paradigm of designing for and demonstrating driver controllability hence cannot apply to the development of ADS with *higher levels of driving automation*. As a consequence, all types of perception errors (e.g. FN and FP) are potentially safety relevant because it is not possible to validate the SOTIF by assuming and demonstrating that the driver is able to control the consequences of perception errors.

3.3 Established Test Methods

Section 3.2 shows that existing safety concepts focusing on functional safety and controllability are not sufficient to develop a reliable environment perception for ADS with *higher levels of driving automation*. Therefore, it is crucial to explicitly assess *perception reliability* to validate the SOTIF.

This Section investigates the applicability of established automotive test procedures on demonstrating an ADS's *perception reliability*. Scenario based testing together with field tests represents the state of the art in testing ADAS[48] [40, 44, 56, 172, 178–181]. Section 3.2.1 revealed that scenario based testing is also central for the verification of functional safety in ISO 26262. We therefore first review in Section 3.3.1 how test cases are derived with the scenario based approach and outline in Section 3.3.2 how tests are executed.

[48] As there are currently only level 1-2 ADSs in the market, the state of the art in testing level 3-5 ADSs cannot be described.

3.3.1 Test Case Generation: Scenario Based Testing

Driving scenarios define use cases, support the system development and define test cases for ADSs [40, 44, 56, 178–182]. A driving scenario is for instance described in terms of the road geometry, the spatial and temporal constellation of traffic participants with their expected states, infrastructure elements, static obstacles, hazards, the ego-vehicles intended actions, environmental conditions, the drivers' conditions and the drivers' goals [40, 56, 178–183].

In the context of scenario based testing, it has to be distinguished between test case generation and test execution [184]. The following terminology is proposed in [178] for the generation of test cases, which is motivated by an inconsistent use of terms like scenario and situation [178, 179, 182]:

- Scene: geometry and type of roads, position of traffic signs and lights, static obstacles, environmental conditions, dynamic elements (other traffic participants, state of traffic lights).
- Situation: the combination of the scene, the ego-vehicle and the designated actions of the driver or of the automated driving functionality.
- Scenario: a sequence of situations, including the expected actions of the actors (i.e. traffic participants).

Similar definitions are given in [182], which slightly deviate from [178]. The situation is in [182] interpreted to contain only the elements of the scene that are relevant for an actor to achieve its driving goals. A scenario is in [182] defined as the temporal development of a sequence of scenes with prescribed actions and events between the scenes and including the goals of the actors. Because of an inconsistent use of scenarios in different phases of ADAS development, [180] propose to distinguish between functional, logical and specific scenarios (with decreasing level of abstraction). The terminology is here not presented to discuss which definitions are more appropriate but to clarify the concept of scenario based testing. For further references on terminology we refer to [180, 182].

The specification of system requirements, the risk assessment of driving scenarios and the range of parameter values at the systems' interfaces are according to [185] the basis for generating test cases. Additionally, Section 3.2.1 discussed that test cases are derived from expert knowledge, the statistical distribution of environment conditions and field experience. The identification of test cases from field tests is for instance described in [61, 186].

Based on such information, scenarios and associated parameters relevant for an automated driving functionality are identified [184, 187, 188]. The test cases are then systematically generated by detailing and varying the relevant parameters of the driving scenarios and by specifying pass/fail

criteria [40, 43, 56, 71, 178, 179, 182, 184, 188]. To this end, the parameters of the test scenarios are for example discretized into classes [184, 188]. To generate the test cases, combinatorial methods are applied to the discretized parameters [184, 188]. In the context of ADSs, relevant test cases for ADSs were identified and collected in the PEGASUS project [181].

To manage the test effort in practice, test cases are restricted to the most representative scenarios, e.g. by evaluating their probability (i.e. the exposure, see e.g. [169]) or by selecting critical scenarios from experience and expert knowledge [40, 56, 61, 71]. [63] for instance describes an approach to select test scenarios. The approach limits the level of detail of the test scenarios by means of a relevance factor derived from the probability of the scenario and from controllability [63]. Ultimately, the selected test cases are collected in scenario catalogues and are executed with a variety of test methods, which are described in Section 3.3.2 [40, 44, 183].

While scenario based testing demonstrates to a large degree an acceptable risk for ADAS by assessing controllability and functional safety, it has limitations when applied to demonstrate the SOTIF and *perception reliability* with *higher levels of driving automation* [40, 44]. The two central reasons for the limited applicability of scenario based testing in this context are:

1.) The ADS and its environment perception have to handle potentially an infinite number of driving situations with a large number of context variables (see Section 3.1.4) [40, 61, 170, 184].
2.) The ADS is not able to rely on the drivers' abilities and controllability in case of failures (see Section 3.2.2) [44].

Point 1.) makes it difficult to explicitly specify and execute all relevant test cases [40, 61, 170] and to specify what constitutes an absence of perception errors (see Section 3.1.4). The difficulty of comprehensively specifying test cases is closely related to the difficulty of comprehensively specifying the requirements for an ADS and its environment perception [40, 71], as already discussed in 3.2.1.

Consider as an example the sensor specification "detect all relevant objects at all times while not indicating non-existing objects". Clearly, compliance with this requirement cannot be demonstrated through test cases because of the imprecise specification. To verify this requirement by means of test cases, one would need to specify all relevant objects that the perception must be able to detect as well as the situations in which these must be detected. For particular objects and specific driving situations, the requirements are specifiable [40, 63]. From these specifications, test cases could be derived to verify compliance with the requirements. Due to the large number of context variables, the combinations of their states, objects and possible driving situations (see Section 3.1.4), there is no guarantee that the specifications are sufficiently complete [40] to claim

the absence of unreasonable risk. The challenge is that not necessarily all relevant context variables and situations are known. If all context variables and situations are known, the number of test cases to systematically explore the space of the context variables grows exponentially with each additional dimension of test parameters due to the curse of dimensionality [115, 189]. Hence, strictly, one would need to define an uncountable number of test cases to claim the absence of unreasonable risk [66, 170, 184]. This seems not achievable at present.

Point 2.) invalidates the underlying assumption to restrict the test scenarios to the most relevant scenarios [44, 66]. As already explained in Section 3.2.2, it is assumed that a driver who is able to control the vehicle in one situation is also able to control the vehicle in similar situations due to her perceptive, cognitive and sensory-motoric abilities [44]. An ADS of *higher levels of driving automation* however excludes the driver[49] when the automated driving functionality is engaged.

Assuming similarly that the performance of environment perception evaluated in exemplary test cases is transferrable to all situations could turn out to be wrong [141] because of potential unaccounted influences of context variables (see Section 3.1.4) [190]. A small variation in the environment could due to the involved complexities lead to a perception error, e.g. because a different amount of photons hits a lidar's photo diode, and hence, cause a different ADS behavior in similar situations (see Section 2.1) [71]. Therefore, one cannot simply assume an absence of perception errors in all possible situations by successfully passing some test cases.

For ADAS it was further argued to restrict testing to situations in which an intervention (e.g. emergency brake) is safety critical [61]. The possibility of perception errors however renders almost all situations as potentially safety-relevant for an ADS with *higher levels of driving automation*, which is continuously and actively driving in its use case (e.g. highway pilot). Restricting test cases to exemplary situations is therefore inadmissible because relevant situations could be neglected otherwise [40].

It is concluded that scenario based testing is a tool to improve an ADS during its development and is applicable to verify an ADS's requirements. However, scenario based testing does not automatically allow to claim the absence of unreasonable risk due to perception errors *without additional measures*. To formally claim the absence of unreasonable risk with the described scenario based approach, one has to execute an uncountable number of test cases.

[49] An exception is a level 3 functionality which passes the control to the diver within an adequate time frame in case a system limitation is detected [14, 49]. This aspect is not further discussed because perception errors cannot always be anticipated for the purpose of handing over the control to a driver in a timely manner.

3.3.2 Test Execution

The generated and specified test cases are executed with a variety of test methods [40, 44]. These include simulations, controlled tests on proving grounds and field tests [40, 44]. This Section evaluates if the different test methods allow to validate *perception reliability*.

X-in-the-loop Simulations

Simulation methods execute test cases in virtual test drives to evaluate system specifications and to verify requirements early in the development process, when the system and its components are not yet fully realized. The drivers with their behavior, the vehicles, the environment and all relevant interactions have to be modeled to execute the test scenarios virtually. [40, 44, 71, 89, 183, 186, 187, 191–195]

As the development of the system progresses, models of functionalities, software code and the hardware of different components are realized (see Figure 3.6). The realized parts of a system are combined with virtually simulated representations of system parts that are not yet realized, together with virtual representations of the environment and the drivers. It is distinguished between different types of X-in-the-loop simulations methods, depending on which parts of the system, the environment and the drivers are represented virtually or in reality. To distinguish different simulation methods, the X in X-in-the-loop (XiL) is substituted by model (MiL), software (SiL), hardware (HiL) or vehicle (ViL): [40, 44, 56, 66, 183, 184, 187, 188, 191–197][50]

- MiL: Algorithms modeling the system and its intended functionalities are tested in a fully virtual environment.
- SiL: A realized software code is tested in a simulation environment that potentially takes computational performance constraints of the not yet realized hardware into account. The vehicle, the drivers and the environment are virtually simulated. A SiL can also be combined with real measured input data.
- HiL: Parts of the system (e.g. specific control units) are physically realized and combined with virtual representations of the remaining vehicle. Partly the vehicle, the drivers and the environment are simulated virtually. A HiL can be combined with real measured input data.
- ViL: The vehicle is realized but parts of the environment such as infrastructure and other traffic participants are simulated virtually. An interface at the sensor level provides simulated signal responses from the virtual environment. The vehicle's reactions derived from the simulated sensor signals (and/or the driver) are executed on a test stand or on a

[50] The large number of publications addressing simulation methods for ADSs highlight the research activity on simulation based testing.

proving ground. For example, a ViL simulation enables to move a real vehicle through a virtual construction site or to safely test AEB systems.

XiL simulations are also listed as possible test methods in ISO 26262 and the Code of Practice [56, 174]. For more details on XiL we refer to [195, 198–200].

The main advantages of simulation methods are that they can be employed early in the development [44], they are reproducible [44, 56, 183, 191, 193, 196], they are inexpensive once established [196], they allow to test a large number of test cases virtually with limited effort [44, 56, 66, 183, 191, 201], component failures / errors can be injected to evaluate system behavior in case of failures [56, 183, 193], and they are safe [44, 56, 193, 196, 201]. The reference truth is known in a virtual environment. The main disadvantage of simulations is the simplification of reality by reducing the complexity of a problem with models [44, 191, 193].

A prerequisite to demonstrate *perception reliability* by means of simulation is that the simulations are sufficiently accurate [40, 44, 66, 191, 193, 197, 201]. It is reminded that the goal of an ADS's safety validation is to demonstrate Eq. (3.2). *Perception reliability* enters Eq. (3.2) through Eq. (3.3). Sufficiently accurate therefore means that an estimation of λ_{per} with simulation is only deviating from the true perception error rate with an acceptable estimation error. To be able to demonstrate Eq. (3.2), the simulation estimation error on λ_{per} has to be approximately one order of magnitude smaller than the risk acceptance criteria discussed in Section 3.1.1. This illustrates that demonstrating the validity of simulation methods to be used for the purpose of an ADS's safety validation is at least as hard as directly demonstrating the safety of an ADS (see Section 3.1.2) [66].

Setting up accurate simulations for *perception reliability* is particularly challenging [71, 191]. To become a valid simulation that accurately estimates λ_{per}, one would need to know a priori which context variables and environmental constellations could potentially cause perception errors to include them in the simulation (see Section 3.1.4) [191]. Additionally one would need to be able to accurately model the environment and the interactions of an ADS's perception with the environment [191, 193]. Discrepancies between simulated and corresponding real sensor data in [197] as well as discrepancies of camera based object detections and state estimates (e.g. object distance) in a simulated image compared with the corresponding real image exemplarily demonstrate these challenges [89, 193].

While accurate simulations certainly can be realized for some effects, it is challenging to accurately simulate all relevant effects. Even if one was able to devise a valid simulation method to assess *perception reliability*, the computational costs to achieve this high degree of validity might be

prohibitive [191]. According to [66, 191] no simulation methods exists today that allow to *solely* demonstrate ADS safety and with it *perception reliability* by simulation.

It is concluded that simulation methods are an important ADS development tool to improve the quality of an ADS, especially in early phases of the V-Model process [183, 196]. Simulations are an opportunity to verify compliance with specified requirements [191, 192], e.g. requirements on environment perception. Real software code can be implemented in the simulations. A simple solution to account for changes in the perception system is for instance to freeze the system's hardware at one point, collect (physical) sensor raw data once, and implement a SiL tool chain that represents the implemented algorithms and timing behavior of the sensors to further process the sensing raw data. Whenever the (sensor or fusion) software is updated, a new SiL can be run to obtain test data with the most recent software version.

Simulations allow to optimize an ADS's sensor architecture and the design of individual sensors, thereby increasing *(sensor) perception reliability* during system development. At present, simulation methods however do not allow to learn the *perception reliability* for an ADS's safety validation at the end of the system development, because of the involved simplifications of reality. It is furthermore not clear if simulation methods will become sufficiently realistic in the near future to allow for an ADS's safety validation by *application of simulation alone*.

Controlled Field Tests on Proving Grounds

Due to the limitations of virtual simulations, each ADS functionality is still tested in reality [44, 187, 191, 194]. One option is to execute controlled field tests on proving grounds with an integrated prototype according to test cases described in scenario catalogues [44, 56, 64, 66, 141, 186]. The test execution on proving grounds requires to control the relevant test scenario's parameters [55]. The aim is to successfully pass all test cases for a certain number of times, see e.g. [56]. To enable reproducibility of the test cases without risk for the involved personal, slap cars, driving robots[51] and dummy pedestrians are used [55, 141, 186, 190, 202].

As a main advantage, controlled field tests are more realistic than virtual simulations [55]. They are to a certain degree reproducible, safe and allow to inject faults for an evaluation of system behavior under failures [44, 56]. A practical limitation is that not all scenario parameters such as snowfall, spray water or fog can be controlled easily. Also, for safety reasons, testing the ADS in critical situations involves simplifications such as slap cars [194]. A slap car has different

[51] A driving robot is not to be confused with an ADS because it does not react to the environment. Instead, it is a robot that controls a vehicle in analogy to a human driver by physically actuating the acceleration and brake pedals as well as the steering wheel in a predefined way. A driving robot allows to precisely program potentially safety critical driving maneuvers. [190]
Alternatively, one can also pre-program a path in a test vehicle if equipped with suitable actuation for driverless maneuvering.

properties than a real car, e.g. it has different optical reflection properties and a different radar cross section [141]. Measures to mitigate these differences are taken [141, 190], e.g. prismatic reflectors are installed on slap cars to adjust the radar cross section [141]. Nevertheless, the involved simplifications of reality potentially lead to a different behavior of an ADS's environment perception. In comparison with field tests in real traffic, the variability of the driving situations on a proving ground is limited because the situations are artificially generated [44, 56]. In real traffic, for example, the variability in potential traffic participants is larger (e.g. different sizes, shapes and colors of cars or trucks).

In principle, controlled tests on proving grounds allow to learn *perception reliability*, because the real behavior of an ADS's environment perception is observed. On proving grounds, it is comparatively easy to set up a reference truth for observing perception errors, e.g. by equipping the test vehicles with differential GPS (DGPS) [190, 194, 195]. Tests on proving grounds are particularly well suited to demonstrate the absence of systematic perception errors[52]. Limitations of testing environment perception on proving grounds are simplifications concerning the selection of relevant traffic participants (e.g. dummy pedestrian) and the difficulty of testing all relevant situations (e.g. limited variability of relevant objects). The latter point is related to the limitations of scenario based testing, as discussed in Section 3.3.1. Furthermore, instead of qualitatively evaluating each test case as either passed or failed a, it is preferable to explicitly demonstrate Eq. (3.4), which requires to quantify the perception failure rate λ_{per}. It is challenging to generate a quantitative statement on λ_{per} by means of controlled field tests on proving grounds.

Field Tests in Real Traffic

Field tests in real traffic are performed as a final means of safety validation under realistic conditions before an automated driving functionality is released [35, 40, 44, 56, 61, 64, 66, 186, 187, 193, 194]. Either test cases described in scenario catalogues occur by chance in the field, the test cases are actively sought (e.g. with an event based field test approach, which aims at restricting the test to critical or relevant situations [61]), or most commonly, the field test is performed for a predefined number of test kilometer to demonstrate a low system failure probability [35, 44, 54, 56, 61, 66, 141, 187].

To ensure public safety, field tests are typically performed at the end of the development cycle, after conducting controlled field tests on proving grounds [44, 66, 194]. For ethical reasons, one cannot actively generate safety critical situations in the field to test e.g. AEB [56]. Exemplarily, an AEB functionality is validated with 2 Mio km of driving in [64]. In [203], the risk of ADAS

[52] A systematic error is defined as a: "failure related in a deterministic way to a certain cause, that can only be eliminated by a change of the design or of the manufacturing process, operational procedures, documentation or other relevant factors" [62]. In contrast, a random error would be related to a certain cause in a probabilistic way.

equipped vehicles compared with driver only vehicles is studied in extensive field tests. Moreover, large scale naturalistic driving studies are conducted to identify relevant critical situations and corresponding human behavior under real life conditions [204, 205], which can be used to develop ADSs and as a benchmark for ADS performance.

The central advantage of field tests in real traffic is that test conditions are most realistic [44, 55, 56, 201]. Field tests automatically lead to a high variability of driving situations and a variability of relevant objects (e.g. different sizes, shapes, colors and relative position of cars or trucks) [55, 66]. Additionally, field tests allow to identify unknown test cases [55, 61, 186, 191, 201]. By recording relevant unknown critical situations, scenario catalogues can be updated [61, 186]. One is then able to verify the ability and specifications of an ADS to handle these situations in controlled field tests or by means of simulation [186]. Challenges are to conduct the test representatively [40, 54] and to set up a reference truth for the identification of perception errors (see Section 3.1.4). Field tests are not reproducible, are hard to control (i.e. actively trying to test certain situations) and are cumbersome because of the large amount or required testing (see Section 3.1.2) [44, 55, 56, 66, 201].

In combination with demonstrating driver controllability in scenario based testing, field tests allow to demonstrate an acceptable risk of ADAS with a manageable test effort (e.g. 2 Mio km to demonstrate a sufficiently low false positive rate for an AEB functionality [64]). Without the assumed driver controllability, empirically demonstrating an acceptable risk for *higher levels of driving automation* is not practical due to the approval trap (see Section 3.1.2). According to Section 3.1.4, the approval trap also applies to a demonstration of *perception reliability*. Hence, demonstrating an ADS's SOTIF and *perception reliability* seems not directly possible with field tests alone due to the impractically large required test effort. Implementing scenario based testing in field tests (e.g. an event based field test [61]) has similar limitations when applied to ADSs with *higher levels of driving automation* as already discussed in Section 3.3.1. Aside of ethical concerns, restricting and actively testing predefined critical situations in the field neglects that for ADSs with *higher levels of driving automation*, a seemingly uncritical situation can become critical due to perception deficiencies.

Summary

Testing is used as a development tool to increase the quality and safety of a system and as a means of validating if an acceptable system safety has been achieved [71, 141]. All currently established test methods are useful development tools for ADSs. Most importantly, field tests in real traffic and partly controlled field tests on proving grounds are utilized to validate system safety. However, none of these methods are in their current form sufficient to formally demonstrate *perception reliability* and with it the safety of an ADS with *higher levels of driving automation*.

4 Environment Perception Reliability

According to [35], an important step towards addressing the gap in ADS safety validation methods is to define a metric that allows to formally assess the performance of an ADS. We argue that this metric should be the risk of an ADS, here expressed as the rate λ_{sys} of fatal accidents in Eq. (3.3). The risk of an ADS has to comply with risk acceptance criteria (see Section 3.1.1). Further, [31] state that a lack of perception metrics makes it difficult to assess the performance of the environment perception. [35] proposes to individually introduce metrics for the perception, cognition and the actuation of an ADS. In this manuscript, the corresponding individual metrics are according to Section 3.1.3 the respective failure rates of the perception λ_{per}, the automated driving function module λ_{funct} and the actuation λ_{act}.

The goal of a *perception reliability* analysis is to estimate λ_{per} for a demonstration of Eq. (3.4). It is not straightforward to directly assess λ_{per} due to the difficulties discussed in Section 3.1.4. Directly assessing λ_{per} requires a comprehensive definition of safety-critical perception errors or alternatively necessitates to test the perception and automated driving functionality jointly. In particular, the difficulty of defining safety-relevant perception errors, which depends e.g. on the path planning and the actual driving situation, makes it hard to assess λ_{per} directly.

To address these challenges, we introduce additional *perception reliability* metrics in Section 4.1 that allow to assess the performance of environment perception independent of the automated driving functionality and actual driving situation. Based on these *perception reliability* metrics and on targets for λ_{per}, we derive individual *sensor perception reliability* requirements in Section 4.2. Finally, we estimate the test effort to validate the derived *perception reliability* requirements in Section 4.3.

4.1 Environment Perception Reliability Metrics

This Section is based on our publications in [39, 161] and is partly taken from our publications in [39].

In Sections 4.1.1-4.1.4 we introduce *perception reliability* metrics accounting for environment perception uncertainties [40]. In a second step, the *perception reliability* metrics are related to λ_{per} in Section 4.1.5.

The basis for describing *perception reliability* with metrics are the three fundamental types of uncertainties in environment perception [60, 67], which are schematically illustrated in Figure 4.1:

- Existence uncertainty: uncertainty on whether existing objects are detected (true positive TP vs. false negative FN) and on whether non-existing ghost objects are wrongly indicated (false positive FP vs. true negative TN).
- Classification uncertainty: uncertainty on the semantic types of detected objects (e.g. car, truck, pedestrian, bicycle, motorcycle, static obstacle, traffic sign, etc.).
- State uncertainty: uncertainty on the state of physical quantities of detected objects (object position, object velocity, object acceleration, object size, etc.).

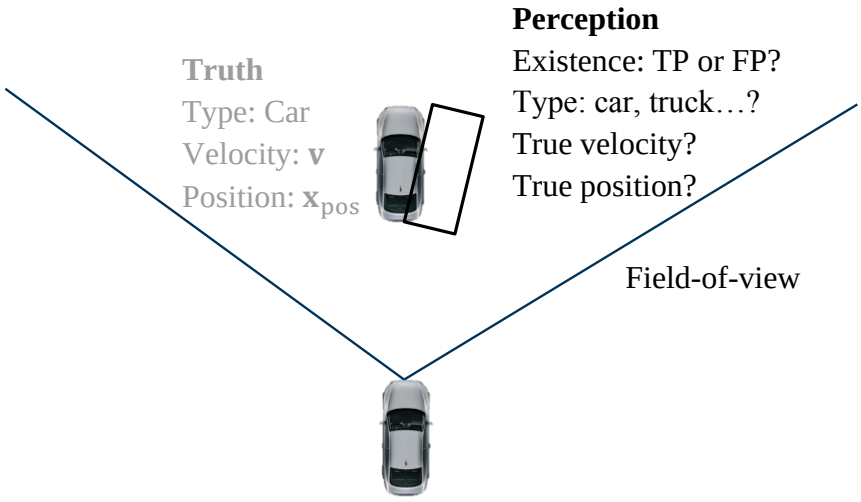

Figure 4.1 Schematic illustration of the existence, classification and state uncertainty due to noisy and potentially deficient environment sensing. The black bounding box is the output of the perception. Without knowledge of the ground truth, it is uncertain whether the indicated black object exists in reality (true positive TP or false positive FP?), which object type it is (car, truck, etc.?) and what state it is in (velocity, position, etc.?). Taken from [39].

Each of these uncertainties can lead to safety-critical perception errors and hence contribute to λ_{per}. Independent of the safety-criticality of a specific perception error, the overall perception performance is quantified by metrics for these fundamental uncertainties [161]. We present the *perception reliability* metrics on the level of the perception module (= fused environment model), however they equally apply to individual sensors if they provide information in the respective uncertainty domain (object detection, object classification and object states). On the grounds of the generic environment sensor model in Figure 2.4, the *perception reliability* metrics are used for all sensing technologies (radar, camera and lidar) on the object data level, i.e. each sensing technology is described with these metrics.

A type of uncertainty not illustrated in Figure 4.1 is association uncertainty [59], which occurs in a redundant multi-sensor system with sensor data fusion. Association uncertainty is the uncertainty on which sensor object detections are due to the same (real) object, represented as a track in sensor data fusion (i.e. object to track association uncertainty) [59]. Explicitly modeling perception errors due to sensor data fusion is not in scope of this thesis. Moreover, as explained in Section 1.3, we exclude situation awareness in the definition of the metrics.

4.1.1 Existence Uncertainty

Signal processing in a single sensor proceeds with a variety of algorithms from the signal over raw data to object data (see Figure 2.4). The data of multiple sensors is then combined in a sensor data fusion, see Section 2.1.5.

To exemplarily illustrate one reason behind existence uncertainty, consider the hypothetical lidar signal in Figure 4.2. A raw data point is detected if the signal intensity exceeds a detection threshold [40], which is a setting of the sensor. With threshold 1, a false negative (FN) error occurs, with threshold 2 a true positive (TP) detection results. Threshold 3 results in a true positive (TP) detection as well as a false positive (FP) error, if the sensor is designed to detect multiple signal peaks as objects (multi-target capability [102]). This example illustrates how the detection threshold is a compromise between FP and FN errors [40]. Depending on the full signal processing chain and processing algorithms, these potential errors in raw data can in some cases in combination with other perception errors lead to perception deficiencies in the (fused) environment model.

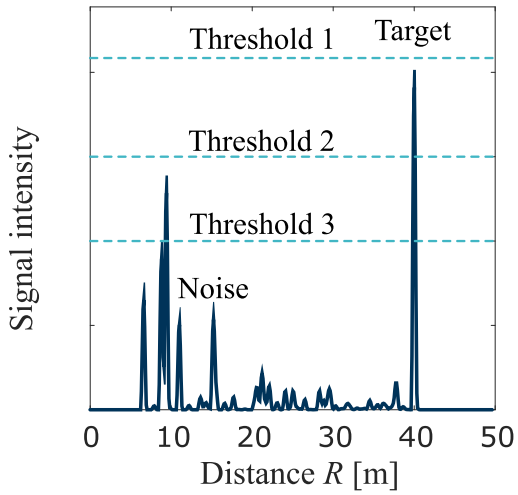

Figure 4.2 Example of how perception errors arise in the existence uncertainty domain with a lidar. At 40 m a real object is present while the signal in the range < 40 m is due to noise. Depending on the detection threshold, the real object is indicated or not (true positive vs false negative) and the noise is wrongly indicated as an object or not (false positive vs true negative). Adapted from [105].

In the following we model the perception probabilistically because of the large number of context variables **E** (see Section 3.1.4), due to system complexity and due to inherent uncertainties of environment perception [40]. Particularly, the detection performance in the existence uncertainty domain is described with the theory of signal detectability [206–208]. The task of the perception is to detect (event $D = 1$) an existing object (event $O = 1$) and not indicate (event $D = 0$) non-existing objects (event $O = 0$). As illustrated with the confusion matrix in Figure 4.3, and as explained with the example of Figure 4.2, the output of the perception in the context of existence

uncertainty is therefore either a true positive (TP), a false negative (FN), a false positive (FP) or a true negative (TN).

	O=1	O=0
$D = 1$	TP	FP
$D = 0$	FN	TN

Figure 4.3 Confusion matrix for the existence uncertainty. $O = 1$: object present; $O = 0$ no object present; $D = 1$: detection; $D = 0$: no detection; TP: true positive; FN: false negative; FP: false positive; TN: true negative. Taken from [157].

The confusion matrix in Figure 4.3 represents a binary problem interpretation: Either the perception output is a detection or not, and either the underlying truth is an object being present or not. In an environment model, multiple objects are present. Therefore one needs to find a suitable binary representation of the environment model in order to apply the confusion matrix in Figure 4.3. An example is to evaluate a limited area of the field-of-view (FOV), in which $O = 1$ is the event of at least one object being present. The area should be limited such that the case of more than one object being present can be neglected. Another possible binary interpretation is to analyze detections of the closest preceding vehicle in the driving path up to a certain distance. Either an object is detected in the driving path or not.

With the confusion matrix in Figure 4.3, the perception performance in the context of existence uncertainty is described with a probability of detection POD and a probability of false alarm PFA [206–208]. POD is defined as the conditional TP probability[53]:

$$\mathrm{POD} = \Pr(D = 1 | O = 1) \tag{4.1}$$

PFA is defined as the conditional FP probability:

$$\mathrm{PFA} = \Pr(D = 1 | O = 0) \tag{4.2}$$

By modifying the settings (sensor detection thresholds, filters, gates etc.) of the perception module, the POD and PFA can be varied, which leads to a receiver operating characteristic (ROC) curve [206–208].

[53] An assessment of *perception reliability* is for instance interested in the POD. It is pointed out, for an automated driving functionality to make a decision, the a posteriori probability of object or track existence given an indication of an object by the environment perception (and given additional information) is needed. The POD is $\Pr(D = 1 | O = 1)$, while the a posteriori probability of object existence is (on a simple basis) $\Pr(O = 1 | D = 1)$. The POD is required to accurately estimate $\Pr(O = 1 | D = 1)$, that is, the POD is an input quantity to fusion algorithms (see Section 2.2.2).

In practice, an object in the environment model is associated with an often heuristically derived existence measure p [60, 67]. ROC analysis [206–208], which is crucial for representing existence uncertainties, is explained next with the example of the existence measure p and the detection threshold t_t in analogy to [209]. p is usually normalized. Alternatively, a classifier such as a CNN outputs a probability of object existence p in a binary detection problem. To count as a valid object detection, p has to exceed a detection threshold t_t. The implemented threshold t_t is a design setting [40].

Conditional on an object being present $O = 1$ or not $O = 0$, p is itself a sample from a population, described with a probability density function (PDF) $f_{P|O}(p|O)$. Figure 4.4a) shows examples of $f_{P|O}(p|O = 1)$ and $f_{P|O}(p|O = 0)$. For a specific t_t, the POD is:

$$\mathrm{POD} = \int_{t_\mathrm{t}}^{1} f_{P|O}(p|O = 1)\, dp \tag{4.3}$$

and the PFA:

$$\mathrm{PFA} = \int_{t_\mathrm{t}}^{1} f_{P|O}(p|O = 0)\, dp \tag{4.4}$$

These integrals correspond to the shaded areas in Figure 4.4a). Plotting the POD over the PFA for different t_t in Eqs. (4.3) and (4.4) leads to a ROC curve, as shown in Figure 4.4b).

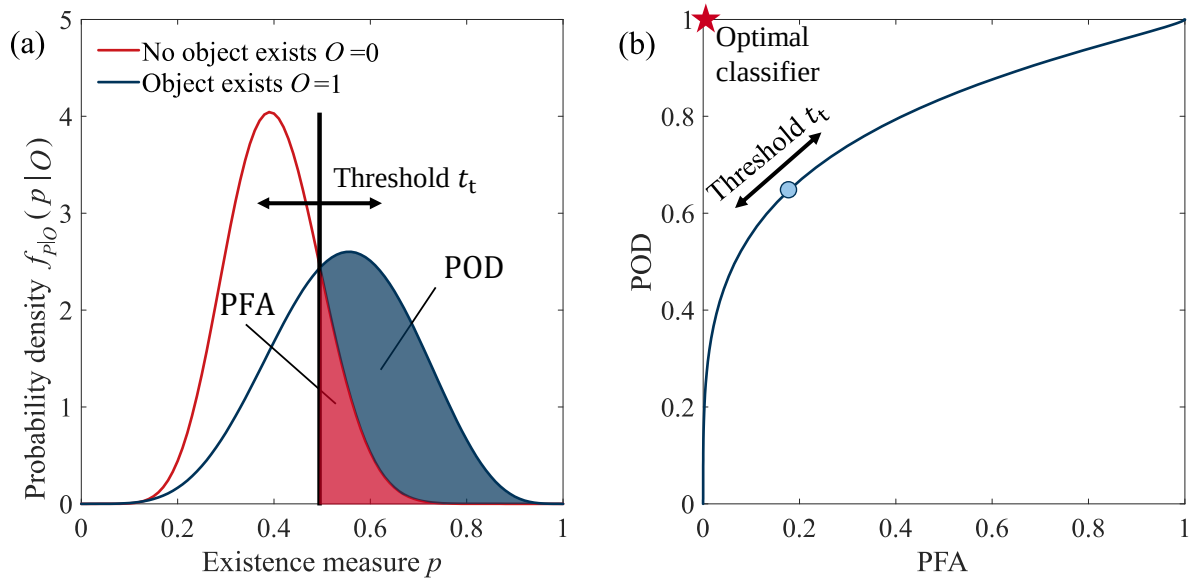

Figure 4.4 **Receiver operating characteristic (ROC) analysis: In (a) the threshold t_t is modified, leading to a different POD and PFA. POD and PFA are the respective areas under the PDFs $f_{P|O}(p|O = 1)$ and $f_{P|O}(p|O = 0)$, as illustrated by the blue and the red shaded areas. In (b), the resulting POD over PFA is shown, which leads to the ROC curve. Each threshold t_t corresponds to a specific point on the ROC curve in (b). A perfect classifier is the red star in (b) with POD=1 and PFA=0. In analogy to Fig.1 in [209].**

The overall detection performance of the perception module w.r.t. existence uncertainties is described with the ROC curve, while one POD and PFA pair are one operating point on this curve for a specific setting. For example, the blue point in Figure 4.4b) on the ROC curve corresponds to the threshold in Figure 4.4a).

This example shows how the POD and PFA are related to the ROC curve. Lowering the detection threshold leads to a larger POD but also to an increased PFA. Hence, sensor settings such as detection thresholds allow to tradeoff FP and FN errors. The inherent performance of a receiver (e.g. a sensor) is thus determined by the two PDFs in Figure 4.4a). Altering the low level signal processing of a single sensor might modify the two PDFs in Figure 4.4a), but the effect on overall performance must be evaluated by considering both, POD and PFA.

Ultimately, the interest is in the frequencies of FP and FN errors for a specific operating characteristic (point on the ROC curve). The probability of a FN error $\Pr(\mathrm{FN})$ in a specific area of the FOV depends on the probability $\Pr(O = 1)$ of at least one object being present in that particular area and on POD:

$$\Pr(\mathrm{FN}) = \Pr(D = 0 \cap O = 1) = (1 - \mathrm{POD}) \cdot \Pr(O = 1) \tag{4.5}$$

Likewise, the FP probability $\Pr(\mathrm{FP})$ in a particular area of the FOV depends on the probability of no object being present in that area $\Pr(O = 0)$ and on PFA:

$$\Pr(\mathrm{FP}) = \Pr(D = 1 \cap O = 0) = \mathrm{PFA} \cdot \Pr(O = 0) \tag{4.6}$$

The TP and TN probabilities are derived in the same manner.

Sometimes POD and PFA are termed TP rate and FP rate[54] [208]. To avoid confusion, we explicitly distinguish between probabilities and rates. The rates are the expected number of occurrences per time interval (or per distance interval) and the probabilities are the expected number of occurrences in a discrete trial. For the FP rate λ_{FP}, the following identity holds:

$$\lambda_{\mathrm{FP}} \approx \mathrm{PFA} \cdot \Pr(O = 0) \cdot \frac{1}{\Delta \mathrm{t}} \tag{4.7}$$

where $\Delta \mathrm{t}$ is the duration of one trial. Similarly, the FN rate λ_{FN} is defined by:

$$\lambda_{\mathrm{FN}} \approx (1 - \mathrm{POD}) \cdot \Pr(O = 1) \cdot \frac{1}{\Delta \mathrm{t}} \tag{4.8}$$

How to define $\Delta \mathrm{t}$? It is assumed that the environment perception is updated cyclically with a time of t_{cycle}. Usually, the automated driving functionality only reacts if an object is indicated over

[54] This terminology is however misleading because rates imply an underlying continuous reference (e.g. time) while a probability refers to a frequency in the limit of discrete trials. Hence, we consider a FP rate λ_{FP} not to be identical but related to PFA.

multiple cycles, i.e. the perception is validated with a multi-cycle heuristic [29, 59, 60, 106]. Therefore, an error only becomes safety-critical if it persist for a critical time t_{crit}, which is a multiple of t_{cycle} and is derived from the implemented multi-cycle heuristics. $\Delta\mathrm{t}$ is set equal to t_{crit}.

4.1.2 State Uncertainty

State uncertainties arise because of a variety of noise sources, which are exemplarily discussed in Section 2.1 [40]. Noise leads e.g. to uncertainty in a raw distance measurements [210–212], which together with imperfect processing algorithms translates to uncertainty in state quantities of object data.

In Section 2.2.2, it was described how the Bayes filter is used to estimate object states given sensor observations. Ultimately, the estimates of object states are the basis for the path planning[55]. Let $\mathbf{X}$ be the vector of true object states, with elements as for instance as in Table 2.2. $\widehat{\mathbf{X}}$ is the estimated state of an object with true state $\mathbf{X}$. With an additive error model, the *perception reliability* of state uncertainties is described by the joint PDF $f_{\Delta\mathbf{X}}(\Delta\mathbf{x})$ of the deviations $\Delta\mathbf{X}$ between $\widehat{\mathbf{X}}$ and $\mathbf{X}$:

$$\widehat{\mathbf{X}} = \mathbf{X} + \Delta\mathbf{X} \tag{4.9}$$

A common choice is to model $\Delta\mathbf{X}$ with a normal model, i.e. $f_{\Delta\mathbf{X}}$ is the multivariate normal PDF. Under a multivariate normal model, if no systematic errors are present, the mean vector is zero. In this case, the state uncertainty is fully described with a covariance matrix. If further, the elements in $\Delta\mathbf{X}$ are uncorrelated, the state uncertainty is fully described with the standard deviations of the elements in $\Delta\mathbf{X}$. In the general case, $f_{\Delta\mathbf{X}}$ is not multivariate normal and one has to determine $f_{\Delta\mathbf{X}}$ on the basis of data.

We point out that Eq. (4.9) is only one way of describing state uncertainties. An alternative to the additive error model in Eq. (4.9) is for instance a multiplicative error model:

$$\widehat{\mathbf{X}} = \mathbf{X} \cdot \Delta\mathbf{X} \tag{4.10}$$

$\Delta\mathbf{X}$ again follows a PDF $f_{\Delta\mathbf{X}}$ that must be determined from data.

The most important aspect of state uncertainties for *perception reliability* are extreme but rare $\Delta\mathbf{X}$. Therefore, the main concern in quantifying state uncertainty is to accurately estimate the tails of $f_{\Delta\mathbf{X}}$. A normal distribution for $\Delta\mathbf{X}$ is not necessarily an accurate model of the tails of $\Delta\mathbf{X}$. An

[55] Also, the uncertainty in the estimated states (e.g. a credible region) is relevant for the path planning. The accuracy in the uncertainty estimation is not considered here.

alternative to describe state uncertainties with focus on extreme $\Delta\mathbf{X}$ is provided by extreme value theory [213].

For instance, one modeling option in extreme value theory is based on the block-maxima method. With the block-maxima method, the maxima of an independent and identically distributed scalar ΔX is extracted from a fixed number of data points, for instance, from all data in a given time interval (e.g. 5 min):

$$Y_{\mathrm{n}} = \max[\Delta X_1, \dots, \Delta X_{\mathrm{n}}] \tag{4.11}$$

n is the number of data points corresponding to the time interval of chosen length and Y_{n} is the corresponding maximum in this time interval. Y_{n} follows an extreme value distribution such as the generalized extreme value distribution [213]. The advantage of this modeling approach is the focus on safety-relevant extreme values of ΔX.

All given modeling examples are suitable ways of describing *perception reliability* in the state uncertainty domain. The goal is to determine the (joint) PDF of the deviations between the true and the estimated states.

4.1.3 Classification Uncertainty

Typically, each object in an environment model is classified (see Section 2.2.2) and associated with semantic information. Object classes are for instance car, truck, pedestrian and cyclist, the type of a traffic sign or the state of a traffic light. Wrong classifications potentially are perception errors.

Classification uncertainty is represented by extending the confusion matrix in Figure 4.3 with additional rows and columns [161]. This means, O is not binary but categorical. For instance, a categorical O has as states: No object present, car, truck, pedestrian, etc.. The classification uncertainty is then described with conditional probabilities in analogy to Eqs. (4.1)-(4.2), for instance, the conditional probability of classifying a pedestrian as a car. Table 4.1 is a generic example of a confusion matrix to describe classification uncertainty.

Table 4.1 **Exemplary illustration of a confusion matrix to describe classification uncertainty with 3 object classes. Consider for instance the following classes: $O = 0$: No object present; $O = 1$: car; $O = 2$: truck; $O = 3$: pedestrian. The classification performance is expressed in terms of conditional probabilities of indicating (event D) one of the objects under the respective truth (event O). The sum of each column is one. Taken from [39].**

	$O = 0$	$O = 1$	$O = 2$	$O = 3$
$D = 0$	$\Pr(D = 0 \mid O = 0)$	$\Pr(D = 0 \mid O = 1)$	$\Pr(D = 0 \mid O = 2)$	$\Pr(D = 0 \mid O = 3)$
$D = 1$	$\Pr(D = 1 \mid O = 0)$	$\Pr(D = 1 \mid O = 1)$	$\Pr(D = 1 \mid O = 2)$	$\Pr(D = 1 \mid O = 3)$
$D = 2$	$\Pr(D = 2 \mid O = 0)$	$\Pr(D = 2 \mid O = 1)$	$\Pr(D = 2 \mid O = 2)$	$\Pr(D = 2 \mid O = 3)$
$D = 3$	$\Pr(D = 3 \mid O = 0)$	$\Pr(D = 3 \mid O = 1)$	$\Pr(D = 3 \mid O = 2)$	$\Pr(D = 3 \mid O = 3)$

4.1.4 Higher Order Uncertainties

The *perception reliability* metrics evaluated over a limited amount of data (or time) are themselves not constant but follow a probability distribution because of the influence of stochastic context variables **E**, see also Section 3.1.4. This means, the *perception reliability* metrics evaluated with a limited amount of data are random variables themselves. In [161], we termed the uncertainty in the *perception reliability* metrics *higher order uncertainties*.

Figure 4.5 illustrates *higher order uncertainties* with the example of state uncertainties. The relevant context variable in this example is rainfall intensity I. Let the state uncertainties be described with the standard deviation $\sigma_{\Delta X}$ of ΔX. ΔX is the deviation between the estimated and the true state of a physical quantity of interest (e.g. x-coordinate of an object).

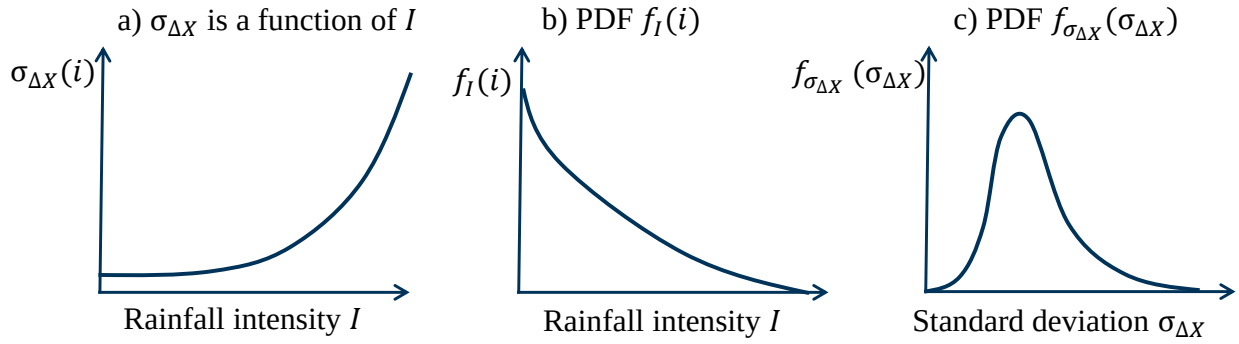

Figure 4.5 **Schematic illustration of higher order uncertainties with the example of state uncertainties. (a) The standard deviation $\sigma_{\Delta X}$ describing the state uncertainty is a function of rainfall intensity I. (b) Rainfall intensity is a random variable following the PDF $f_I(i)$. (c) Because of the relationships in (a) and (b), the standard deviation is itself a random variable following the PDF $f_{\sigma_{\Delta X}}(\sigma_{\Delta X})$. Adapted from [161].**

Figure 4.5a) shows that the variability in ΔX increases with the rainfall intensity, i.e. $\sigma_{\Delta X}$ increases with I. The rainfall intensity I is a random variable described with a PDF $f_I(i)$, schematically visualized in Figure 4.5b). Combining the functions in Figure 4.5a) and Figure 4.5b) leads to the

standard deviation $\sigma_{\Delta X}$ being a random variable described with a PDF $f_{\sigma_{\Delta X}}(\sigma_{\Delta X})$ as shown in Figure 4.5b). $f_{\sigma_{\Delta X}}(\sigma_{\Delta X})$ therefore describes *higher order uncertainties* in ΔX. The considerations in Figure 4.5 can be transferred to the existence and classification uncertainty domains.

Following Section 3.1.4 in analogy to Eq. (3.5), if the amount of data is large and representative, global *perception reliability* metrics are estimated from the data. Then, *higher order uncertainties* do not have to be specifically taken into account. If the *perception reliability* metrics are estimated from a limited amount of data, *higher order uncertainties* must be accounted for.

4.1.5 Relationship between the Reliability Metrics and the Perception Failure Rate

With the description of perception uncertainties by *perception reliability* metrics in the previous Sections, we define perception performance metrics separately from an automated driving functionality. This leads to the advantage that during the ADS development, one is able to estimate the *perception reliability* metrics in a first step independently from e.g. the path planning in the automated driving function module.

To be able to judge whether specific values of the *perception reliability* metrics are acceptable for an automated driving functionality, they are in a second step related to the safety-critical failure rate λ_{per} of the perception module, which depends on the specific automated driving functionality under consideration. We discuss how the *perception reliability* metrics relate to λ_{per} with the example of existence uncertainties, i.e. with FN and FP errors. FP and FN sensor errors are among the most safety-critical perception error types [42].

The metrics introduced in Section 4.1.1 refer to specific areas of the FOV. Let the FOV be partitioned into J sub-areas to enable the binary interpretation of Figure 4.3 and let the index $j = 1, \dots, J$ identify a specific area. For instance, the FOV is partitioned into equally sized cells as in Figure 4.6. In reality, in each area only a fraction of the FN and FP perception errors are safety-critical, i.e. would lead to an accident. In case of FN it is:

$$\lambda_{\mathrm{FN}_{\mathrm{crit},j}} = \lambda_{\mathrm{FN},j} \cdot p_{\mathrm{FN},j} \tag{4.12}$$

where $\lambda_{\mathrm{FN},j}$ is the FN error rate of the perception module in the j^{th} area of the FOV according to Eq. (4.8), schematically illustrated in Figure 4.6b). $p_{\mathrm{FN},j}$ is the fraction of all FN errors that according to the perception error definition in Section 3.1.4 are safety-critical in area j, as schematically presented in Figure 4.6a). $\lambda_{\mathrm{FN}_{\mathrm{crit},j}}$ is the resulting rate of safety-critical FN errors of the perception module in the j^{th} area. $\lambda_{\mathrm{FP}_{\mathrm{crit},j}}$ is defined analogously:

$$\lambda_{\mathrm{FP}_{\mathrm{crit},j}} = \lambda_{\mathrm{FP},j} \cdot p_{\mathrm{FP},j} \tag{4.13}$$

For simplicity, it is not distinguished between p_{FN} and p_{FP} in Figure 4.6a). Eq. (4.12) for instance multiplies the cell values in Figure 4.6a) with the corresponding cell values in Figure 4.6b).

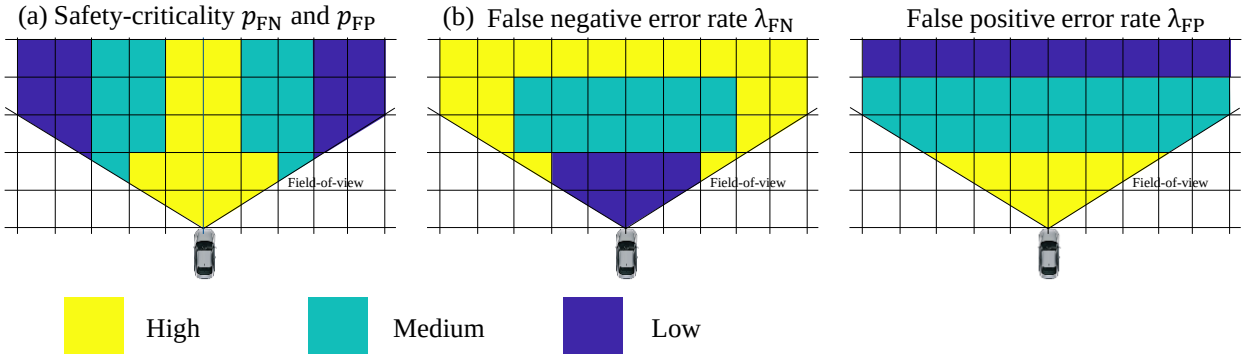

Figure 4.6 Exemplary schematic illustration of partitioning the FOV into smaller areas. (a) the safety-criticality of FN and FP errors, p_{FN} and p_{FP}, over different areas of the FOV. (b) The FN error rate λ_{FN} in different areas of the FOV. (c) The FP error rate λ_{FP} in different areas of the FOV. Taken from [39].

Under the conservative assumption of mutually exclusive safety-relevant FN and FP errors in the different cells, the contribution of all areas to λ_{per} is the sum over Eq. (4.12) and Eq. (4.13) w.r.t. all areas. This is equivalent to interpreting the different areas in the FOV as a series system [153, 154]:

$$\lambda_{\mathrm{per}} \approx \lambda_{\mathrm{FN_{crit}}} + \lambda_{\mathrm{FP_{crit}}} = \sum_{j=1}^{J} \lambda_{\mathrm{FN_{crit},}j} + \sum_{j=1}^{J} \lambda_{\mathrm{FP_{crit},}j} \tag{4.14}$$

where $\lambda_{\mathrm{FN_{crit}}}$ is the safety-relevant FN error rate and $\lambda_{\mathrm{FP_{crit}}}$ the safety-relevant FP error rate. The contribution of perception errors due to the other uncertainty domains are for simplicity not explicitly considered in Eq. (4.14). However, other error types such as large object position deviations (state uncertainty) or association errors in the sensor data fusion could also be defined as FP and FN errors.

A challenge is to quantify the fractions of safety-critical perception errors p_{FN} and p_{FP} for a specific automated driving functionality. The quantification of p_{FN} and p_{FP} has to consider the path planning and situation interpretation. One possible solution strategy to this challenge is to apply heuristics to define the safety-criticality to some degree separately from the path planning and a given situation. An exemplary heuristic is to consider FN and FP errors in a limited area around the driving path as safety-relevant if the error is present for a time $\geq t_{\mathrm{crit}}$. All other FNs and FPs are neglected.

In terms of Figure 4.6a) this heuristic is realized by setting $p_{\mathrm{FN}} = 1$ and $p_{\mathrm{FP}} = 1$ in the area around the driving path, i.e. schematically in the area with high safety relevance in Figure 4.6a) (yellow). In all other areas it is $p_{\mathrm{FN}} = 0$ and $p_{\mathrm{FP}} = 0$. Similar heuristics could be defined for other error

types and uncertainties, e.g. position deviations that exceed an upper bound for a time $\geq t_{crit}$ in the driving path are safety critical.

An alternative to the heuristics for the safety-criticality is to evaluate p_{FN} and p_{FP} in depth (e.g. by simulation) for different automated driving functionalities. These approaches allow to demonstrate the safety of different automated driving functionalities in an ADS by combining *perception reliability* metrics (e.g. $\lambda_{\mathrm{FP},j}$) with the safety-criticality depending on the specific automated driving functionality (e.g. p_{FN}). Then, the *perception reliability* metrics only have to be learned once for a given perception architecture. Without loss of generality we omit the index j of the different areas in the following.

4.2 Sensor Perception Reliability Requirements: Addressing the Approval Trap

Section 4.1.5 approximates λ_{per} with Eq. (4.14). An inductive (empirical) demonstration of an acceptable λ_{per} is however subject to the approval trap (see Figure 3.3). To address the approval trap, we outline an alternative deductive strategy to demonstrate λ_{per}. This Section is based on our publications [39, 214] and is partly taken from our publication [39].

A deductive demonstration of λ_{per} takes *perception reliabilities* of components /sub-functionalities (i.e. individual sensors) of the perception module into account. Similar to Section 3.1.3, the perception module is decomposed into its sub-functionalities and individual sensors. The *perception reliability* is then demonstrated on the level of individual sensors, i.e. by validating *sensor perception reliability*.

To validate *perception reliability* at the sensor level, *sensor perception reliability* requirements are derived. Because FP and FN errors are among the most important perception error types, we only consider existence uncertainty in the following. Starting from a global risk acceptance criterion in Section 3.1.1, acceptable rates $\lambda_{\mathrm{TLS}_{\mathrm{per}}}$[56] are defined for the perception module according to Eq. (3.3). The aim is to demonstrate $\lambda_{\mathrm{per}} \leq \lambda_{\mathrm{TLS}_{\mathrm{per}}}$. Thereafter, from $\lambda_{\mathrm{TLS}_{\mathrm{per}}}$, acceptable values $\lambda_{\mathrm{TLS}_{\mathrm{FN,crit}}}$ and $\lambda_{\mathrm{TLS}_{\mathrm{FP,crit}}}$ are derived with Eq. (4.14) for the perception module. We then derive requirements on FP and FN error rates, $\lambda_{\mathrm{TLS}_{\mathrm{FN},i}}$ and $\lambda_{\mathrm{TLS}_{\mathrm{FP},i}}$ of individual sensors $i = 1, \ldots, n$ from acceptable values of $\lambda_{\mathrm{TLS}_{\mathrm{FN,crit}}}$ and $\lambda_{\mathrm{TLS}_{\mathrm{FP,crit}}}$ of the perception module. n is the number of redundant sensors with overlapping FOV and i identifies a specific sensor. Demonstrating

[56] To highlight the difference between requirements and actual rates, requirements are indexed with TLS (target level of safety). E.g. $\lambda_{\mathrm{TLS}_{\mathrm{per}}}$ is the requirement on λ_{per}.

perception reliability on the level of the individual sensors, i.e. on the level of λ_{FN_i} and λ_{FP_i} can help to overcome the approval trap.

4.2.1 Decomposition of Environment Perception: The k-out-of-n Vote

An exemplary functional block diagram (FBD) for a generic perception module is shown in Figure 4.7. The FBD is the basis for a decomposition of the perception module into its sub-functionalities and components (i.e. sensors). Different sensors, which in some parts of the FOV are redundant, collect information about the environment. This information is combined in a sensor data fusion (see Section 2.2.2) [57, 58, 110], which is here treated as a black box.

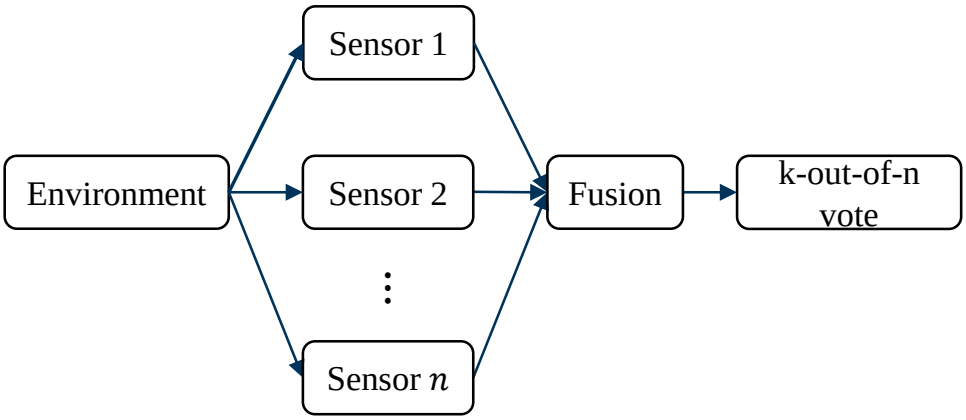

Figure 4.7 Exemplary functional block diagram representing the perception module. Adapted from [39].

The task of sensor data fusion is to associate object detections of different sensors to tracks and to filter the corresponding state estimates, i.e. object tracking (see Section 2.2.2) [57–59, 67]. After this step, a decision has to be made – prior to path planning – on whether an object track at the output of sensor data fusion is credible and serves as an input to the path planning. The decision of accepting an object track at the fusion's output may for instance be represented by a k-out-of-n vote [65, 156, 215], which means, an object track is the input to the path panning if at least k-out-of-n sensors indicate a tracked object (and if additionally, the object is indicated for some minimum number of discrete time steps by the respective k-out-of-n sensors). The k-out-of-n vote implies a k-out-of-n system. A k-out-of-n system is a widely used model to analyze the reliability of technical systems [150]. A special case of the k-out-of-n vote is a majority-voter with $k = \left\lfloor \frac{n}{2} + 1 \right\rfloor$, which is utilized in [42] to derive requirements on FP and FN sensor error rates.

The appropriateness of this simple interpretation of sensor data fusion depends on the object association algorithm and how the decision about object track existence is made in (or after) sensor data fusion. For instance, a probabilistic association algorithm such as the Joint Integrated Probabilistic Data Association (JIPDA, see Section 2.2.2) does not allow for the k-out-of-n representation because sensor object-to-track associations are probabilistic [67, 118]. In contrast,

the more common global nearest-neighbor association on the basis of Mahalanobis distance (see Section 2.2.2) [59, 67, 117] allows for the k-out-of-n representation, because sensor object-to-track associations are deterministic.

Note that one of the main sources of perception errors in the sensor data fusion are errors related to object-to-track associations (i.e. because of association uncertainties, see Section 2.2.2) [59]. In the following we neglect association errors but point out that many association errors can again be interpreted as FN and FP errors.

Regarding the fusion architecture, the k-out-of-n vote is an appropriate modeling approach for a decentralized fusion [58] of individual sensors' object detections (and their respective tracks). In contrast, the k-out-of-n vote does not appear to directly fit to a centralized sensor data fusion architecture [58] on the basis of raw data. One could however argue for a supervision of the fusion with a k-out-of-n voter based on individual sensors [42], which would again allow for the FBD in Figure 4.7. For these reasons, a k-out-of-n representation of sensor data fusion is used as the modeling approach in this work[57].

4.2.2 Deriving Sensor Perception Reliability Requirements

Sensor perception reliability requirements for each individual sensor are derived with the k-out-of-n voting approach in combination with statistical models for perception errors. It is here not distinguished between different sensors for the purpose of assigning reliability requirements. That is, each sensor i is assigned the same acceptable $\lambda_{\mathrm{TLS}_{\mathrm{FN},i}} = \lambda_{\mathrm{TLS}_{\mathrm{FN,sensor}}}$ and $\lambda_{\mathrm{TLS}_{\mathrm{FP},i}} = \lambda_{\mathrm{TLS}_{\mathrm{FP,sensor}}}$. $\lambda_{\mathrm{TLS}_{\mathrm{FN,sensor}}}$ and $\lambda_{\mathrm{TLS}_{\mathrm{FP,sensor}}}$ have a corresponding TLS on $\mathrm{POD}_{\mathrm{sensor}}$ and on $\mathrm{PFA}_{\mathrm{sensor}}$ according to Eqs. (4.7)-(4.8).

Two simple statistical models are presented to derive *perception reliability* requirements [39, 65, 156, 214]: the binomial model assuming statistical independence between sensor errors and the beta-binomial model, taking sensor error dependence into account. To describe the dependence in sensor errors with the beta-binomial model, the pairwise correlation coefficient of FP and FN errors among different sensors is introduced.

More sophisticated statistical models to explicitly distinguish between the FP and FN error rates in different sensors are proposed in Section 5.4. These detailed models are not suited for setting initial requirements due to the large number of free model parameters. Instead they should be

[57] "All models are wrong but some are useful" (Statistician George Box 1978). A more detailed representation of sensor data fusion likely requires the actual software code of sensor data fusion in combination with a simulation framework. Such a detailed simulation approach is probably difficult to set up in the beginning of the development of an ADS. In the words of mathematician Norbert Wiener (1945): "The best material model of a cat is another, or preferably the same, cat."

applied to learn and validate *sensor perception reliability*, and subsequently *perception reliability*, of a developed sensor system. Parts of Section 4.2.2 are taken and adapted from our publications in [39, 65, 156].

Statistically Independent Sensor Errors: Binomial Model

For a FP detection, at least k-out-of-n sensors have to make a FP error at the same time. Additionally, FP errors have to occur at a similar location in the FOV for sensor data fusion to associate the FP errors in the individual sensors with each other. We do not further model these spatial aspects but interpret the problem binary according to Section 4.1.1, e.g. by focusing on a limited area of the FOV. Explicitly modeling the spatial aspects is related to addressing association errors in sensor data fusion as a separate error type.

With the binary problem interpretation, a detection in sensor i is a binary random variable D_i. For a discrete point in time, $D_i = 1$ is the event of an object indication in sensor i and $D_i = 0$ is the event of no object indication in sensor i. Because we do not distinguish between different sensors for deriving *sensor perception reliability* requirements, $\mathrm{PFA}_{\mathrm{sensor}}$ is the conditional FP probability for all sensors $i = 1, \ldots, n$, as defined in Eq. (4.2).

Under the assumption of statistically independent errors among different sensors, the number of sensors to make an FP error is binomially distributed with parameter $\mathrm{PFA}_{\mathrm{sensor}}$. Hence the PFA of the perception module is related to the individual sensors' $\mathrm{PFA}_{\mathrm{sensor}}$:

$$\mathrm{PFA} = \sum_{i=k}^{n} \binom{n}{i} \cdot \mathrm{PFA}_{\mathrm{sensor}}^{i} \cdot (1 - \mathrm{PFA}_{\mathrm{sensor}})^{n-i} \tag{4.15}$$

where n is the number of redundant sensors in a particular area of the FOV and k defines the k-out-of-n voter. Eq. (4.15) is the conditional FP probability of the perception module in a specific area of the FOV. In analogy to the derivation of FP errors, a TP detection occurs in the perception module if at least k-out-of-n sensors make a true-positive (TP) detection. It follows that:

$$\mathrm{POD} = \sum_{i=k}^{n} \binom{n}{i} \cdot \mathrm{POD}_{\mathrm{sensor}}^{i} \cdot (1 - \mathrm{POD}_{\mathrm{sensor}})^{n-i} \tag{4.16}$$

$1 - \mathrm{POD}$ is the conditional FN probability of the perception module in a specific area of the FOV.

To derive reliability requirements for FN errors in individual sensors, one has to define in a first step targets on λ_{FN} from $\lambda_{\mathrm{TLS}_{\mathrm{FN,crit}}}$ with Eq. (4.12). In a second step, the targets on λ_{FN} are transformed to targets on POD of the perception module with Eq. (4.8). Thereafter, targets on $\mathrm{POD}_{\mathrm{sensor}}$ are derived from the targets on POD by inverting Eq. (4.16). The relationship of targets on $\mathrm{POD}_{\mathrm{sensor}}$ to $\lambda_{\mathrm{TLS}_{\mathrm{FN,sensor}}}$ is then again given with Eq. (4.8). Equivalently, requirements on FP errors are derived.

Statistically Dependent Sensor Errors: Sensor Error Correlation Coefficient

As discussed in Section 3.1.4, a certain degree of statistical perception error dependence is expected among different sensors. The assumption of independence in Eqs. (4.15)-(4.16) is therefore questionable. To describe the statistical dependence in e.g. FP errors, we define the pairwise correlation coefficient $\rho_{\mathrm{FP}_{i,j}}$ of FP errors among pairs of sensors $i,j\epsilon[1,\ldots,n]$ [216]:

$$\rho_{\mathrm{FP}_{i,j}} = \frac{\mathrm{E}\left[D_i \cdot D_j | O=0\right] - \mathrm{E}[D_i|O=0]\cdot\mathrm{E}[D_j|O=0]}{\sqrt{\mathrm{E}[D_i|O=0](1-\mathrm{E}[D_i|O=0])\cdot\mathrm{E}[D_j|O=0]\cdot(1-\mathrm{E}\left[D_j|O=0\right])}} \tag{4.17}$$

where E[ ] denotes the expectation operator. $\mathrm{E}[D_i|O=0]$ is the expectation of D_i conditional on $O=0$. As we do not distinguish between different sensors in assigning *sensor perception reliability* requirements, it is:

$$\mathrm{E}[D_i|O=0] = \mathrm{PFA}_{\mathrm{sensor}} \tag{4.18}$$

for all $i=1,\ldots,n$. Not distinguishing between different sensors it further holds:

$$\mathrm{E}\left[D_i \cdot D_j | O=0\right] = \Pr\left(D_i=1|D_j=1,O=0\right)\cdot\mathrm{PFA}_{\mathrm{sensor}} \tag{4.19}$$

$\Pr\left(D_i=1|D_j=1,O=0\right)$ is the conditional probability of a FP error in sensor i given a FP error in sensor j.

Each pair of sensors is assumed to have the same correlation coefficient $\rho_{\mathrm{FP}_{i,j}} = \rho_{\mathrm{FP}}$ for the purpose of deriving *sensor perception reliability* requirements. Inserting Eqs. (4.18)-(4.19) into Eq. (4.17) leads to:

$$\rho_{\mathrm{FP}} = \frac{\Pr\left(D_i=1|D_j=1,O=0\right)\cdot\mathrm{PFA}_{\mathrm{sensor}} - \mathrm{PFA}_{\mathrm{sensor}}^2}{\mathrm{PFA}_{\mathrm{sensor}} - \mathrm{PFA}_{\mathrm{sensor}}^2} \tag{4.20}$$

On the one hand, with statistically independent FP errors it is:

$$\Pr\left(D_i=1|D_j=1,O=0\right) = \Pr(D_i=1|O=0) = \mathrm{PFA}_{\mathrm{sensor}} \tag{4.21}$$

and the correlation coefficient becomes $\rho=0$. On the other hand, if it is certain that sensor i makes a FP error when sensor j makes a FP error, i.e. $\Pr\left(D_i=1|D_j=1,O=0\right)=1$, the correlation coefficient becomes $\rho=1$. This is equivalent to full dependence.

When $\mathrm{PFA}_{\mathrm{sensor}}$ is small, it holds $\mathrm{PFA}_{\mathrm{sensor}}^2 \ll \mathrm{PFA}_{\mathrm{sensor}}$ and $\mathrm{PFA}_{\mathrm{sensor}}^2 \approx 0$. Then, Eq. (4.20) can be simplified to enhance the interpretability of ρ_{FP}:

$$\rho_{\mathrm{FP}} \approx \Pr\left(D_i=1|D_j=1,O=0\right) \tag{4.22}$$

That is, the correlation coefficient ρ_{FP} is approximately equal to the conditional probability of a FP error in sensor i given a FP error in sensor j. It follows that even a small numeric value of ρ_{FP} can indicate a strong statistical dependence. For instance, if $\mathrm{PFA}_{\mathrm{sensor}} = 10^{-6}$ and the correlation coefficient is $\rho = 10^{-3}$, then the conditional probability of an FP error in sensor i given an FP error in sensor j is approximately 1000 times larger than $\mathrm{PFA}_{\mathrm{sensor}}$. Hence, the strength of the statistical dependence can be interpreted with the ratio of ρ_{FP} to $\mathrm{PFA}_{\mathrm{sensor}}$.

In analogy to ρ_{FP}, the correlation coefficient of FN errors ρ_{FN} between different sensors is defined as:

$$\rho_{\mathrm{FN}} = \frac{\Pr\left(D_i=0 \middle| D_j=0, O=1\right)\cdot\left(1-\mathrm{POD}_{\mathrm{sensor}}\right)-\left(1-\mathrm{POD}_{\mathrm{sensor}}\right)^2}{\left(1-\mathrm{POD}_{\mathrm{sensor}}\right)-\left(1-\mathrm{POD}_{\mathrm{sensor}}\right)^2} \approx$$
$$\approx \Pr\left(D_i = 0 \middle| D_j = 0, O = 1\right) \tag{4.23}$$

Statistically Dependent Sensor Errors: Beta-Binomial Model

We propose the beta-binomial model [216–221] to describe statistically dependent FP and FN sensor errors in different sensors. Aside of describing the statistical dependence among perception errors, the beta-binomial model has the advantage of conceptually representing the temporal variability in sensor perception error rates (see Section 3.1.4). The temporal variability in sensor perception error rates implies e.g. that also the sensors' probabilities of false alarm are temporally variable.

Let the time be adequately discretized. For example, a discrete trial in which D_i is either zero or one corresponds to a time interval of t_{crit}, as discussed in Section 4.1.1. Let $m = 1,2,3, \ldots M$ identify a discrete point in time. M is the total number of discrete time steps. The sensors' probability of false alarm at point in time m is $\mathrm{PFA}_{\mathrm{sensor},m}$ and the sensors' probability of detection is $\mathrm{POD}_{\mathrm{sensor},m}$.

The beta-binomial model is presented generically in the following. To this end, the beta-binomial model parameters p, p_m and ρ are introduced. In case of no object being present $\{O = 0\}$, $p = \mathrm{PFA}_{\mathrm{sensor}}$, $p_m = \mathrm{PFA}_{\mathrm{sensor},m}$ and $\rho = \rho_{\mathrm{FP}}$. In case of an object being present $\{O = 1\}$, $p = \mathrm{POD}_{\mathrm{sensor}}$, $p_m = \mathrm{POD}_{\mathrm{sensor},m}$ and $\rho = \rho_{\mathrm{FN}}$. The average of p_m over a large number of m is p:

$$p = \lim_{M\to\infty} \frac{1}{M} \sum_{m=1}^{m=M} p_m \tag{4.24}$$

With these definitions, Figure 4.8 summarizes how both aspects, the FP and FN sensor error dependence and the temporal variability in sensor error rates, are accounted for with the beta-binomial model.

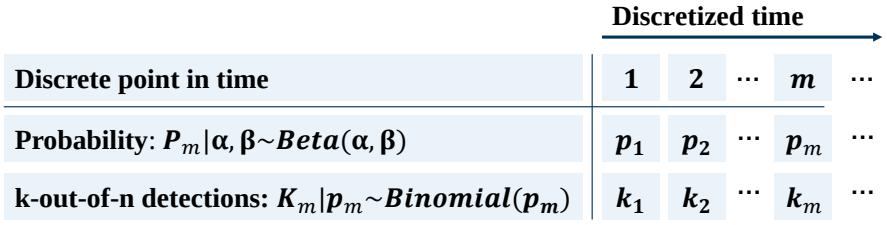

Figure 4.8 The beta-binomial model: at each discrete point in time m, the probability P_m is a sample from a beta distribution with parameters α and β. Given a specific p_m, the number K_m-out-of-n sensors to indicate an object is binomially distributed. Taken and adapted from [156].

First, the variability in p_m at randomly selected discrete points of time is described with a beta distribution $f_{P_m}(p_m|\alpha,\beta)$:

$$f_{P_m}(p_m|\alpha,\beta) = \frac{\Gamma(\alpha+\beta)\cdot {p_m}^{\alpha-1}\cdot(1-p_m)^{\beta-1}}{\Gamma(\alpha)\cdot\Gamma(\beta)} \tag{4.25}$$

α and β are the distribution parameters and $\Gamma(a) = \int_0^\infty u^{a-1}\cdot\exp(-u)\,\mathrm{d}u$ is the gamma function. As is common in the Bayesian literature [114], the notation $f_{P_m}(\cdot\,|\alpha,\beta)$ makes it explicit that the probability distribution of P_m is conditional on the distribution parameters α and β. The beta distribution is bounded by zero and one and is flexible in representing the probability density, hence it is an obvious choice for modeling p_m.

Second, p_m is a factor common to all sensors at point of time m, inherently accounting for the statistical dependence in errors among different sensors. Let $K = \sum_{i=1}^{n} D_i$ be the number of sensors indicating an object at a discrete point in time in a specific area of the FOV. For a given value of p_m at point of time m, the indication of an object in the different sensors is assumed to be conditionally statistically independent and thus the number K-out-of-n sensor to indicate an object is binomial distributed (see Figure 4.8). The assumption of this type of dependence structure is justified by the fact that P_m collectively accounts for context variables (i.e. influencing factors) that are common to all sensors.

Integrating over the uncertainty in the beta distributed binomial parameter p_m leads to the beta-binomial distribution, which here models the probability of exactly k-out-of-n statistically dependent sensors to indicate an object [217, 218, 222]:

$$p_K(k_m|\alpha,\beta) = \Pr(K = k_m|\alpha,\beta) = \binom{n}{k_m}\frac{\Gamma(\alpha+\beta)\cdot\Gamma(\alpha+k_m)\cdot\Gamma(\beta+n-k_m)}{\Gamma(\alpha)\cdot\Gamma(\beta)\cdot\Gamma(\alpha+\beta+n)} \tag{4.26}$$

To facilitate the interpretation of the model parameters in Eq. (4.26), α and β are expressed in terms of the average probability p in a large number of randomly selected points in time and the correlation coefficient ρ [217]:

$$\alpha = \frac{p \cdot (1-\rho)}{\rho}, \beta = \frac{(1-p) \cdot (1-\rho)}{\rho} \tag{4.27}$$

p is the mean of p_m according to Eq. (4.24). The beta-binomial model in Eq. (4.26) is identical to the binomial model under statistical independence $\rho \to 0$ [218]. The beta-binomial model implies equi-correlation[58].

With these definitions, the PFA of the perception module accounting for statistical dependence among FP sensor errors is:

$$\mathrm{PFA} = \sum_{i=k}^{n} \Pr(K = k | p = \mathrm{PFA}_{\mathrm{sensor}}, \rho = \rho_{\mathrm{FP}}) \tag{4.28}$$

$\Pr(K = k | p = \mathrm{PFA}_{\mathrm{sensor}}, \rho_{\mathrm{FP}})$ is the beta-binomial distribution defined with Eqs. (4.26)-(4.27).

Similarly, the POD of the perception module accounting for the statistical dependence in FN sensor errors is:

$$\mathrm{POD} = \sum_{i=k}^{n} \Pr(K = k | p = \mathrm{POD}_{\mathrm{sensor}}, \rho = \rho_{\mathrm{FN}}) \tag{4.29}$$

The inverse of Eq. (4.29) allows to set requirements on $\mathrm{POD}_{\mathrm{sensor}}$ for a given ρ_{FN} and given requirements on POD. Equivalently, Eq. (4.28) is the basis for setting requirements on $\mathrm{PFA}_{\mathrm{sensor}}$.

4.2.3 Numerical Examples

This Section exemplarily derives *sensor perception reliability* requirements from requirements on *perception reliability*.

Deriving Sensor Perception Reliability Requirements

It is here hypothetically assumed that the TLS for safety-critical FN and FP perception errors in a specific area of the FOV is $\lambda_{\mathrm{TLS_{FN,crit}}} = \lambda_{\mathrm{TLS_{FP,crit}}} = 10^{-9}\,\mathrm{h}^{-1}$. With these requirements, we derive the TLS $\lambda_{\mathrm{TLS_{FN,sensor}}}$ and $\lambda_{\mathrm{TLS_{FP,sensor}}}$ for individual sensors.

On the safe side, we set the criticality $p_{\mathrm{FN}} = p_{\mathrm{FP}} = 1$ in Eqs. (4.12)-(4.13) and directly insert $\lambda_{\mathrm{TLS_{FN,crit}}}$ and $\lambda_{\mathrm{TLS_{FP,crit}}}$ into Eqs. (4.7)-(4.8) to derive requirements on POD and PFA of the perception module. Also on the safe side, we set $\Pr(O = 1) = \Pr(O = 0) = 1$ in Eqs. (4.7)-(4.8). Collecting more information on $\Pr(O = 1)$, p_{FN} and p_{FP} would lead to lower *sensor perception reliability* requirements than those derived in the following.

[58] Equi-correlation signifies that the pairwise correlation coefficients $\rho_{i,j}$ of errors among sensors i and j are identical for all pairs of sensors $i, j \in [1, \ldots, n]$, $i \neq j$.

The transformation between continuous error rates $\lambda_{\mathrm{TLS_{FN,crit}}}$ / $\lambda_{\mathrm{TLS_{FP,crit}}}$ and the requirements on discrete probabilities POD /PFA is according to Eqs. (4.7)-(4.8), with $\Delta t = t_{\mathrm{crit}} = 0.5$ s. With requirements on POD and PFA, requirements on $\mathrm{POD_{sensor}}$ and $\mathrm{PFA_{sensor}}$ are derived under the assumption of statistically independent sensor errors with the inverse relationships of Eqs. (4.15)-(4.16). Finally, the requirements on $\mathrm{POD_{sensor}}$ and $\mathrm{PFA_{sensor}}$ are inserted into Eqs. (4.7)-(4.8) to express the *perception reliability* requirements in terms of $\lambda_{\mathrm{TLS_{FN,sensor}}}$ and $\lambda_{\mathrm{TLS_{FP,sensor}}}$. Figure 4.9a) shows the resulting *sensor perception reliability* requirements for $n = 3$ and Figure 4.9b) for $n = 5$ redundant sensors under different voting schemes, assuming independent sensors.

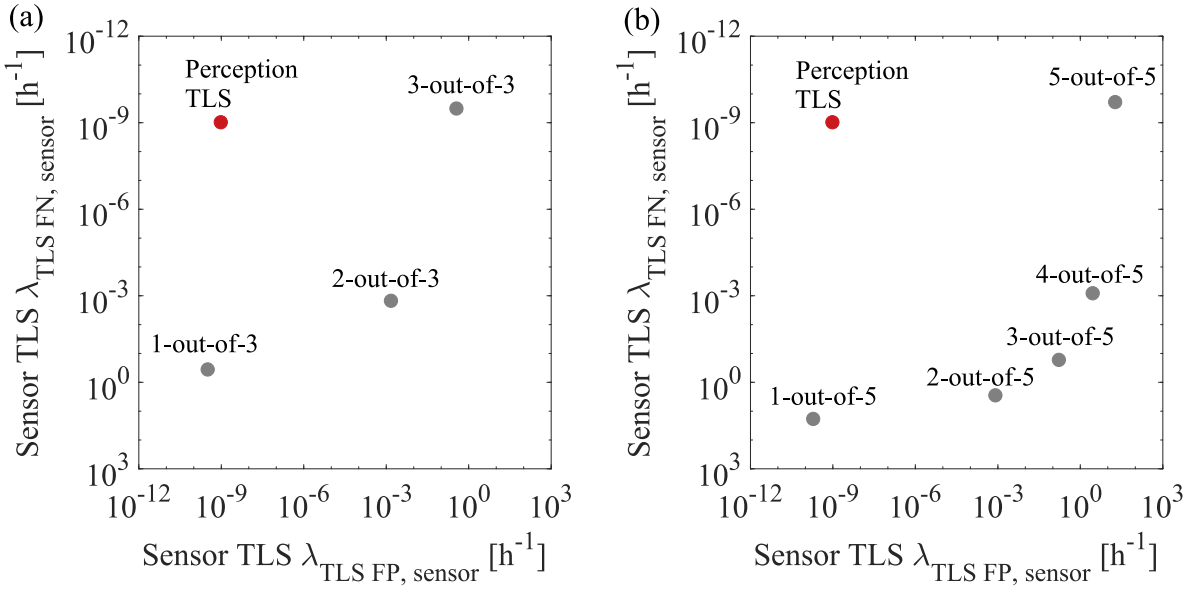

Figure 4.9 Acceptable sensor error rates (grey points) $\lambda_{\mathrm{TLS_{FN,sensor}}}$ and $\lambda_{\mathrm{TLS_{FP,sensor}}}$ to comply with the perception module's requirements (red point) of $\lambda_{\mathrm{TLS_{FN,crit}}} = \lambda_{\mathrm{TLS_{FP,crit}}} = 10^{-9}\,\mathrm{h}^{-1}$ under different voting schemes, when assuming independent sensor errors. (a) The sensor system consists of n=3 and (b) of n=5 redundant sensors. Taken and adapted from [39].

We next consider potential sensor error dependencies with Eqs. (4.28)-(4.29) to derive *sensor perception reliability* requirements without the independence assumption. The required error rates $\lambda_{\mathrm{TLS_{FN,sensor}}}$ and $\lambda_{\mathrm{TLS_{FP,sensor}}}$ for individual sensors are derived exemplarily for selected ρ and are illustrated in Figure 4.10. In the calculations, the correlation coefficients for FP and FN errors are identical, i.e. $\rho = \rho_{\mathrm{FN}} = \rho_{\mathrm{FP}}$.

In case of $\rho = 0$ (statistically independent sensor errors), the requirements are as in Figure 4.9. With increasing correlation coefficients, the points in Figure 4.10 move towards the perception module's TLS, as indicated schematically with the arrows. For full dependence $\rho = 1$, the individual sensor's requirements are identical to the perception module's TLS. With $\rho = 1$, the error occurrence is simply a Bernoulli trial (for instance, either an FP error in all sensors occurs with probability PFA or it does not occur).

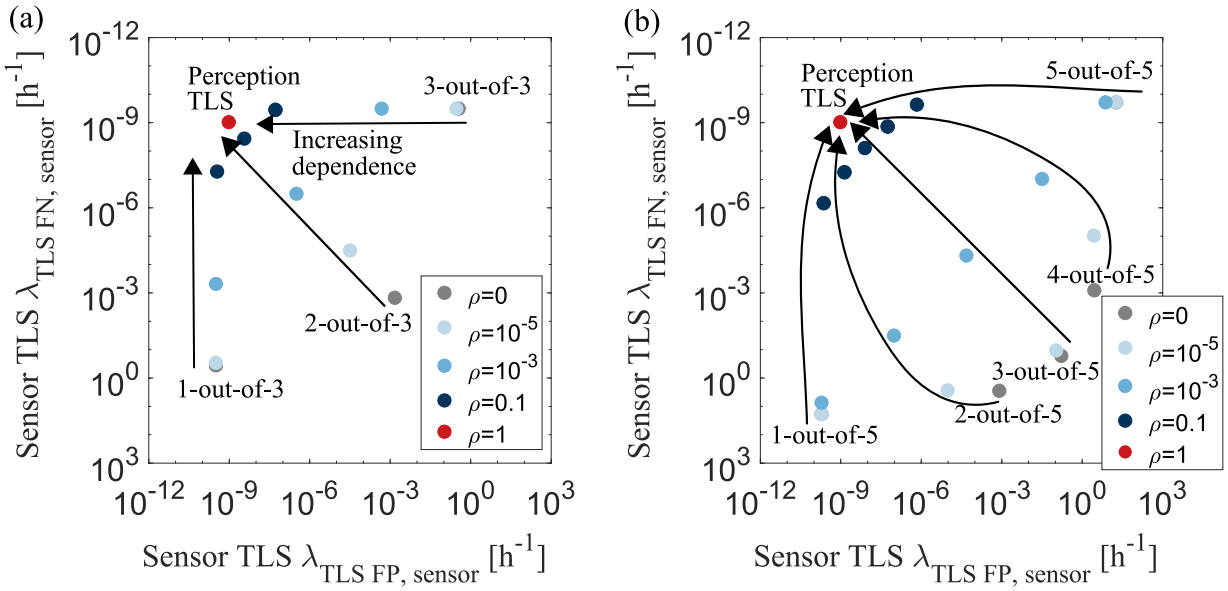

Figure 4.10 **Acceptable sensor error rates $\lambda_{TLS_{FN,sensor}}$ and $\lambda_{TLS_{FP,sensor}}$ to comply with the perception module's requirements (red point) of $\lambda_{TLS_{FN,crit}} = \lambda_{TLS_{FP,crit}} = 10^{-9}\ h^{-1}$ under different voting schemes. The colors of points correspond to different values of the correlation coefficient ρ. The arrows indicate schematically how the acceptable sensor error rates change with increasing ρ. (a) The sensor system consists of $n = 3$ and (b) of $n = 5$ redundant sensors. Taken and adapted from [39].**

Sensor Perception Reliability Requirements in Function of the TLS

Figure 4.11 presents the relationship between *sensor perception reliability* requirements $\lambda_{TLS_{FN,sensor}}$ and the perception TLS $\lambda_{TLS_{FN,crit}}$ with the example of the 2-out-of-3 vote for different values of the correlation coefficient ρ (see Figure 4.9a).

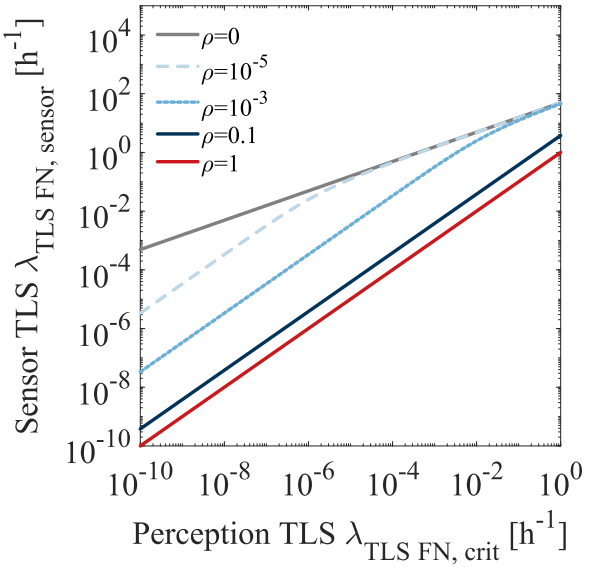

Figure 4.11 **Relationship between the perception TLS $\lambda_{TLS_{FN,crit}}$ and requirements on sensor FN error rates $\lambda_{TLS_{FN,sensor}}$ in case of a 2-out-of-3 voting with different values of the correlation coefficient ρ. Under the 2-out-of-3 voting, the requirements for FP error rates are identical to $\lambda_{TLS_{FN,sensor}}$ if $\rho_{FN} = \rho_{FP}$.**

The relationship between the perception TLS and sensor requirements are identical for FN errors (Figure 4.11a) and FP errors in case of the 2-out-of-3 vote and assuming $\rho_{\mathrm{FN}} = \rho_{\mathrm{FP}}$. This however does not hold in general.

Sensor Perception Reliability Requirements in Function of ρ

Figure 4.12 depicts the relationship between the correlation coefficient ρ_{FN} and sensor requirements $\lambda_{\mathrm{TLS_{FN,sensor}}}$ for a perception TLS of $\lambda_{\mathrm{TLS_{FN,crit}}} = 10^{-9}\,\mathrm{h}^{-1}$ under a 2-out-of-3 voting. The relationship between the correlation coefficient ρ_{FP} and $\lambda_{\mathrm{TLS_{FP,sensor}}}$ under the 2-out-of-3 voting is as in Figure 4.12. With statically independent sensor errors ($\rho \to 0$), the *sensor perception reliability* requirements are identical to the grey point in Figure 4.10a). With fully dependent sensor errors $\rho \to 1$, the requirements are identical to the perception TLS $\lambda_{\mathrm{TLS_{FN,crit}}} = 10^{-9}\,\mathrm{h}^{-1}$ (red point in Figure 4.10a). Small numeric values of the correlation coefficient can according to Eq. (4.22) indicate a strong dependence.

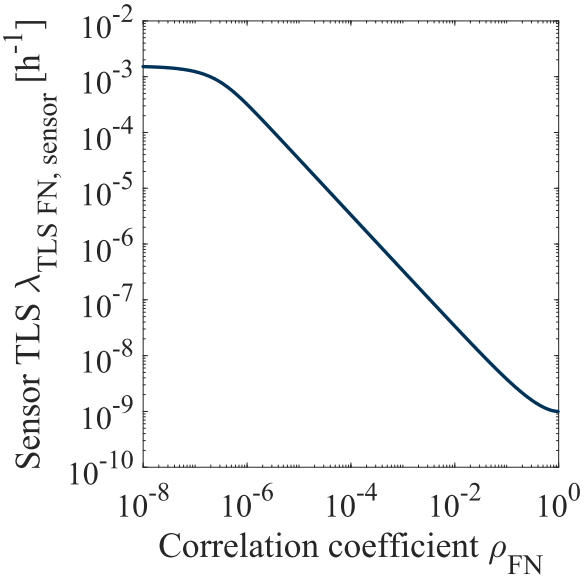

Figure 4.12 Relationship between requirements on sensor error rates $\lambda_{\mathrm{TLS_{FN,sensor}}}$ and the correlation coefficient among FN sensor errors ρ_{FN} under a 2-out-of-3 voting to comply with a perception TLS of $\lambda_{\mathrm{TLS_{FN,crit}}} = 10^{-9}\,\mathrm{h}^{-1}$. The requirements for FP errors under a 2-out-of-3 are as in the Figure. In analogy to Figure 6a) in [65].

4.2.4 Discussion and Conclusions

In the case of statistically independent sensor errors, sensor redundancy together with the k-out-of-n voting allows for larger acceptable values of error rates at the sensor level compared with the perception module, see Figure 4.11. For example, under the assumption of independence, the TLS of an individual sensor is $\approx 1.55 \cdot 10^{-3}\ \mathrm{h}^{-1}$ for a perception TLS of $10^{-9}\ \mathrm{h}^{-1}$ with a 2-out-of-3 voting scheme (Figure 4.9a). The sensor requirement of $1.55 \cdot 10^{-3}\ \mathrm{h}^{-1}$ can be validated by a test effort in the orders of magnitude of 10^3 - 10^4 h. With independent sensor errors, one can thus overcome the approval trap and approach an empirical safety validation (given the assumed models and system decomposition). Not explicitly considered in these calculations are errors introduced by fusing data of individual sensors.

While the assumption of independence is not appropriate, the statistical error dependence between lidar, radar and camera sensors is likely small due to the complementary physical measurement principles [42, 156, 157]. As shown in Figure 4.12, with a weak statistical dependence, requirements on the sensor level are still considerably lower than requirements on the level of the perception module. It follows from Figure 4.10 and Figure 4.12 that one does not only need to evaluate the error rates of individual sensors but also the sensor error dependencies, given their strong effect on system safety.

The *sensor perception reliability* requirements with statistical dependence (Figure 4.12) are derived with the beta-binomial model, which is only one among many possible dependence models. A limitation of the beta-binomial model is that it does not allow to model certain extreme types of statistical dependence [156], which is discussed in detail in Section 6.1.4.

The beta-binomial model additionally assumes exchangeability between different discrete points in time, i.e. no information is conveyed by the exact temporal order of the points in time m. This does not hold if – in addition to the statistical error dependence among the different sensors – the errors in subsequent points in time are also statistically dependent.

In principle, the statistical model illustrated in Figure 4.8 could be extended with an autocorrelation function or Markov chain for $p_1 \ldots p_m$ to model the dependence between subsequent points in time. Alternatively, the dependence could also be reduced by only taking every m^{th} discrete point in time into account, i.e. by thinning out. Another solution strategy for potential temporal dependence is to formulate the model parameters as a function of context variables, which is further be addressed in Section 5.3. Given the state of the context variables, subsequent points in time can be considered to be approximately conditionally independent and exchangeability holds. For simplicity, we do not include context variables here. If the time interval large ($M \to \infty$), the overall frequency of exactly k-out-of-n sensor object detections can without bias be expressed with Eq. (4.26). Therefore, the dependence between subsequent points in time is for simplicity

neglected in the following. It is pointed out though, if the interest is in the frequency of multiple sensor errors in a row, an unjustified assumption of exchangeability would lead to an underestimation of this frequency.

The reliability requirements shown e.g. in Figure 4.9 and Figure 4.10 assume sensor redundancy. In certain situations, sensor redundancy could however be restricted because the performance of individual sensors is systematically impaired. For example, the performance of a lidar could systematically be affected by dirt on the sensor cover [101], or the performance of a camera could systematically be too low in the night. To account for these systematic effects, one either has to restrict the availability of the automated driving functionalities in these situations, define safety mechanisms that mitigate these systematic effects (e.g. illumination and cleaning), or adapt the calculations presented in this Section (e.g. 80% of the time one can assume $n = 3$ sensors and 20% of the time only $n = 2$ sensors are available).

We understand the methods and models in this Section as a means to estimate requirements on *sensor perception reliability*. For evaluating the *sensor perception reliability*, models as proposed in 5.4 are more suitable because they explicitly distinguish error rates of different sensors.

4.3 Perception Reliability Validation: Test Effort Estimation

To plan tests, it is important to estimate the required test effort for a validation of the *perception reliability* requirements, or more generally, a given target level of safety. We therefore present a method to estimate the empirical test effort required to demonstrate a given target level of safety (TLS). For the sake of generality, the target level of safety is in this Section denoted with λ_{TLS}, which could for instance be the requirements $\lambda_{\mathrm{TLS}_{\mathrm{FN,sensor}}}$ or $\lambda_{\mathrm{TLS}_{\mathrm{FP,sensor}}}$ on the individual sensor level, as derived with Section 4.2. Alternatively λ_{TLS} could be the TLS on safety-critical FN and FP errors on the level of the perception module $\lambda_{\mathrm{TLS}_{\mathrm{FN,crit}}}$ or $\lambda_{\mathrm{TLS}_{\mathrm{FP,crit}}}$, the TLS for the perception module $\lambda_{\mathrm{TLS}_{\mathrm{per}}}$ or the TLS for the ADS $\lambda_{\mathrm{TLS}_{\mathrm{sys}}}$. This Section is based on and partly taken from our publication in [65].

The empirical test effort for a given TLS on ADS safety is often estimated with Null Hypothesis Significance Testing (NHST) [35, 41, 44, 215]. NHST is based on the frequentist interpretation of probability. Because NHST is frequently misinterpreted and counterintuitive [223–227], we present an alternative to estimate the test effort with Bayesian statics[59] [114]. We find the Bayesian approach easy to implement and its interpretation more intuitive. An advantage of the Bayesian

[59] In contrast to frequentist approaches such as NHST, the Bayesian interpretation of probability treats observed data as fixed and the parameters that produced the data as random. Further information on the frequentist and Bayesian point of view of probability can be found in [228–231].

method compared to NHST is its flexibility, which allows easier application to non-standard problems. Bayesian methods for reliability assessments are well-known and widely used [150, 232–241].

In particular, we briefly describe the Poisson distribution as the statistical model for the test effort estimation in Section 4.3.1. In Section 4.3.2, we describe how to account for a non-stationary error rate in the test (see Section 3.1.4). The Bayesian approach to estimate the test effort is then explained in Section 4.3.3 and Section 4.3.4 provides numerical examples for test effort estimations.

4.3.1 Statistical Model: The Poisson Distribution

The Poisson distribution is applied in this thesis to estimate the test effort. The Poisson model is also used in e.g. [35, 44] to estimate the test effort for ADS safety validation. To highlight the underlying assumptions when applying the Poisson model to perception errors, it is explained how a Poisson distribution is in the limit related to the binomial model [242].

Let the time be adequately discretized. For example, a discrete trial in which a failure event either occurs or not occurs corresponds to a time interval of t_{crit}, as discussed in Section 4.1.1. Depending on the system under consideration and the scope of a test, a failure event could among others be a predefined safety-critical perception error, a FP error on the sensor or fusion level, a FN error on the sensor or fusion level, or an accident of the ADS. Let $m = 1,2,3, \dots M$ identify a discrete point in time. The number of failure events X in M discrete trials is described with the binomial distribution:

$$p_X(x|p,M) = \frac{M!}{x!\cdot(M-x)!} \cdot p^x \cdot (1-p)^{M-x} \tag{4.30}$$

p is the (generic) probability of a failure event and M is the total number of trials. In the limit, as $M \to \infty$, the binomial distribution leads to the Poisson distribution [242]:

$$p_X(x|\lambda,t) = \Pr(X = x|\lambda,t) \approx \frac{(\lambda\cdot t)^x}{x!} \cdot e^{-\lambda\cdot t} \tag{4.31}$$

The Poisson distribution models the number of failures $X\epsilon\{0,1,2, \dots\}$ in a time interval t for a given system failure rate λ. Here λ is a generic failure rate. Depending on the system and the scope of the test, λ could for instance be the failure rate of the ADS λ_{sys}, the perception failure rate λ_{per}, the rate of safety-critical FN errors of the perception module $\lambda_{\mathrm{FN_{crit}}}$, the rate of safety-critical FP errors of the perception module $\lambda_{\mathrm{FP_{crit}}}$, the FN error rate λ_{FN_i} of sensor i, or the FP error rate λ_{FP_i} of sensor i.

We use the approximation in Eq. (4.31) because the Poisson model is derived from a binomial model in the limit by letting the time interval that corresponds to a discrete trial go to zero, which leads to $p \to 0$ and $M \to \infty$ [242]. For the environment perception, there is however the restriction that a discrete trial cannot be smaller than the measurement cycle time (i.e. the update frequency of the environment model).

Two central requirements are assumed by Eq. (4.31) [242]:

1.) The probability p of a failure event and hence the error rate λ are constant.
2.) The number of failure events in non-overlapping time intervals are independent of each other.

Both requirements are not met by environment perception (see Section 3.1.4).

Point 1.) is addressed in Section 4.3.2. In [65] we proposed to account for the temporal dependence, point 2.), by interpreting the relevant failure events as perception errors that occur for *at least* t_{crit}. The interpretation of perception errors in [65] is however not directly compatible with the relationship between FN /FP error rates and the POD /PFA in Eqs. (4.7)-(4.8), which discretize the continuous time into intervals of *exactly* t_{crit}. Potential strategies to address point 2.) are discussed in Section 4.2.4 in the context of exchangeability. As the purpose of this Section is to initially estimate the test effort, we assume for simplicity in analogy to the discussion in Section 4.2.4 exchangeability in the discrete points in time. Assuming exchangeability neglects the statistical dependence over time, but does not lead to a bias in the estimated failure rate λ (if learned from a large amount data, i.e. $t \to \infty$).

4.3.2 Non-Stationary Error Rate

To account for a non-stationary rate of failures (see Section 3.1.4), $\lambda \cdot t$ in Eq. (4.31) is replaced by the mean number of failures $\mu(t)$ in the time interval t [242]:

$$p_X(x|\mu(t),t) = \Pr(X = x|\mu(t),t) \approx \frac{\mu(t)^x}{x!} \cdot e^{-\mu(t)} \tag{4.32}$$

where $\mu(t)$ is [242]:

$$\mu(t) = \int_0^t \lambda(t)dt \tag{4.33}$$

The average rate of failure events $\bar{\lambda}$ is in analogy to Eq. (3.5):

$$\bar{\lambda} = \lim_{T\to\infty} \frac{1}{T}\int_0^T \lambda(t)dt = \int_{\mathbf{e}} \lambda(\mathbf{e}) \cdot f_{\mathbf{E}}(\mathbf{e})d\mathbf{e} \tag{4.34}$$

where **e** is the vector of context variables (see Section 3.1.4).

To exemplify the calculations of $\mu(t)$ and $\bar{\lambda}$, consider that the variability in $\lambda(t)$ is caused by weather conditions. Let the weather for illustrative purposes be characterized by the four conditions sunny, rainy, snowy and cloudy weather. This means that the context variable weather has possible states $E \in \{\text{sunny}, \text{rainy}, \text{snowy}, \text{cloudy}\}$. When the time interval t is large, the mean number of error occurrences $\mu(t)$ is approximated as:

$$\mu(t) = [\Pr(E = \text{sun}) \cdot \lambda(e = \text{sun}) + \Pr(E = \text{rain}) \cdot \lambda(e = \text{rain}) + \\ + \Pr(E = \text{snow}) \cdot \lambda(e = \text{snow}) + \Pr(E = \text{cloudy}) \cdot \lambda(e = \text{cloudy})] \cdot t \quad (4.35)$$

where e.g. $\Pr(E = \text{sun})$ is the exposure probability towards sunny weather (i.e. the context variable is in state sunny) and $\lambda(e = \text{sun})$ is the (mean) failure rate in sunny weather. The average failure rate $\bar{\lambda}$ in this example is:

$$\bar{\lambda} = \Pr(E = \text{sun}) \cdot \lambda(e = \text{sun}) + \Pr(E = \text{rain}) \cdot \lambda(E = \text{rain}) + \\ + \Pr(E = \text{snow}) \cdot \lambda(e = \text{snow}) + \Pr(E = \text{cloudy}) \cdot \lambda(e = \text{cloudy}) \quad (4.36)$$

It is $\Pr(E = \text{sun}) + \Pr(E = \text{rain}) + \Pr(E = \text{snow}) + \Pr(E = \text{cloudy}) = 1$. Additional context variables are considered by letting **E** be a vector, for example E_1 for weather and E_2 background illumination (e.g. day and night). It follows from Eqs. (4.35)-(4.36), to correctly estimate a representative $\bar{\lambda}$, a test drive of total duration t has to be split into $t_{\text{sun}} = \Pr(E = \text{sun}) \cdot t$; $t_{\text{rain}} = \Pr(E = \text{rain}) \cdot t$; $t_{\text{snow}} = \Pr(E = \text{snow}) \cdot t$; $t_{\text{cloudy}} = \Pr(E = \text{cloudy}) \cdot t$.

Under a varying failure rate $\lambda(t)$, the probability of the number of failure events in a large time interval t can be described by Eq. (4.31), in which $\lambda \cdot t$ is replaced by $\mu(t)$ (or equivalently by replacing λ with $\bar{\lambda}$). For ease of notation, λ is in the following not explicitly replaced with $\bar{\lambda}$ but it is assumed that the non-stationary error rate is represented as outlined in this Section.

4.3.3 Bayesian Test Design

In this Section, we derive the necessary test drive effort to demonstrate $\lambda < \lambda_{\text{TLS}}$, e.g. to demonstrate Eq. (3.2) or requirements such as in Figure 4.10. The problem of demonstrating $\lambda < \lambda_{\text{TLS}}$ is related to inferring an unknown mean rate λ of failure events from a limited amount of data, where the data consists of the number of failure events x observed in a time interval t. We use Bayesian statistics [114] to solve this problem. For a detailed treatment of Bayesian reliability analyses we refer to textbooks [150, 232].

Posterior Distribution of the Failure Rate λ

Bayes' theorem allows to infer λ from x observed in t:

$$f(\lambda|x,t) \propto f(\lambda) \cdot p_X(x|\lambda,t) \tag{4.37}$$

$f(\lambda|x,t)$ is the posterior probability density function (PDF) of the failure rate λ for a given observed number of failures x in the time interval t. $f(\lambda)$ is the prior PDF of λ and $p_X(x|\lambda,t)$ is the likelihood of λ given the observation of x in t. The likelihood is defined by the Poisson PMF in Eq. (4.31). The symbol $\propto$ expresses that the posterior distribution is proportional to the prior and the likelihood up to a constant.

A convenient choice for the prior distribution in case of a Poisson likelihood is the gamma distribution. The gamma distribution is the conjugate distribution to the Poisson likelihood, which means that both $f(\lambda)$ and $f(\lambda|x,t)$ in Eq. (4.37) have the gamma distribution [114]. The gamma PDF is:

$$f(\lambda) = \frac{b^a}{\Gamma(a)} \cdot \lambda^{a-1} \cdot \exp(-b \cdot \lambda) \tag{4.38}$$

where a and b are the parameters of the gamma distribution and $\Gamma(a) = \int_0^\infty u^{a-1} \cdot \exp(-u)\,\mathrm{d}u$ is the gamma function. The corresponding gamma cumulative distribution function (CDF) $F(\lambda|x,t)$ is:

$$F(\lambda|x,t) = \frac{\gamma(a,b\cdot\lambda)}{\Gamma(a)} \tag{4.39}$$

where $\gamma(a, b \cdot \lambda) = \int_0^{b\cdot\lambda} u^{a-1} \cdot \exp(-u)\,\mathrm{d}u$ is the incomplete gamma function.

The prior distribution is described by $f(\lambda)$ with parameters a' and b'. The parameters of the posterior $f(\lambda|x,t)$ are denoted a'' and b'' and are obtained as:

$$a'' = a' + x \tag{4.40}$$

$$b'' = b' + t \tag{4.41}$$

The posterior mean $\hat{\lambda}$ of the unknown error rate λ follows directly from a'' and b'':

$$\hat{\lambda} = \frac{a''}{b''} = \frac{a'+x}{b'+t} \tag{4.42}$$

The posterior mean is the point estimate of λ that minimizes the mean squared error $\mathrm{E}\left[(\hat{\lambda} - \lambda)^2\right]$ [115].

Prior Distribution

Prior parameters a' and b' have to be selected to evaluate Eqs. (4.40)-(4.41). A commonly accepted formal rule to construct an (objective) prior distribution when no prior information is available was defined by Jeffreys [114, 243, 244]. The property that makes Jeffreys' prior non-informative is its invariance to re-parameterizations [114]. Jeffreys' prior yields $a' = 0.5$ and $b' \to 0$. Assuming that no errors are observed in a time interval t, the point estimate of the failure rate according to Eq. (4.42) with Jeffreys prior is $\hat{\lambda} = 0.5/t$. This means that even if no failure events are observed, the posterior mean estimate is not zero.

Eq. (4.42) supports the interpretation of the prior parameters as a' prior error observations in a prior test time interval b' (see [232] page 89). By comparing Eq. (4.38) with (4.31), it is argued in [114] page 52 that the prior parameters can be interpreted as $a' - 1$ prior observations in a prior time interval b'. Following this interpretation, a weakly informative prior in case of no prior information could also be selected as $a' = 1$ and $b' \to 0$. $a' = 1$ and $b' \to 0$ result in the same numerical test effort t as NHST. If substantial information about λ prior to the analysis is available, it can easily be incorporated into a' and b' following the interpretations given. With a finite number of model parameters and a large amount of data ($t \to \infty$), the choice of the prior is irrelevant in the limit [114].

Test Effort Estimation

The basis for the test effort estimation is the probability of the system's failure rate being smaller than the target level of safety $\Pr(\lambda < \lambda_{\mathrm{TLS}}|x,t)$ for a given observed number of failures x in time interval t. Exemplarily, Figure 4.13 shows that the posterior probability of $\lambda < \lambda_{\mathrm{TLS}}$ is the area under the posterior distribution $f(\lambda|x,t)$ up to λ_{TLS}. $f(\lambda|x,t)$ is obtained by updating the prior $f(\lambda)$ with the likelihood $p_X(x|\lambda,t)$. The probability $\Pr(\lambda < \lambda_{\mathrm{TLS}}|x,t)$ (blue shaded area in Figure 4.13) is calculated by inserting λ_{TLS} together with a'' and b'' into the CDF in Eq. (4.39)[60].

To empirically validate $\lambda < \lambda_{\mathrm{TLS}}$, it is imposed that the posterior probability of $\lambda < \lambda_{\mathrm{TLS}}$ for a given test outcome (x,t) has to exceed an upper limit γ:

$$\Pr(\lambda < \lambda_{\mathrm{TLS}}|x,t) > \gamma \tag{4.43}$$

γ is the required level of credibility to accept the hypothesis $\lambda < \lambda_{\mathrm{TLS}}$. To estimate the test effort before conducting a test, the test outcome x is fixed at different values ($x = 0,1,2,\ldots$) and Eq. (4.43) is solved for the required test effort t.

[60] The validity of the probability $\Pr(\lambda < \lambda_{\mathrm{TLS}}|x,t)$ is subject to the adequacy of the underlying modeling assumptions.

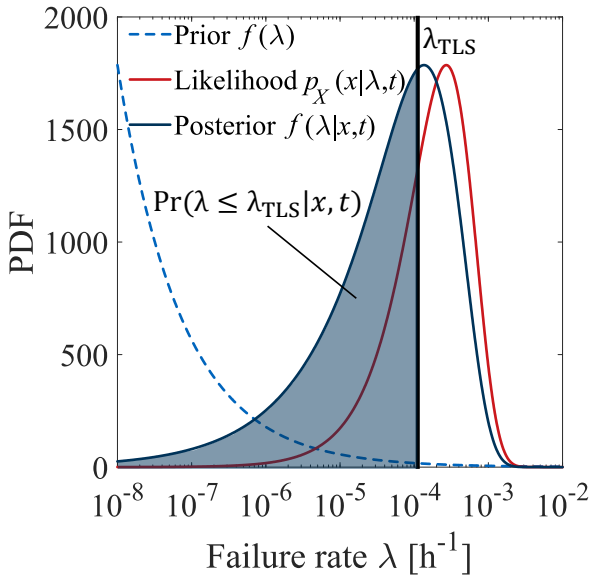

Figure 4.13 How is the probability $\Pr(\lambda < \lambda_{TLS}|x,t)$ derived? At first, the prior PDF is updated with the likelihood for a given number of observed failure events x in a time interval t to form the posterior distribution. The area under the posterior PDF up to the target level of safety λ_{TLS} is then the probability $\Pr(\lambda < \lambda_{TLS}|x,t)$.

The full approach to estimate the test effort is summarized:

1.) Select a suitable statistical model for the number of failures x in time interval t under a failure rate λ. Here the Poisson model is selected.
2.) Derive the posterior distribution of λ given the observed number of failures x in time interval t with Bayesian parameter inference.
3.) Relate the posterior distribution to the probability of $\lambda < \lambda_{TLS}$, i.e. determine $\Pr(\lambda < \lambda_{TLS}|x,t)$ from the posterior CDF.
4.) Impose that the probability of complying with the TLS for a given postulated test outcome x exceeds an upper limit, for instance $\Pr(\lambda < \lambda_{TLS}|x,t) > 0.95$.
5.) Insert λ_{TLS} into $\Pr(\lambda < \lambda_{TLS}|x,t) > 0.95$, fix x at different values (0,1,2…) and solve for t.

The results are the acceptable number of failure events x for a given test effort t, which allow to conclude with at least γ probability that the TLS λ_{TLS} is complied with.

4.3.4 Numerical Examples

The test effort estimation is in this Section demonstrated with numerical examples.

Exemplary Test Effort Estimation

The test effort to demonstrate $\lambda < \lambda_{TLS}$ is exemplarily derived for $\lambda_{TLS} = 1.55 \cdot 10^{-3}\ \mathrm{h}^{-1}$. A TLS of $\lambda_{TLS} = 1.55 \cdot 10^{-3}\ \mathrm{h}^{-1}$ is the requirement on the FP /FN error rates for an individual sensor under statistically independent sensor errors with a 2-out-of-3 voting, as derived in Figure 4.9. Jeffreys' prior parameters $a' = 0.5$ and $b' = 0$ are selected for all calculations in this Section. Jeffreys' prior reflects ignorance on the failure rate λ before conducting a test. With Jeffreys' prior, the prior probability $\Pr(\lambda < \lambda_{TLS} = 1.55 \cdot 10^{-3}\ \mathrm{h}^{-1}) = 1.4 \cdot 10^{-9}$ of $\lambda < \lambda_{TLS}$ is essentially zero. The required level of credibility in Eq. (4.43) is selected as $\gamma = 0.95$.

Figure 4.14a) shows the probability $\Pr(\lambda < \lambda_{TLS}|x,t)$ that λ is smaller than the TLS $\lambda_{TLS} = 1.55 \cdot 10^{-3}\ \mathrm{h}^{-1}$ in function of the test effort t for the cases of $x = 0$, $x = 1$ and $x = 2$ events. To estimate the test effort t to demonstrate $\lambda < \lambda_{TLS}$, the inverse of the functions in Figure 4.14a) at $\Pr(\lambda < \lambda_{TLS}|x,t) = 0.95$ are determined, as indicated for the case $x = 0$ with the black arrow. For instance, with zero acceptable events, a test effort of $t \approx 1250$ h demonstrates $\lambda_{TLS} = 1.55 \cdot 10^{-3}\ \mathrm{h}^{-1}$.

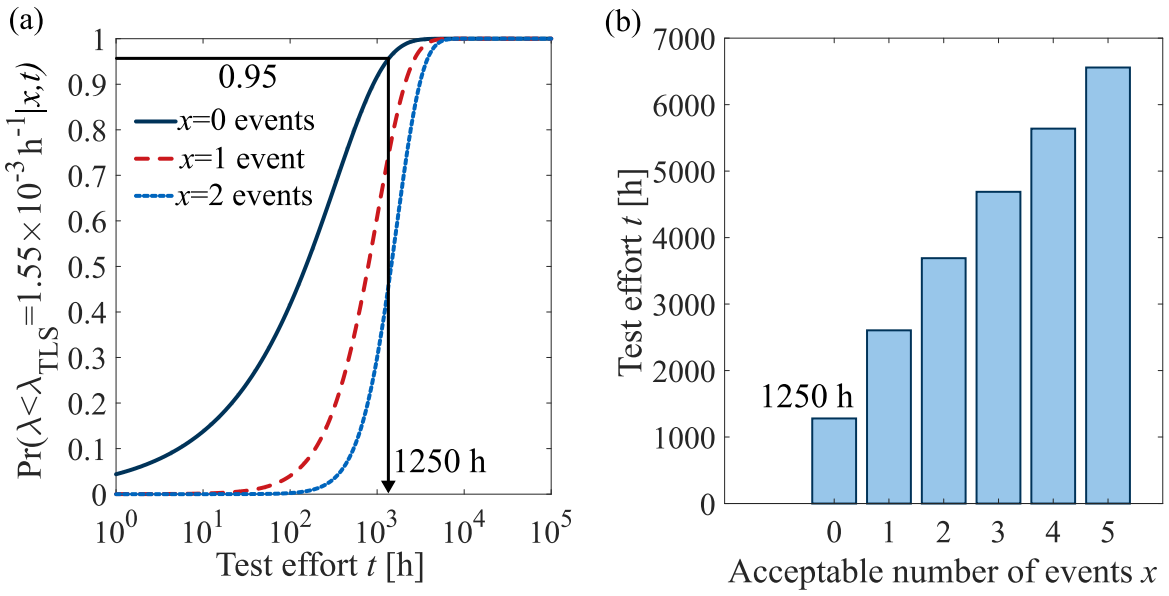

Figure 4.14 (a) Posterior probability $\Pr(\lambda < \lambda_{TLS}|x,t)$ of $\lambda < \lambda_{TLS} = 1.55 \cdot 10^{-3}\ \mathrm{h}^{-1}$ (see 2-out-of-3 voting under independence in Figure 4.9a) in function of the test effort t, for the cases of $x = 0$, $x = 1$ and $x = 2$ events during the test. The black arrow indicates the test effort required to demonstrate $\Pr(\lambda < \lambda_{TLS}|x,t) = 0.95$ in case of $x = 0$ acceptable events. (b) Number of acceptable events x in a test with the corresponding test effort t such that $\Pr(\lambda < \lambda_{TLS}|x,t) = 0.95$. In analogy to Figure 3 in [65].

Note that Figure 3.3 was derived in analogy to Figure 4.14 for a TLS of $\lambda_{\mathrm{TLS}} = 10^{-9}\,\mathrm{h}^{-1}$ to highlight the approval trap.

Test Effort Estimation in Function of Sensor Error Correlation Coefficient

In this example, the methods of Sections 4.2 and 4.3 are combined to estimate the test effort under a 2-out-of-3 vote to comply with a TLS of $\lambda_{\mathrm{TLS_{FN,crit}}} = \lambda_{\mathrm{TLS_{FP,crit}}} = 10^{-9}\,\mathrm{h}^{-1}$, in function of the correlation coefficients ρ_{FN} and ρ_{FP} by applying the beta-binomial model. The results are presented in Figure 4.15 for a credibility level of $\gamma = 0.95$ and zero acceptable FN errors. Under a 2-out-of-3 vote, Figure 4.15 also applies to FP errors. This does however not hold in the general case. The calculations are in analogy to Figure 4.14, by estimating the test effort corresponding to the requirements in Figure 4.12.

In case of statistically independent FP or FN errors among different sensors ($\rho \to 0$), the test effort is 1250 h as in Figure 4.14 for $x = 0$ acceptable events. With fully dependent sensors ($\rho \to 1$), the test effort is $1.92 \cdot 10^{9}$ h, which is identical to directly demonstrating the TLS of $\lambda_{\mathrm{TLS_{FP,crit}}} = 10^{-9}\,\mathrm{h}^{-1}$ on the level of the perception module.

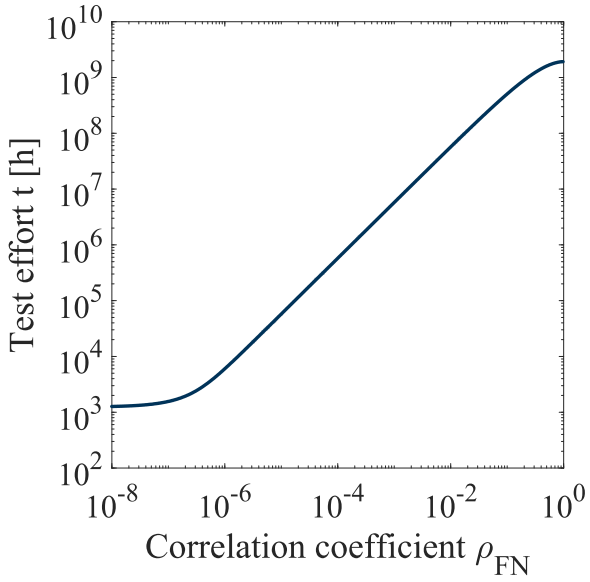

Figure 4.15 Test effort t for $\Pr(\lambda < \lambda_{\mathrm{TLS}} = 10^{-9}\,\mathrm{h}^{-1} | x, t) = 0.95$ with zero acceptable events (errors) in function of the FN sensor error correlation coefficient. The illustrated test effort corresponds to the requirements in Figure 4.12a). In analogy to Figure6b) in [65]. An identical test effort is obtained for FP errors under the 2-out-of-3 voting.

4.3.5 Discussion and Conclusions

It follows from the approval trap (Figure 3.3) that a direct ADS safety validation with Eq. (3.2) is not practical. Likewise, as pointed out in Section 3.1.4, validating the *perception reliability* by demonstrating $\lambda_{\mathrm{per}} < \lambda_{\mathrm{TLS_{per}}}$ on the level of the perception module (fused environment model) is subject to the approval trap.

To address the approval trap, we proposed to conceptualize environment perception in terms of individual sensors and to validate *perception reliability* on the sensor level by exploiting sensor redundancy. With the described modeling assumptions, exploiting redundancy reduces the required test effort for $\lambda_{\mathrm{TLS_{per}}}$ considerably, if sensor errors are statistically independent among different sensors, as illustrated in Figure 4.14. For instance, with $n = 3$ redundant and statistically independent sensors, the effort is here estimated as ≈ 1250 h failure free testing to validate target error rates of 10^{-9} h^{-1} on the level of the perception module (fused environment model). Subject to the assumptions, the described approach hence might make it possible to overcome the approval trap and "to drive to safety".

It is pointed out that even though all combinations of acceptable events x and the corresponding test efforts t in Figure 4.14b) demonstrate compliance with the target level of safety with 95 % credibility, it is better to select the test design a-priori, and not to adjust it based on the observed number of errors during the test. If a stopping criteria is selected during the test based on the observed number of events, the testing can be biased for a given failure rate λ [245, 246].

A challenge is to plan a representative test (see Section 3.1.4) as a prerequisite for correctly estimating the average rate of failure events $\bar{\lambda}$ according to Eq. (4.34). To address this challenge, we account for a non-stationary error rate in function of the context variables **E**. In practice, it is however difficult to know a priori all important context variables together with their probabilities. Additionally, not all context variables such as dirty sensor covers or the strength of snowfall can be easily measured and quantified in practice. Also, the probability of for instance $\Pr(E = rain)$ varies geographically. A chance to address the varying exposure probability towards the context variables in different regions is to learn e.g. $\lambda(e = \mathrm{rain})$, $\lambda(e = \mathrm{sun})$ independently of the geographical region. Eq. (4.36) then allows to combine the error rates with the exposure probabilities for each region, to estimate $\bar{\lambda}$ for each region separately. With this approach, one can also include additional context variables **E** with (systematic) effects on sensor performance, such as darkness or dirt.

As is already discussed in Section 4.2.4, it is not automatically given that errors among different sensors are statistically independent. To show the influence of the statistical dependence, Figure 4.15 estimates the test effort in function of the correlation coefficient of FP /FN sensor errors

among different sensors. Depending on the statistical dependence, the test effort is estimated between $1250\,\mathrm{h}$ and $1.92 \cdot 10^9\,\mathrm{h}$. This clearly highlights the important role of statistical dependence among different sensors for *perception reliability*. An important conclusion is therefore, that one not only has to assess the respective (sensor) error rates but also the sensor error dependence from test data. Simultaneously validating sensor error rates and sensor error dependence however might require longer tests than the efforts estimated in this Chapter, which are conditional on given correlation coefficients.

While the statistical dependence in errors among different sensor is accounted for with the beta-binomial model, we neglect the temporal error dependence in the test effort estimation by assuming exchangeability in Section 4.3.1. Modeling both types of dependence simultaneously has to be addressed in the future. In particular, it should be studied from data which dependence structure adequately describes perception errors among different sensors and which dependence structure adequately represents the temporal perception error dependence.

It has to be understood that the modeling assumptions behind the methods presented in this Section are flexible. For instance, the Bayesian test effort estimation in Section 4.3.3 is flexible with respect to the statistical model assumptions in Section 4.3.1. Ultimately, with the availability of data collected in tests, all modeling assumptions such as statistically independent sensor errors over time or among different sensors must be validated. If modeling assumptions do not hold, the models and the respective required test effort have to be adapted. Hence, the test efforts derived in this Chapter should be interpreted as initial estimates, which have to be updated dynamically in the course of the validation.

5 Assessing Sensor Perception Reliability

In the previous Chapter we advocate addressing the approval trap for ADSs with *higher levels of driving automation* by inductively (empirically) demonstrating *perception reliability* at the sensor level. Requirements on the sensor level are derived through a deductive decomposition of the perception module's reliability metrics. Exemplarily, Figure 4.15 depicts the estimated test effort with this approach. Ultimately, *sensor perception reliability* has to be learned in tests from data to quantify the perception failure rate λ_{per}. With λ_{per}, an ADS's safety is validated by demonstrating Eq. (3.4).

In this Section, we describe various methods to assess *sensor perception reliability*, which mostly can also be applied to assess the *perception reliability* of the fused environment model. The methods we introduce build on established test and safety validation methods, as reviewed in Sections 3.2 and 3.3:

- Qualitative and semi-quantitative analysis methods
- Virtual simulations
- Tests on proving grounds
- Field tests

We start in Section 5.1 with a presentation of a qualitative and semi-quantitative method to assess the risk of perception deficiencies. The method identifies potential context variables **E** with influence on sensor performance and preliminarily assesses the risk associated with **E**. This method can for instance be applied in the concept phase in the context of a preliminary hazard and risk analysis (see Figure 3.6).

Due to the integral part of virtual simulations in the development of ADSs, we outline in Section 5.2 with the example of a lidar's detection performance under rainy conditions how sensor *perception reliability* metrics (see Section 4.1) can be estimated by simulations [105]. Such a simulation method could help to inform the design of a sensor (e.g. in the design phase of the V-Model, see Figure 3.6) and to initially verify compliance with *sensor perception reliability* requirements, when the sensor of interest is not yet physically realized.

With progressing development of the sensors and their integration into the system's perception module, the option of testing the sensors in reality becomes available. In Section 5.3 we propose a method that allows to learn and predict *sensor perception reliability* from tests on proving grounds. Evaluating *sensor perception reliability* on proving grounds with the proposed method could contribute to a *perception reliability* validation.

Finally, we generically outline in Section 5.4 how to learn *(sensor) perception reliability* metrics in the existence uncertainty domain (see Section 4.1) from data collected in field tests. The prerequisite is a reference truth, which allows the identification of perception deficiencies. If the field test is conducted representatively, this is the most realistic test method, and therefore could be utilized to obtain a final validation of *(sensor) perception reliability*.

5.1 Semi-Quantitative Sensor Perception Reliability Analysis

This Section is partly taken from our publication in [39].

One of the first steps in a risk analysis is a qualitative or semi-quantitative hazard and consequence analysis [242]. It is based on expert knowledge, experience and an understanding of the physical principles and processes behind the sensor technology. For instance, a preliminary hazard analysis and risk assessment according to ISO 26262 (Section 3.2.1) is a semi-quantitative method.

As outlined in Section 3.1.4, it is important for a *perception reliability* analysis to consider the influence of context variables **E**. This Section therefore presents a semi-quantitative analysis method to estimate the influence of **E** on *perception reliability*. On the basis of established hazard analysis techniques [152], we present in Section 5.1.1 how to identify context variables that potentially cause perception deficiencies. In Section 5.1.2 we assess the risk of perception deficiencies associated with the identified context variables.

5.1.1 Hazard Analysis: Identification of Relevant Context Variables

The main purpose of a hazard analysis for technical systems is to identify "what can go wrong?", "how can it happen?" and "what controls exist" (or can be implemented)? [242]. Hazard analysis techniques are a fundamental basis for the identification of hazards, failure modes, failure /error causes, failure effects, test cases, influencing factors, countermeasures (safety goals) and potential error diagnostics [152]. It is applied to obtain a deep understanding of the system from a safety perspective, to obtain initial estimates of system risk and is the basis for designing a safe system. The most widely applied methods in this context are various forms of Hazard Analyses, Event Tree Analysis (ETA), Fault Tree Analysis (FTA) and Failure Modes and Effect Analysis (FMEA) [152]. These methods are well-known and are therefore not presented here, for further information it is referred to [150, 152].

To identify relevant context variables, failure modes ("Manner by which an item fails" [152]), are analyzed in a first step, for instance with FMEA, brainstorming, functional block diagrams of a sensor's perception, or other suitable methods. The (sensor's) perception failure modes are related to the existence, state and classification uncertainty domains (see Section 4.1).

Based on a failure mode, one can identify in a deductive manner (e.g. with a FTA, analyzing the physical principles and functional processes behind the sensor technology, expert knowledge, experience, mind maps or other suitable methods) corresponding failure mechanisms ("physical, chemical or other process that has led to a failure" [247]), failure causes ("Circumstances during design, manufacture or use that have led to a failure" [247]) and root causes ("fundamental causes […], causes upon which remedial actions can be decided." [150]). The root causes are part of the context variables **E**, the state of all conditions with influence on the perception performance. Note that one failure mode can have many root causes. This step is exemplified for a few failure modes of a lidar sensor in Figure 5.1.

Root cause	Failure cause	Failure mechanism	Failure mode
Snowfall	High particle concentration in atmosphere	Large atmospheric backscattering	False positive
Fog	High particle concentration in atmosphere	Large atmospheric backscattering	False positive
Object surface material and color	Backscattering of energy at object not sufficient	Losses due to reflective properties of object	False negative
Rainfall	High particle concentration in atmosphere	Large atmospheric transmission losses	False negative
Snow accumulation on sensor cover	Blocking the optical path on sensor cover	Reduced transmitter / receiver efficiencies	Blindness / unavailability

Figure 5.1 Exemplary identification of root causes potentially leading to perception deficiencies in an automotive lidar. The root causes are context variables (see Section 3.1.4). Adapted from [39].

5.1.2 Semi-Quantitative Risk Analysis: Assessing the Influence of Context Variables

To identify the most critical root causes, a semi-quantitative risk analysis is performed. To this end, the probability of exposure $\Pr(E)$ towards an identified context variable E as well as the context variable's consequences $c(E)$ on perception performance are estimated. For simplicity, it is assumed that a given context variable E is binary, it either occurs or not. Further, each context variable is assessed individually, i.e. interactions and the joint effect of being exposed to multiple context variables are not assessed. With these conditions, the $\mathrm{Risk}(E)$ of a context variable is expressed as:

$$\mathrm{Risk}(E) = \Pr(E) \cdot c(E) \tag{5.1}$$

The consequences $c(E)$ on perception performance can be interpreted qualitatively as the severity of the perception performance impairment under E (e.g. low, medium, high), or more quantitatively as a rate $\lambda_{\text{per}}(E)$ of perception errors in function of E[61]. The results of this risk analysis are visualized in a risk matrix [242] by plotting $\Pr(E)$ over $c(E)$.

Aside of the risk, it is important whether a certain context variable is diagnosable. For example, if a context variable is easy to diagnose, safety mechanisms that mitigate adverse effects can be devised. It is proposed to classify each context variable into one of the following three categories:

- Easy to diagnose: safety mechanisms are devised that either mitigate the effect of the context variable, or restrict the availability of the automated driving functionality in the presence of context variables that lead to inacceptable performance impairment. Methods of ISO 26262 are applicable. With adequate safety mechanisms, the context variable is not directly relevant for perception reliability.
- Hard to diagnose: at first relevant for perception reliability. Test should further evaluate under which conditions the risk associated with a context variable is unacceptable. For example, it could be evaluated, which rainfall intensity leads to an unacceptable perception reliability. Considering the test results with the associated risk, it should be determined if it is feasible to diagnose the context variable and to devise respective safety mechanisms.
- Probably impossible to diagnose: the influence of the context variable is hard to mitigate. Its influence should be accounted for in a perception reliability analysis.

Figure 5.2 shows a risk matrix together with a diagnosability assessment for the context variables with potential influence on a lidar's performance identified in Figure 5.1. The lines in Figure 5.2 are equi-risk lines when both the occurrence frequency and the consequences are in log scale [242]. The equi-risk lines allow to prioritize /rank the context variables according to their *preliminarily estimated* risk. For instance, one would conclude that snowfall is expected to be a larger risk for the perception performance of a lidar than rainfall which in turn is a larger risk than fog.

The snow accumulation on a lidar's sensor cover completely blinds the sensor and is therefore the context variable with the largest risk in Figure 5.2. A blindness detection can however be implemented, which mitigates the large risk. If a sensor is blind, one could activate a heating and cleaning mechanism while relying for a short time on redundant sensors. If the sensor is still blind after these measures, the automated driving functionality could be deactivated. Therefore, the

[61] The risk in Eq. (5.1) is not to be confused with the risk of Eq. (3.1). In Eq. (5.1), the consequences are evaluated on the (sensor) perception level and in Eq. (3.1) on the ADS level. The selection of the consequences determines how to interpret Eq. (5.1). The interpretation of Eq. (5.1) is here however not central, because the purpose is to evaluate the influence of the context variables relative to each other and not to evaluate the absolute risk of a context variable.

snow accumulation on the sensor cover is not critical[62] for the *perception reliability* but might reduce the availability.

In contrast, it is probably not possible to detect if an object's surface material reflects too little energy towards the lidar to detect the object[63]. The object's surface material is therefore a very important influence to account for in the *perception reliability* analysis. Rainfall is an example of a context variable classified as hard to diagnose because the exact rainfall intensity is not readily measurable in a vehicle. At first, rainfall is therefore relevant for *perception reliability*. Further tests and studying the feasibility of diagnosing the presence of rainfall are a chance to mitigate the risk of perception deficiencies, for example by restricting the sensor availability if the rainfall exceeds a critical intensity threshold. One would however need to develop solutions to diagnose the exceedance of the critical intensity threshold.

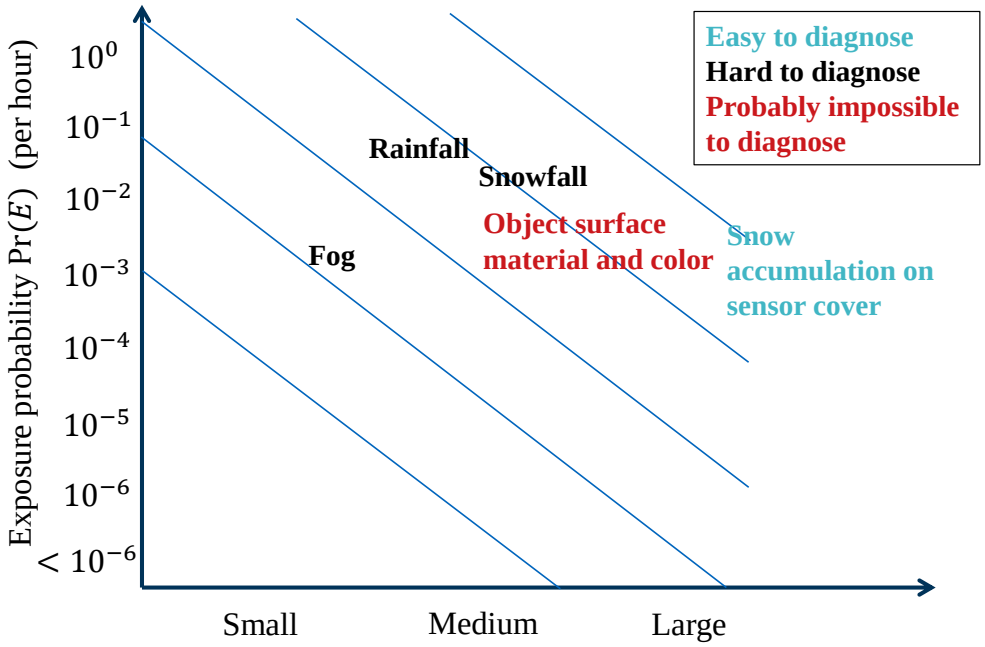

Figure 5.2 Exemplary risk matrix for an initial semi-quantitative risk assessment of context variables with potential influence on the performance of a lidar sensor. Colors represent the diagnosabiltiy of different context variables. Adapted from [39].

[62] It is pointed out, for the snow accumulation not to be critical, the blindness detection has to work flawlessly and without time delay. Additionally, the deactivation of the automated driving functionality must not lead to a risk.
[63] An exception is the availability of credible sensor information from one or more redundant sensors. The difficulty then is however that one cannot be certain which sensor is correct (the lidar not indicating an object, or the other sensor(s) indicating an object, which could also be a FP error).

5.1.3 Discussion

The purpose of the analysis is to identify context variables (conditions that influence *perception reliability* and /or could cause perception deficiencies) and to evaluate their relative influence on *perception reliability*. The identification of context variables **E** with their associated risk helps to further develop virtual simulations, to describe and prioritize test cases, to allocate resources to testing, and to plan representative field tests. Results such as Figure 5.2 are the basis for more detailed quantitative analyses according to the following Sections in later stages of the system safety validation. Additionally, the method in this Section is an important development tool to devise risk reduction measures (i.e. safety mechanisms) for context variables that are diagnosable. These risk reduction measures increase the SOTIF.

The semi-quantitative risk assessment in Section 5.1.2 assesses each context variable individually, and the joint effect of multiple context variables is not considered. Perception errors are however likely caused by a combination of factors /conditions in the environment. A combination of factors that cause a perception error are hard to identify and to assess early in the development, as no data is available. In principle, one can extend the analysis to assess each combination of context variables. This is however impractical. Considering that the purpose of the analysis is to identify the most critical context variables, it might be sufficient to evaluate each context variable separately. It is further pointed out, because the risk associated with each context variable is assessed individually, the sum of Eq. (5.1) over all context variables and their states is not a proper estimate of total risk[64].

Also, the analysis does not account for the fact that some context variables are continuous parameters. For instance, weak rainfall has a different influence on *perception reliability* than strong rainfall. To account for continuous parameters, one could introduce additional sub categories (e.g. weak rainfall, strong rainfall) in the semi-quantitative risk assessment. Ultimately, the testing should account for the continuous nature of the context variables.

A limitation is that there is no certainty that all important context variables are identified with this analysis, which to a large degree is based on expert knowledge and not on data. Due to the subjective nature of the semi-quantitative risk assessment, results such as Figure 5.2 are by no means comprehensive and should only be interpreted qualitatively.

[64] More correctly accounting for all context variables jointly, the total risk is (in the case of discrete context variables) the sum over all combinations of states in the joint space **E** of the context variables: $\text{Risk} = \sum_{\mathbf{E}} \Pr(\mathbf{E}) \cdot c(\mathbf{E})$. It is however not practicable to estimate $\Pr(\mathbf{E})$ and $c(\mathbf{E})$ without data. Also, **E** might consist of many individual context variables, which renders an assessment of all combinations of states in **E** as impracticable.

5.2 Virtual Simulation: Estimation of a Lidar's Detection Performance

This Section is partly taken from our publications in [39, 105].

Virtual simulations are an important test and verification tool in the development of ADSs, as outlined in Section 3.3. Simulations are an opportunity to estimate *(sensor) perception reliability* early in the ADS development. Complex system interactions can be accounted for in a simulation by implementing real software code to be employed in an ADS. In the context of environment perception, it can be distinguished between statistics-based simulations [248, 249], physics-based simulations [104, 250–252] or a combination of both [105, 253].

Statistical simulations are an option to reproduce the statistical sensor error behavior in a virtual environment. Statistical simulations can for example be combined with actual software code of sensor data fusion in a SiL. The SiL then allows to obtain an estimation of *perception reliability* (fused environment model), based on *sensor perception reliabilities*. While we see the methods in Section 4.2 as a means to obtain an initial estimate of sensor requirements early in the development of an ADS, such a SiL could be used to refine the initially derived requirements in final phases of the development. A prerequisite for valid statistical simulations is to learn the *sensor perception reliabilities* in the different uncertainty domains (see Section 4.1) from data, accounting for relevant context variables [157]. Methods to learn *sensor perception reliabilities* are outlined in Sections 5.3, 5.4 and 6.2.

The main idea of physics-based simulations is to reproduce the physical processes behind a sensing technology in a computer model, for instance by means of a ray-tracing simulation [252]. Challenges for physics-based simulations are the need to know relevant influencing factors (i.e. context variables **E)** and to be able to physically model the influence of **E** on the sensing processes. As discussed in Section 3.3, simulations themselves need to be validated, leading to additional test effort in the real world [44, 66, 201]. To the best of our knowledge, currently no comprehensive and widely accepted simulation framework exists to solely validate *sensor perception reliability* virtually.

Even without extensive validation of a simulation, a sound physical simulation approach can however be used early in the development to qualitatively estimate the effect of adverse influencing factors **E** on *sensor perception reliability* and to optimize the sensor design. In the following, we exemplarily present such a simulation framework for lidar sensors.

5.2.1 Simulating the Effect of Rainfall on a Lidar Sensor: Background and Motivation

A challenge is to quantify the effect of context variables on a sensor's performance early in the ADS development. A drawback of lidar sensors is for instance their sensitivity towards adverse weather conditions such as rainfall, snowfall and fog [102, 104, 159]. Hence, the weather condition is an important context variable for lidar sensors. To exemplarily address this challenge, we combine statistical and physical simulation methods to estimate the effect of rainfall on lidar *perception reliability* in this Section.

Previously, the lidar equation was employed to estimate the maximum lidar range under adverse weather conditions [104]. The approach used Mie theory [254, 255] to physically describe backscattering and signal attenuation in the atmosphere caused by particles such as raindrops [104]. The model used mean values of the rainfall dependent attenuation coefficient, which describes the lidar's signal attenuation in the atmosphere [104].

We propose a probabilistic extension of the modeling framework in [104], accounting for the variability in parameters such as the rainfall intensity, the number of raindrops per volume segment of laser beam and the raindrop size. We combine basic probabilistic models for these key rainfall parameters with Mie theory [254, 255] and the theory of signal detection [206–208] in a Monte Carlo simulation framework [242]. The developed simulation framework allows to estimate a lidar's raw detection performance, expressed with a receiver operating characteristic (ROC) curve (see Section 4.1).

5.2.2 Physical Lidar Model

As explained in Section 2.1.3, a lidar raw detection and its corresponding distance measurement R are triggered whenever the received signal intensity (i.e. the backscattered laser pulse) exceeds a detection threshold. The received signal is physically modeled with the lidar equation [102–104]. Here we present the lidar equation in reference to [104].

The received signal power $P_{\mathrm{R}}(R)$ at the lidar is described in function of the distance R with a convolution integral:

$$P_{\mathrm{R}}(R) = A_{\mathrm{R}}\eta_{\mathrm{T}}\eta_{\mathrm{R}} \int_{t'=0}^{2\cdot R/c_0} P_{\mathrm{T}}(t')\, H\left(R - c\cdot\frac{t'}{2}\right) dt' \tag{5.2}$$

where A_{R} is the aperture area of the optical receiver, η_{T} and η_{R} are the optical efficiencies of the transmitter and receiver, respectively. The upper bound of the integral is the Time-of-Flight, Eq. (2.2), to distance R and back to the emitting sensor, where $c_0 = 3\cdot 10^8\ \mathrm{m/s}$ is the speed of light. $P_{\mathrm{T}}(t')$ is the transmitted signal at time t', and $H(R)$ is the spatial impulse response function.

$P_T(t')$ of a single laser pulse is modeled:

$$P_T(t) = \begin{cases} P_0 \cdot \sin^2(\pi \cdot t/\tau_p), & 0 \le t \le \tau_p \\ 0, & t > \tau_p \end{cases} \tag{5.3}$$

with P_0 the peak transmit power and τ_p the pulse width.

The spatial impulse response function is the product of the spatial impulse response of the optical channel $H_C(R)$ and the spatial impulse response of targets $H_T(R)$:

$$H(R) = H_C(R) \cdot H_T(R) \tag{5.4}$$

The spatial impulse response of the optical channel $H_C(R)$ is:

$$H_C(R) = \left[\exp\left(-\int_0^R \alpha(r)dr\right)\right]^2 \cdot \frac{\zeta(R)}{2 \cdot \pi \cdot R^2} \tag{5.5}$$

with $\alpha(r)$ the local extinction coefficient and $\zeta(R)$ the crossover function. $\alpha(r)$ is defined in Section 5.2.3.

$\zeta(R)$ models the overlap of the area illuminated by the transmitter and the area observed by the receiver in dependence of the distance R. To explain the crossover function $\zeta(R)$, the case of a bistatic beam configuration with parallel optical axes is depicted in Figure 5.3.

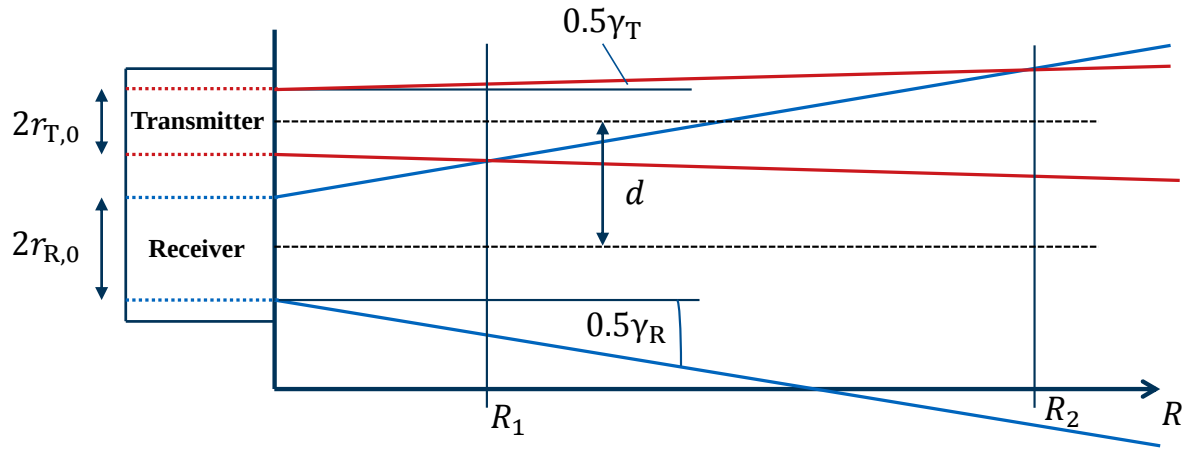

Figure 5.3 Lidar with bistatic beam configuration. The transmitter's optical channel (red) and the receivers optical channel (blue) start to overlap at distance R_1. At distance R_2, the optical channels are completely overlapping. Adapted from Fig.2 in [104].

As Figure 5.3 shows, the transmitter's and receiver's optical channels do not overlap for distances $R < R_1$, hence $\zeta(R < R_1) = 0$:

$$R_1 = \frac{d - r_{T,0} - r_{R,0}}{\tan\left(\frac{\gamma_T}{2}\right) + \tan\left(\frac{\gamma_R}{2}\right)} \tag{5.6}$$

where γ_T and γ_R are the divergence of the transmit beam and receiver's channel, respectively. d is the displacement between the optical axes of the receiver and transmitter, $r_{T,0}$ and $r_{R,0}$ are the radius of the transmission and reception aperture, respectively.

For $R \geq R_2$, the crossover function $\zeta(R \geq R_2) = 1$, i.e. the transmitter's and receiver's optical channels overlap completely:

$$R_2 = \frac{d - r_{R,0} + r_{T,0}}{\tan\left(\frac{\gamma_R}{2}\right) - \tan\left(\frac{\gamma_T}{2}\right)} \tag{5.7}$$

Under the assumption of a homogenous intensity distribution over the beam's cross section, $\zeta(R)$ between $R_1 \leq R \leq R_2$ is modeled with the intersection of the transmit beam's cross section and the receiver's optical channel at distance R (circle-circle intersection):

$$\zeta(R) = \frac{r_T^2 \cdot (\phi_T - \sin(\phi_T)) + r_R^2 \cdot (\phi_R - \sin(\phi_R))}{2 \cdot \pi \cdot r_T^2} \tag{5.8}$$

with r_T the radius of the transmit beam cross section:

$$r_T = R \cdot \tan\left(\frac{\gamma_T}{2}\right) + r_{T,0} \tag{5.9}$$

r_R is defined equivalently. Further it holds:

$$\phi_T = 2 \cdot \arccos\left(\frac{r_T^2 - r_R^2 + d^2}{2 \cdot d \cdot r_T}\right) \tag{5.10}$$

ϕ_R is defined equivalently by swapping r_T and r_R.

Finally, the spatial impulse response of the targets is defined as:

$$H_T(R) = \begin{cases} \Gamma \cdot \delta(R - R_0) + \beta(R), & A_T > A_B(R) \\ \Gamma \cdot \delta(R - R_0) \cdot \frac{A_T}{A_B(R)} + \beta(R), & A_T < A_B(R) \end{cases} \tag{5.11}$$

where Γ is the reflectivity of a hard target (e.g. a car) located at distance R_0, $\delta(R)$ is the dirac function and $\beta(R)$ the backscattering coefficient of soft targets (e.g. raindrops) at distance R. $\beta(R)$ is defined in Section 5.2.3. A_T is the cross sectional area of the target and $A_B(R)$ is the cross section of the lidar beam at distance R.

5.2.3 Electromagnetic Absorption, Scattering and Transmission

Part of a laser pulse's energy is absorbed and diffusely scattered by particles during the propagation through the atmosphere, and only the remaining energy is transmitted [102, 104]. Absorption and scattering are described with the extinction coefficient $\alpha(R)$. Additionally, the backscattering coefficient $\beta(R)$ describes the part of the electromagnetic energy scattered back into the direction

of the emitting source. A physical description of a laser pulse's backscattering and attenuation by particles in the atmosphere (e.g. rain drops) is outlined in this Section with Mie theory to define $\alpha(R)$ and $\beta(R)$ [255, 256]. As a simplification and to separate the effect of precipitation, additional noise sources are neglected.

According to the law of Lambert-Beer, $\alpha(R)$ is defined [257]:

$$\alpha(R) = N \cdot \overline{\sigma}_{\text{ext}} \tag{5.12}$$

where N is the number density of scattering particles per unit volume (e.g. raindrops) and $\overline{\sigma}_{\text{ext}}$ is the mean extinction cross section of the scattering particles. Due to the small size of the laser beam, N and $\overline{\sigma}_{\text{ext}}$ in a specific volume element of the beam are subject to uncertainty, and more precisely it holds:

$$\alpha(R) = \frac{1}{V_{\text{beam}}(R)} \cdot \sum_{i=1}^{n(R)} \sigma_{\text{ext},i} \tag{5.13}$$

where $V_{\text{beam}}(R)$ is a volume element of the beam at distance R, $n(R)$ is the number of scattering particles in the volume element and $\sigma_{\text{ext},i}$ is the extinction cross section of particle i. $\beta(R)$ is defined equivalently by replacing $\sigma_{\text{ext},i}$ with the backscattering cross section $\sigma_{\text{back},i}$.

The extinction and backscattering cross sections of a particle depend on its size as well as the wavelength λ_{wave} of the emitted laser pulse and are defined with the following relationships [255, 258]:

$$\sigma_{\text{ext}}(D_{\text{drop}}, \lambda_{\text{wave}}) = Q_{\text{ext}}(D_{\text{drop}}, \lambda_{\text{wave}}) \cdot \frac{\pi \cdot {D_{\text{drop}}}^2}{4} \tag{5.14}$$

$$\sigma_{\text{back}}(D, \lambda_{\text{wave}}) = Q_{\text{back}}(D_{\text{drop}}, \lambda_{\text{wave}}) \cdot \frac{\pi \cdot {D_{\text{drop}}}^2}{4} \tag{5.15}$$

where $Q_{\text{ext}}(D_{\text{drop}}, \lambda_{\text{wave}})$ and $Q_{\text{back}}(D_{\text{drop}}, \lambda_{\text{wave}})$ are the extinction and backscattering efficiencies, respectively. D_{drop} is the drop diameter. $Q_{\text{ext}}(D_{\text{drop}}, \lambda_{\text{wave}})$ and $Q_{\text{back}}(D_{\text{drop}}, \lambda_{\text{wave}})$ are calculated with Mie theory. Mie theory provides a solution to Maxwell's equations which describe the scattering of electromagnetic radiation. [254, 255]

For the sake of brevity, Mie theory is not reviewed here. Instead the reader is referred to standard textbooks for mathematical details on Mie theory [255, 256]. We implement a widely used Matlab routine developed by [259] based on a code in the appendix of [255] for the calculations of $Q_{\text{ext}}(D_{\text{drop}}, \lambda_{\text{wave}})$ and $Q_{\text{back}}(D_{\text{drop}}, \lambda_{\text{wave}})$. The relevant results of these calculations are presented in Figure 5.4 in function of D_{drop} for $\lambda_{\text{wave}} = 905 \cdot 10^{-9}$ m, the wavelength most common in automotive lidars [102, 104].

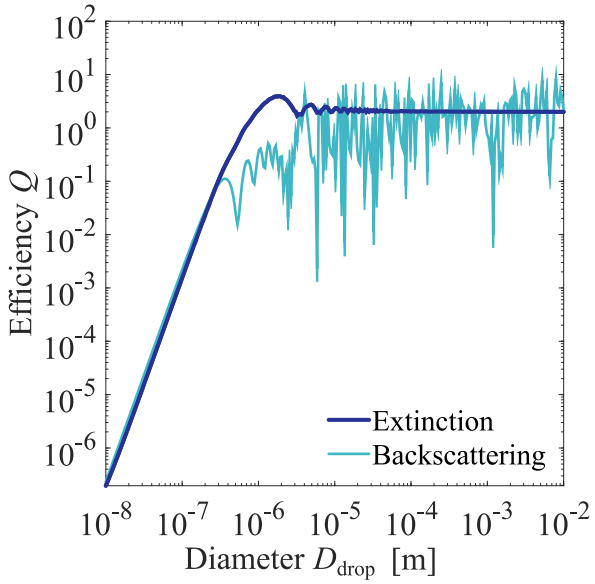

Figure 5.4 **The extinction $Q_{\mathrm{ext}}(D_{\mathrm{drop}}, \lambda_{\mathrm{wave}})$ and backscattering $Q_{\mathrm{back}}(D_{\mathrm{drop}}, \lambda_{\mathrm{wave}})$ efficiencies for a wavelength of $\lambda_{\mathrm{wave}} = 905 \cdot 10^{-9}$ m in dependence of a particle's diameter D_{drop}. Calculations based on [259]. Taken and adapted from [105].**

5.2.4 Probabilistic Modeling of Rainfall

The aim is to estimate the influence of precipitation on lidar performance by means of simulation. As is apparent from Section 5.2.3, key rainfall properties that influence lidar performance are the number of drops in the optical channel and the corresponding drop diameters D_{drop}. Simplifying, we neglect the shape of the drops, which in accordance with Mie's theory is assumed spherical.

The variability in the rain drop size D_{drop} has been extensively studied [260–263]. These studies empirically determine drop size distributions (DSD) with different types of sensors [263]. A DSD describes the number and sizes of drops in a given volume element [260–263]. The DSD tends to depend on the rainfall intensity I, therefore a DSD can empirically be parameterized in terms of I [260, 262–264]. We apply an exponential drop size distribution (DSD), which is found to often accurately describe the temporally averaged DSD, i.e. the distribution of D_{drop} [260, 261, 263]:

$$N(D_{\mathrm{drop}}) = 8000 \cdot \exp(-4.1 \cdot I^{-0.21} \cdot D_{\mathrm{drop}}) \tag{5.16}$$

here I is the rainfall intensity in [mm/h] and D_{drop} is the drop diameter in [mm]. $N(D_{\mathrm{drop}}) \cdot dD_{\mathrm{drop}}$ are the number of drops with diameter between D_{drop} and $D_{\mathrm{drop}} + dD_{\mathrm{drop}}$ per m^3. We study the performance of a lidar conditional on rainfall intensity, therefore we do not model the uncertainty in I.

While many studies empirically confirm the DSD in Eq. (5.16), it is not universally applicable [260, 261, 263]. Most importantly, the type of rainfall (orographic, stratiform, thunderstorm, shower) influences the DSD due to e.g. different wind speeds [262, 263]. The type of rainfall is further correlated with the geographic location. Hence Eq. (5.16) is a good starting point to evaluate the influence of rainfall on a lidar's detection performance, but a detailed empirical study of DSD for different rainfall types at a given geographic location might increase the accuracy of the simulation. The simulation framework presented here is flexible w.r.t the rainfall model, and can be combined with any other suitable DSD. Eq. (5.16) can therefore depending on the application be replaced. Measuring the DSD is however out of scope of this thesis.

We point out that a three parameter gamma DSD [262] sometimes provides a better fit to empirical DSDs because it is a generalization of the exponential distribution. The parameters of the gamma DSD are however not readily defined in the DSD literature, they have to be derived based on measurements.

The integral over Eq. (5.16) determines the average number μ_N of rain drops in a unit volume [m^{-3}]. Due to the small size of the beam, the actual number of drops n_{drops} present in a beam volume element V_{beam} is subject to uncertainty. The uncertainty in the number of drops n_{drops} in a given volume element of the beam is represented with the Poisson distribution [260]:

$$p_{N_{\mathrm{drops}}}(n_{\mathrm{drops}}) = \exp(-\mu_N \cdot V_{\mathrm{beam}}) \cdot \frac{(\mu_N \cdot V_{\mathrm{beam}})^{n_{\mathrm{drops}}}}{n_{\mathrm{drops}}!} \tag{5.17}$$

The exponential DSD of Eq. (5.16) and the Poisson probability mass function (PMF) for the number of drops according to Eq. (5.17) are shown with exemplarily selected values of $I = 5$ mm/h and $V_{\mathrm{beam}} = 0.001$ m^3 in Figure 5.5.

5.2.5 Stochastic Simulation: Signal Detectability Evaluated with Monte Carlo Simulation

Combining the rainfall properties described in Section 5.2.4 with the electromagnetic scattering and absorption defined by Eqs. (5.13)-(5.15) as well as the lidar model in Eq. (5.2) allows to represent the stochastic influence of precipitation on signal intensity.

Noise in the signal due to unwanted backscattering at the raindrops leads to the problem of signal detectability [206–208], which here consists of optimizing a lidar's detection threshold as a tradeoff between the frequency of false positive detections and the frequency of valid detections caused by relevant objects, see Section 4.1. A specific detection threshold determines the sensor probability of detection $\mathrm{POD}_{\mathrm{lidar}}$ and probability of false alarm $\mathrm{PFA}_{\mathrm{lidar}}$ on the raw data level. Plotting $\mathrm{POD}_{\mathrm{lidar}}$ over $\mathrm{PFA}_{\mathrm{lidar}}$ with varying detection thresholds leads to a receiver operating characteristic (ROC) curve, see Figure 4.4.

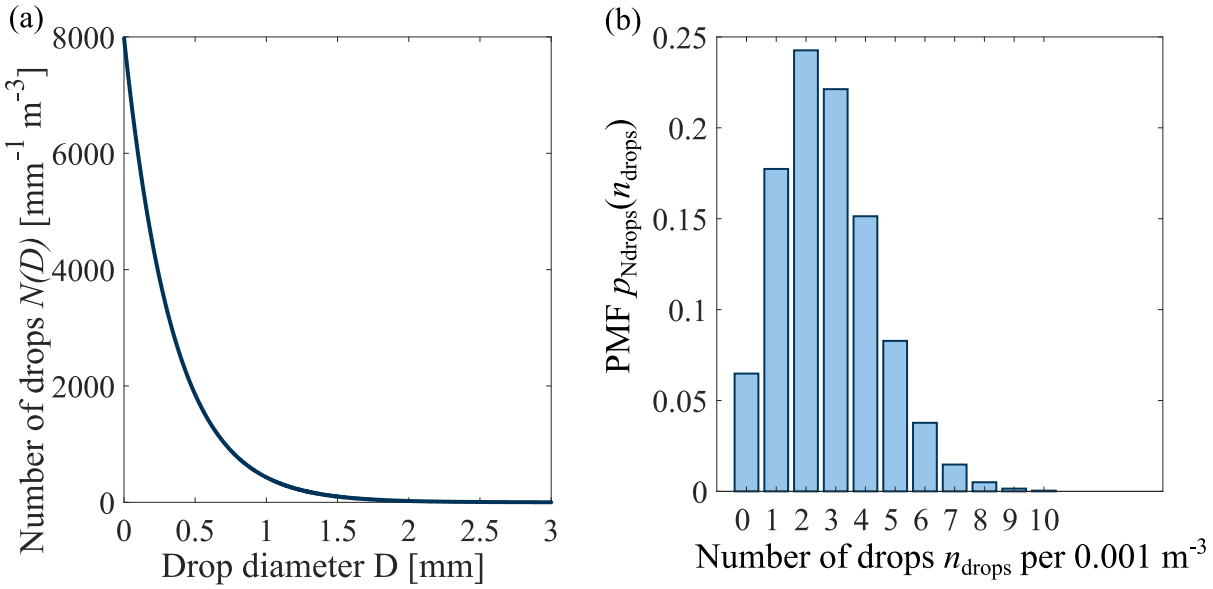

Figure 5.5 Modeling uncertainties in rainfall properties. (a) An exponential drop size distribution with $I = 5$ mm/h and (b) Poisson probability mass function (PMF) of the number of drops n_{drops} in $V_{\mathrm{beam}} = 0.001\ \mathrm{m}^3$.

ROC curves are derived conditional on rainfall intensity I for a target at distance R_0 with Monte Carlo Simulation (MCS). The MCS here artificially generates lidar signals under rainy conditions with the following steps:

1. Discretize the beam into segments of specific length.
2. Sample the number of drops n_{drops} per beam segment from Eq. (5.17).
3. For each drop, sample D_{drop} from Eq. (5.16).
4. For each segment of beam, calculate $\alpha(R)$ and $\beta(R)$ according to Section 5.2.3.
5. Generate the corresponding signal over distance $P_{\mathrm{R}}(R)$ with the lidar model of Eq. (5.2).

Steps 2.-5. are repeated n_{MCS} times for a given rainfall intensity to simulate the stochastic variation in response signal due to random rainfall influences. The fraction of the n_{MCS} generated signals that exceed a specific chosen detection threshold at the target's location R_0 approximates $\mathrm{POD}_{\mathrm{lidar}}$ and the fraction of signals that exceed the detection threshold in the range $R < R_0$, where no target is present, approximates $\mathrm{PFA}_{\mathrm{lidar}}$. With a large number of samples n_{MCS}, the MCS error is negligible. For an explicit estimation of the MCS error we refer to [105, 265].

Varying the detection threshold with the given MCS results leads to a ROC for the underlying R_0 and I. The overall raw data detection performance is fully defined by the ROC curve. To quantify the detection performance with a scalar value, we calculate the area under the ROC curve (AUC). The AUC is a common choice to summarize a ROC curve with a scalar value [208].

5.2.6 Case Study: Evaluating the Effect of Precipitation on a Lidar's Performance

With the developed framework, we study the effect of rainfall on a hypothetical lidar in a case study. In particular, we evaluate the raw data detection performance by analyzing stochastic signal responses of a single laser beam under rainy conditions. At first, we compare two different hypothetical lidar design specifications to demonstrate how the framework allows to optimize the sensor design w.r.t. adverse precipitation effects. Further, we demonstrate how an optimization of the detection threshold allows for an effective filtering of false positive raw detections. Finally, we present the lidar's raw detection performance expressed as AUC, in dependence of a hard target's distance R_0 and rainfall intensity I.

As we are only interested in the received signal due to a hard target relative to unwanted signal caused by rainfall, we set the constants A_R, η_T, η_R in Eq. (5.2) and the peak power P_0 in Eq. (5.3) to one. The emit pulse width is set to $\tau_\mathrm{p} = 5 \cdot 10^{-9}$ s, which is in the range of automotive lidars [102]. We assume a bistatic beam configuration with a displacement of the optical axis $d = 5$ mm. The design specifications of the two hypothetical lidars are summarized in Table 4.1.

Table 5.1 Case study: Design specifications of hypothetical lidar.

Design	Transmitter aperture radius $r_{\mathrm{T},0}$	Receiver aperture radius $r_{\mathrm{R},0}$	Transmit beam divergence γ_T	Receive channel divergence γ_R
1	2 mm	20 mm	0.06°	0.1°
2	2 mm	20 mm	0.03°	0.06°

To demonstrate the tradeoff between true positive and false positive indications of objects by the hypothetical lidars, we study the situation in which a hard target of normal reflectivity $\Gamma = 0.2$ [104] is located at a distance $R_0 = 40$ m and the rainfall intensity is $I = 5$ mm/h. Further, we assume $A_\mathrm{T} > A_\mathrm{B}(R)$ in Eq. (5.11) for all calculations in this case study. Additional to the attenuation caused by the rainfall according to Eq.(5.13), we account for an extinction coefficient of $\alpha = 0.1$ km^{-1} at clear sky [103]. In the simulation, the lidar beam is discretized into segments of 0.1 m length and the total number of signals generated per simulation setting is $n_\mathrm{MCS} = 10^4$. This number of samples limits the relative MCS estimation error on a probability in the magnitudes of 10^{-2} to 10% [105].

Results: Comparing Two Lidar Designs

In a first step, we generate n_MCS synthetic signals with the simulation framework described in Section 5.2.5. Figure 5.6 exemplary presents an evaluation of the spatial impulse response of the optical channel $H_\mathrm{C}(R)$ and the spatial impulse response of the targets $H_\mathrm{T}(R)$ for one of the n_MCS generated signals with lidar design 1 (see Table 4.1). In Figure 5.6a) the maximum response in the optical channel is at about $R = 7$ m because of the influence of the crossover function $\zeta(R)$ in

combination with the $1/R^2$ proportionality of $H_C(R)$. In Figure 5.6b), a peak due to the hard target at $R_0 = 40$ m is visible. The remaining peaks in Figure 5.6b) are due to backscattering at raindrops.

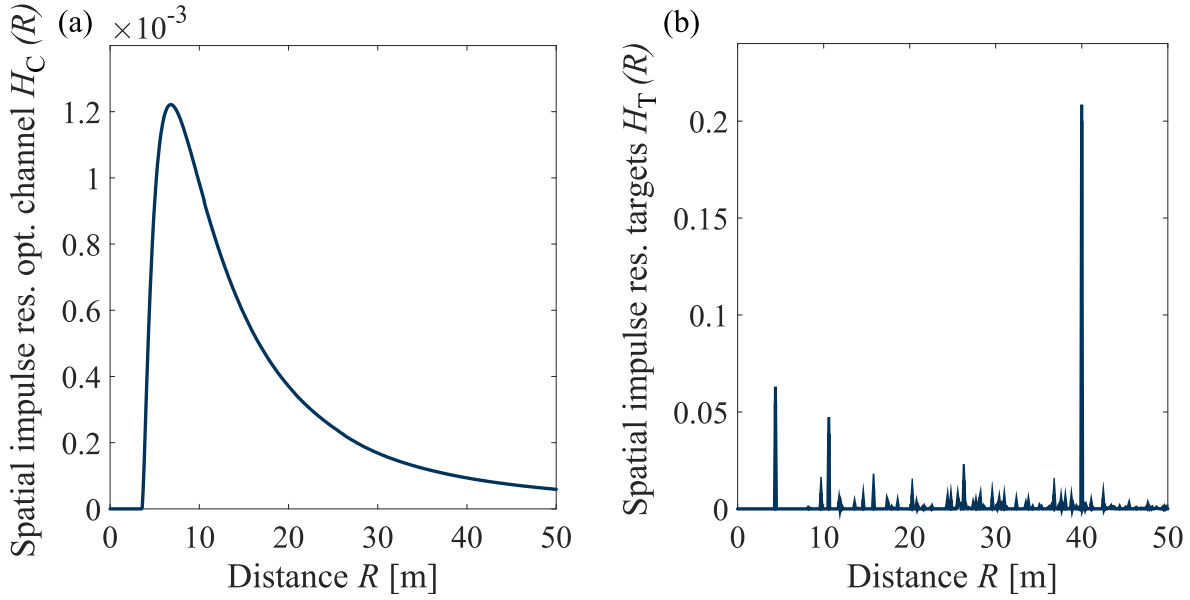

Figure 5.6 Lidar design 1: One sample of (a) spatial impulse response of the optical channel $H_C(R)$ and (b) spatial impulse response of the targets $H_T(R)$ with a rainfall intensity of $I = 5$ mm/h. A hard target is located at $R_0 = 40$ m, all remaining peaks in (b) are due to backscattering at raindrops.

The lidar equation in Eq. (5.2) convolves the product of the impulse response functions in Figure 5.6 with the emitted signal. The resulting signal corresponding to Figure 5.6 is shown in Figure 5.7[65]. Indicated are the response due to the target at $R_0 = 40$ m and the noise due to rainfall in the range $R < 40$ m. The noise is strong only for approximately $R < 20$ m because of the $1/R^2$ proportionality in Eq. (5.2), which is visible in Figure 5.6a). Further illustrated are three static detection threshold values[66]. With threshold 1, the detection threshold is not exceeded. Hence no detection is indicated by the sensor. With threshold 2, only the target is correctly detected at $R_0 = 40$ m. In contrast, with threshold 3, both a FP detection at $R \approx 12$ m as well as the TP target detection at R_0 occur.

[65] This Figure was already presented in Section 4.1.1 to exemplify existence uncertainties.
[66] Static means that the detection threshold is constant w.r.t. the range, or equivalently, the time since laser pulse emission.

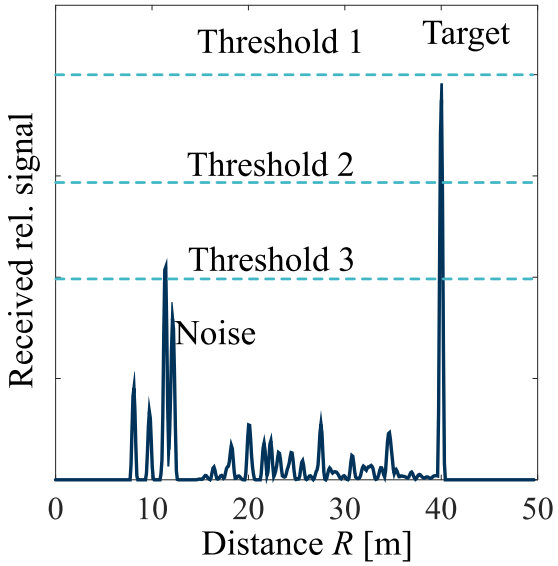

Figure 5.7 **A sample of received relative signal over distance R with lidar design 1 and rainfall intensity $I = 5$ mm/h. A hard target is located at $R_0 = 40$ m, which causes the largest signal peak. The peaks at $R < 40$ m are due to backscattering at raindrops. Depending on the detection threshold, either no detection occurs (threshold 1), only the target would be correctly detected (threshold 2), or a correct target detection and incorrect rainfall detection occur (threshold 3). In analogy [105].**

In a second step, $\mathrm{POD}_{\mathrm{lidar}}$ and $\mathrm{PFA}_{\mathrm{lidar}}$ are estimated with all n_{MCS} signal responses for varying threshold levels. The resulting ROC curves for both sensor designs are illustrated in Figure 5.8.

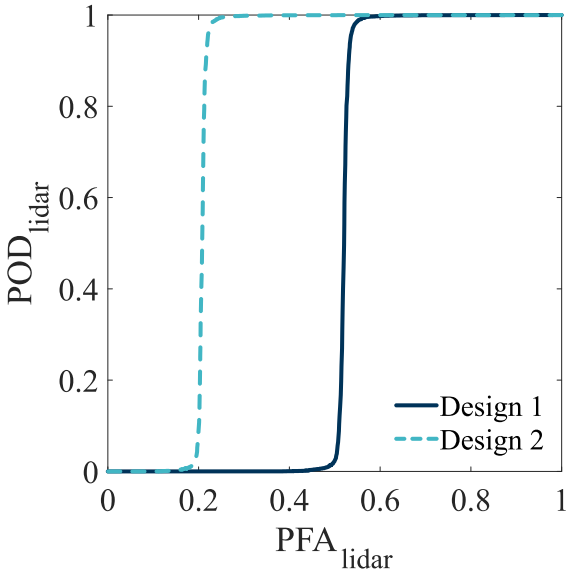

Figure 5.8 **ROC curves of sensor designs 1 and 2 with a rainfall intensity of $I = 5$ mm/h and a target located at $R_0 = 40$ m. The ROC curve shows the receiver's $\mathrm{POD}_{\mathrm{lidar}}$ over $\mathrm{PFA}_{\mathrm{lidar}}$ with varying detection thresholds (e.g. dashed lines in Figure 5.7). With a small detection threshold, $\mathrm{POD}_{\mathrm{lidar}}$ is large but so is $\mathrm{PFA}_{\mathrm{lidar}}$. Increasing the detection threshold leads to a smaller $\mathrm{PFA}_{\mathrm{lidar}}$, but also to a smaller $\mathrm{POD}_{\mathrm{lidar}}$. An optimal receiver has a $\mathrm{POD}_{\mathrm{lidar}} = 1$ and $\mathrm{PFA}_{\mathrm{lidar}} = 0$. Taken and adapted from [105].**

To achieve a $\mathrm{POD_{lidar}} \approx 0.99$ with sensor design 1 would result in a $\mathrm{PFA_{lidar}} \approx 0.55$, which is clearly not acceptable. Decreasing the divergence of the transmit beam and the optical channel according to lidar design 2 would increase the performance of the lidar in this specific situation. With sensor design 2, $\mathrm{PFA_{lidar}} \approx 0.24$ for a $\mathrm{POD_{lidar}} \approx 0.99$. Because of the $1/R^2$ proportionality of the signal, the influence of rainfall is present mostly for small R, see Figure 5.6. Reducing the divergence of the beam results in a lower probability of the beam hitting drops in this critical range and thus decreasing the $\mathrm{PFA_{lidar}}$. This comes at the cost of a larger minimal range, defined by Eq. (5.6). In design 1 the minimal range is $R_1 = 3.6$ m and in design 2 it is $R_1 = 6.4$ m.

Results: Filtering Rainfall Effect

The ROC curves in Figure 5.8 are derived with static detection threshold values of varying magnitudes. Because of the $1/R^2$ proportionality, both the unwanted signal response due to rainfall and the desired signal response due to hard targets are much larger at close distances than at farther distances. One option to filter the adverse rainfall effect is therefore a dynamic threshold proportional to $1/R^2$. The ROC curve for design 1 with such a dynamic threshold is shown in Figure 5.9. Design 1 is selected here because it has a smaller minimum range of $R_1 = 3.6$ m compared with $R_1 = 6.4$ m of design 2. The plot corresponds to the same situation ($R_0 = 40$ m and $I = 5$ mm/h) as in Figure 5.8. By applying the dynamic threshold, a $\mathrm{POD_{lidar}} \approx 0.99$ is achieved with a $\mathrm{PFA_{lidar}} \approx 0.017$.

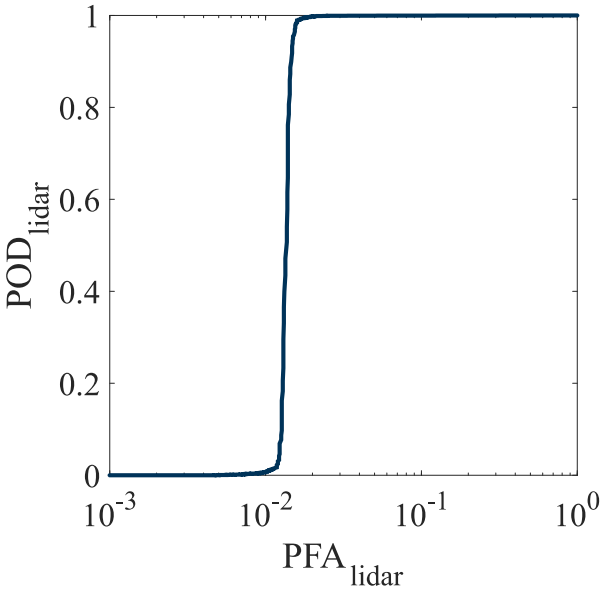

Figure 5.9 ROC curve of sensor design 1 applying a dynamic threshold with a rainfall intensity of $I = 5$ mm/h and a target at $R_0 = 40$ m. Taken and adapted from [105].

The ROC curve in Figure 5.9 estimates the raw detection performance of the lidar considering a single laser pulse. As explained in Section 2.1.3, a lidar for instance scans its environment with multiple laser pulses emitted in different angular directions. Thereafter, object detection and

tracking are performed on the basis of the resulting point cloud (i.e. raw data), which further reduces adverse rainfall influences. Often, a multi cycle heuristic is applied to validate object detections by requiring multiple subsequent detections to gain credibility in the existence of an object [29, 59, 60, 106].

Neglecting the scanning in different directions, it is here assumed that object detections are validated by requiring three subsequent detections. Further, it is assumed that false positives due to rainfall are statistically independent in subsequent measurements. With these assumptions, $\mathrm{PFA}_{\mathrm{lidar}}$ is estimated at the object data level on a highly simplified basis by ${\mathrm{PFA}_{\mathrm{lidar}}}^3 \approx 0.017^3 = 4.9 \cdot 10^{-6}$. However, also $\mathrm{POD}_{\mathrm{lidar}}$ would decrease to ${\mathrm{POD}_{\mathrm{lidar}}}^3 \approx 0.97$ under the assumption of independence if three subsequent detections are required to validate an object detection. These calculations neglect the influence of segmentation algorithms to detect objects in the lidar point cloud and are therefore a rough estimation only.

Results: Rainfall Effect in Function of Target Distance

Figure 5.8 and Figure 5.9 looked at the specific situation of $R_0 = 40$ m and $I = 5$ mm/h. The analysis is repeated with sensor design 1 and the dynamic threshold employed in Figure 5.9 for varying values of target distance R_0 and rainfall intensity I. To summarize the resulting ROC curves with a single scalar, AUC is calculated for each R_0 and I pair. The results are displayed in Figure 5.10.

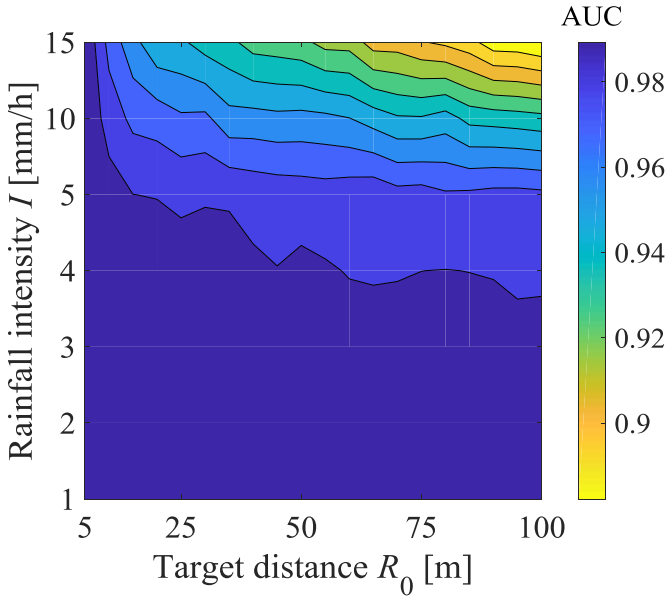

Figure 5.10 Area under ROC curve (AUC) in function of target distance R_0 and rainfall intensity I. The larger the AUC, the better the detection performance of the lidar. With low rainfall intensity and low target distances, the lidar has the largest detection performance. With increasing target distance, and with increasing rainfall intensity, the detection performance of the lidar decreases. Taken and adapted from [105].

The AUC plot indicates that with a small rainfall intensity ($I \leq 4$ mm/h), the performance of the lidar is not impaired substantially. However, for larger rainfall intensities ($I > 4$ mm/h) the lidar performance decreases, especially for larger distances. As a reference, the AUC corresponding to Figure 5.9 is AUC $= 0.986$.

5.2.7 Discussion and Conclusions

We presented a physics-based simulation framework to analyze adverse rainfall effects on lidar sensors. With the simulation, we assessed how sensor design parameters influence adverse rainfall effects on *sensor perception reliability* (Figure 5.8), we devised filtering methods for adverse rainfall effects to optimize *sensor perception reliability* (Figure 5.9) and we quantified *sensor perception reliability* in function of the distance to a valid target and the rainfall intensity (Figure 5.10). Physics-based simulations are therefore an opportunity to assess *sensor perception reliability* accounting for context variables, in this case rainfall.

In the presented simulation framework, we modeled rainfall effects separately from other noise sources (i.e. context variables), such as different weather conditions (e.g. snowfall and fog) or background light. Of special interest in an automotive application is the influence of spray water from the road surface. The spray water might lead to dirt on a sensor's cover [101], alter the distribution of rainfall drop size and is subject to spatial uncertainty (i.e. it is strong directly behind vehicles and decreases in strength in areas where no vehicle is located).

The simulation framework models the response of a single laser beam. Therefore, the laser performance is evaluated on the raw data level. Tracking and multi-cycle heuristics further allow to filter the false positive errors made on the raw data level. Extending the framework could allow to estimate a sensor's performance on the object data level.

For instance, combining the effects of other context variables with the presented framework in a ray tracer is an opportunity for physics-based lidar simulations to virtually estimate the performance of object detection and tracking. In a ray-tracing simulation, the sender and receiver could be simulated as a module. This module could be a black box in case the intellectual property of a third party needs to be protected. An intensity value is assigned to the rays corresponding to a discretization of $P_{\mathrm{T}}(t)$. A MCS is then performed for each ray, similar to Section 5.2.5 by dividing each ray into segments for which $\alpha(R)$ and $\beta(R)$ are calculated. The received signal $P_{\mathrm{R}}(R)$ is then derived for each ray with Eq. (5.2). In a next step, the obtained $P_{\mathrm{R}}(R)$ is provided to the sensor receiver module. The module processes the signal to provide the simulated lidar raw

data. Next, the simulated raw data can be combined with a SiL that simulates the object extraction and tracking in order to evaluate the performance of the whole sensing system. [105][67]

The simulation framework for rainfall effects is here derived by combining well known physics theories (lidar equation and Mie theory) with empirical findings on rainfall properties. Despite the framework's theoretical and empirical foundation, future work could try to validate the proposed simulation framework by means of measurements and experiments, which is not in the scope of this thesis. To do so, the effect of precipitation has to be separated from other noise sources such as a varying background illumination. Alternatively, rainfall could be generated artificially with a rainfall simulator (i.e. a piping system with different nozzles to reproduce a realistic drop size distribution), similar to an experiment described in [104].

Considering the specific example of simulating adverse rainfall effects on lidar sensors, we conclude that simulation methods based on the physical principles behind a sensing technology constitute useful development tools. Simulation methods such as the presented framework are an opportunity to estimate *sensor perception reliability* early in the development process, when testing the sensors in reality is not yet possible. The simulations allow to account for context variables such as rainfall. Due to the complexities of environment perception, it is however not suggested that these simulation methods replace real sensor tests on proving grounds or in the field for a final validation of *perception reliability* (see also the discussion in Section 3.3.2).

5.3 Learning Perception Reliability in Controlled Field Tests on Proving Grounds

This Section is based on our publications in [39, 161] and is partly taken from our publication in [39].

An option to test and validate *perception reliability* is to conduct controlled field tests on proving grounds, following a catalogue of detailed scenarios [40, 44, 56, 178–181]. Significant research effort is made to identify and collect relevant test cases for ADS [181]. Limitations of demonstrating the *perception reliability* of an ADS with *higher levels of driving automation* on proving grounds through scenario based testing are discussed in Section 3.3.

To (partly) overcome the limitations of scenario based testing, we develop a statistical framework to learn *(sensor) perception reliability* from controlled field tests on proving grounds. Two challenges for such a learning framework are potential statistical dependencies over subsequent

[67] This paragraph is due to Jose Roberto Vargas Rivero in [105].

time steps and a variable perception performance (see Section 3.1.4). The variable perception performance leads to *higher order uncertainties* (see Section 4.1.4).

Because of *higher order uncertainties*, it is not sufficient to evaluate the *(sensor) perception reliability* once under controlled conditions on proving grounds. Instead, one has to account for the variability in the context variables **E**, otherwise the estimated *(sensor) perception reliability* is biased (see Section 3.1.4). Furthermore, a prerequisite for learning perception reliability metrics (Section 4.1) with standard statistical inference methods are statistically independent and identically distributed observations (i.e. object indications conditional on a real object being present or not, deviation in state quantities, object classes conditional on the true class). This prerequisite is not fulfilled because of potential statistical dependencies over subsequent time steps and a variable perception performance.

To address the dependence structure in the data and the variable perception performance, we propose a Bayesian hierarchical regression model in Section 5.3.1 to exploit conditional independence properties, which enables the use of standard statistical inference methods. The hierarchical regression model learns the functional relationship between (known) context variables **E** and perception reliability metrics. Section 5.3.2 combines this functional relationship with the exposure $f_{\mathbf{E}}(\mathbf{e})$ towards the context variables to predict global perception reliability metrics, or alternatively, the distribution of observables. The prediction probabilistically accounts for all values of $\mathbf{e}$ in $f_{\mathbf{E}}(\mathbf{e})$, hence it is not required to conduct an infinite number of test cases. The application of this framework is demonstrated with a simple case study in Section 5.3.3, which examines the influence of temperature on a lidar's *perception reliability*.

5.3.1 Learning the Influence of Context Variables on Perception Reliability

The hierarchical regression model to learn the functional relationship between known context variables **E** and relevant reliability metrics is developed in reference to [114]. The reliability metrics are generically denoted with $\boldsymbol{\theta}$ in the following, which are model parameters to be learned. Depending on the respective uncertainty domain, $\boldsymbol{\theta}$ include the reliability metrics as described in Section 4.1, and potentially parameters for sensor error dependencies in case multiple sensors are examined simultaneously. The generic structure underlying the hierarchical model is explained with Figure 5.11.

The subscript j identifies a specific discrete time interval of specific length (e.g. 5 s), in which it is assumed that the model parameters $\boldsymbol{\theta}_j$ and the context variables $\mathbf{e}_j$ are constant. In each time interval j one observes K data points $\mathbf{x}_j = [\mathbf{x}_{j,k}]_{k=1}^{K}$, i.e. the subscript k identifies a specific data point in the interval j. The data **X** could for instance be binary indications of objects, the deviation in estimated state quantities from their true value, or object classes.

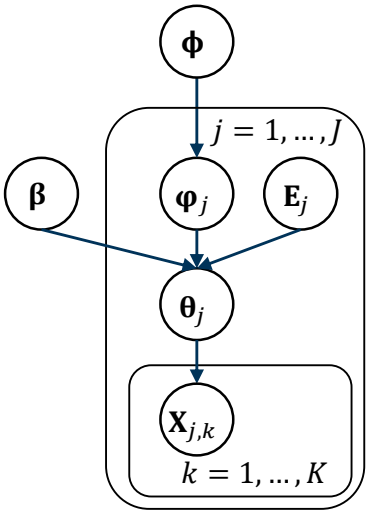

Figure 5.11 Bayesian network [116] for learning the effect of context variables E on *perception reliability* expressed through the model parameters θ. X is the observable data, β quantifies the influence of E on θ, φ are random effects not covered through E and β. ϕ are hyperparameter describing the variability in φ. Taken and adapted from [39].

The data $\mathbf{x}_j$ in a given time interval j can be assumed to be approximately conditionally independent and identically distributed given $\boldsymbol{\theta}_j$:

$$\mathbf{x}_{j,k}|\boldsymbol{\theta}_j \sim f_{\mathbf{X}|\boldsymbol{\theta}_j}(\mathbf{x}_{j,k}|\boldsymbol{\theta}_j) \tag{5.18}$$

$f_{\mathbf{X}|\boldsymbol{\theta}_j}(\mathbf{x}_{j,k}|\boldsymbol{\theta}_j)$ is the PDF of $\mathbf{x}_{j,k}$ for $k = 1, \dots, K$ in time interval j.

To account for *higher order uncertainties*, $\boldsymbol{\theta}_j$ is formulated in function $g(\,)$ of a linear combination of known context variables $\mathbf{e}_j$ and in function of a random effect parameter $\boldsymbol{\varphi}_j$:

$$\boldsymbol{\theta}_j = g(\mathbf{e}_j^{\mathrm{T}} \cdot \boldsymbol{\beta} + \boldsymbol{\varphi}_j) \tag{5.19}$$

$\boldsymbol{\beta}$ describes how $\mathbf{e}_j$ influences $\boldsymbol{\theta}_j$. $\mathbf{e}_j^{\mathrm{T}}$ is the transposed vector of context variables' states. Relevant context variables $\mathbf{E}$ are identified with the procedure in Section 5.1. As one cannot know and quantify all relevant context variables, the random effect parameter $\boldsymbol{\varphi}_j$ quantifies the influence of all factors on $\boldsymbol{\theta}$, which are not explicitly included in $\mathbf{E}$. $\boldsymbol{\varphi}_j$ is described with a statistical model $f_{\boldsymbol{\varphi}|\boldsymbol{\phi}}$ defined by the hyperparameter $\boldsymbol{\phi}$:

$$\boldsymbol{\varphi}_j|\boldsymbol{\phi} \sim f_{\boldsymbol{\varphi}|\boldsymbol{\phi}}(\boldsymbol{\varphi}_j|\boldsymbol{\phi}) \tag{5.20}$$

With the hierarchical model structure of Figure 5.11 and its implied conditional independence properties, the parameters $\boldsymbol{\beta}, \boldsymbol{\phi}$ and $\boldsymbol{\varphi}$ are learned with Bayesian interference [114] from the collection of observed data $\mathbf{x} = [\mathbf{x}_j]_{j=1}^{J}$ and their corresponding context variables $\mathbf{e} = [\mathbf{e}_j]_{j=1}^{J}$:

$$\begin{aligned} &f(\boldsymbol{\beta}, \boldsymbol{\varphi}, \boldsymbol{\phi}|\mathbf{x}, \mathbf{e}) \propto \\ &\propto f(\boldsymbol{\beta}, \boldsymbol{\phi}) \cdot \prod_{j=1}^{J} f_{\boldsymbol{\varphi}|\boldsymbol{\phi}}\left(\boldsymbol{\varphi}_j|\boldsymbol{\phi}\right) \cdot \prod_{k=1}^{K} f_{\mathbf{X}|\boldsymbol{\theta}_j}(\mathbf{x}_{j,k}|\boldsymbol{\theta}_j = g(\mathbf{e}_j^{\mathrm{T}} \cdot \boldsymbol{\beta} + \boldsymbol{\varphi}_j)) \end{aligned} \tag{5.21}$$

where $f(\boldsymbol{\beta}, \boldsymbol{\varphi}, \boldsymbol{\phi}|\mathbf{x}, \mathbf{e})$ is the joint posterior of $\boldsymbol{\beta}$, $\boldsymbol{\varphi}$, $\boldsymbol{\phi}$ given $\mathbf{x}$ and $\mathbf{e}$. $f(\boldsymbol{\beta}, \boldsymbol{\phi})$ is the joint prior of $\boldsymbol{\beta}$ and $\boldsymbol{\phi}$. $f_{\boldsymbol{\varphi}|\boldsymbol{\phi}}(\boldsymbol{\varphi}_j|\boldsymbol{\phi})$ is the likelihood of the random effects as in Eq. (5.20). $f_{\mathbf{X}|\boldsymbol{\theta}_j}(\mathbf{x}_{j,k}|\boldsymbol{\theta}_j = g(\mathbf{e}_j^{\mathrm{T}} \cdot \boldsymbol{\beta} + \boldsymbol{\varphi}_j)$ is the likelihood of the model parameters, defined by Eqs. (5.18)-(5.19).

Given the influencing factors $\mathbf{e}_j$, the model in Figure 5.11 is fully defined by the joint PDF of $\boldsymbol{\phi}$ and $\boldsymbol{\beta}$. Therefore, inference of the random effects $\boldsymbol{\varphi}_j$ in a particular time interval j is not important for a reliability prediction. Eq. (5.21) can be marginalized over $\boldsymbol{\varphi} = [\boldsymbol{\varphi}_j]_{j=1}^{J}$ to obtain the relevant posterior $f(\boldsymbol{\beta}, \boldsymbol{\phi}|\mathbf{x}, \mathbf{e})$:

$$f(\boldsymbol{\beta}, \boldsymbol{\phi}|\mathbf{x}, \mathbf{e}) \propto \int_{\boldsymbol{\varphi}} f(\boldsymbol{\beta}, \boldsymbol{\varphi}, \boldsymbol{\phi}|\mathbf{x}, \mathbf{e})\mathrm{d}\boldsymbol{\varphi} \tag{5.22}$$

In Section 5.3.3 we demonstrate with a simple example how to learn $\boldsymbol{\phi}$ and $\boldsymbol{\beta}$ from data $\mathbf{x}$, i.e. how to estimate the posterior $f(\boldsymbol{\beta}, \boldsymbol{\phi}|\mathbf{x}, \mathbf{e})$ in Eq. (5.22).

5.3.2 Predicting Perception Reliability

Ultimately, the goal is to predict future observable data $\tilde{\mathbf{x}}$ given the test data $\mathbf{x}$ and $\mathbf{e}$, accounting for the exposure (i.e. occurrence frequency) $f_{\mathbf{E}}(\tilde{\mathbf{e}})$ of the context variables $\mathbf{E}$[68]. This prediction is obtained in analogy to Eq. (3.5) by marginalizing over all parameters in the joint distribution in Figure 5.11, given the evidence on $\mathbf{x}$ and $\mathbf{e}$:

$$\begin{aligned} &f(\tilde{\mathbf{x}}|\mathbf{x}, \mathbf{e}) = \\ &= \int f(\boldsymbol{\beta}, \boldsymbol{\phi}|\mathbf{x}, \mathbf{e}) \cdot f_{\boldsymbol{\varphi}|\boldsymbol{\phi}}(\boldsymbol{\varphi}|\boldsymbol{\phi}) \cdot f_{\mathbf{E}}(\tilde{\mathbf{e}}) \cdot f_{\mathbf{X}|\boldsymbol{\theta}}\left(\tilde{\mathbf{x}}|g(\tilde{\mathbf{e}}^{\mathrm{T}} \cdot \boldsymbol{\beta} + \boldsymbol{\varphi})\right) d\boldsymbol{\phi} d\boldsymbol{\varphi} d\boldsymbol{\beta} d\tilde{\mathbf{e}} \end{aligned} \tag{5.23}$$

where $f(\tilde{\mathbf{x}}|\mathbf{x}, \mathbf{e})$ is the posterior predictive distribution of the observables and $f_{\mathbf{X}|\boldsymbol{\theta}}\left(\tilde{\mathbf{x}}|\boldsymbol{\theta} = g(\tilde{\boldsymbol{e}}^{\mathrm{T}} \cdot \boldsymbol{\beta} + \boldsymbol{\varphi})\right)$ is the sampling distribution in Eq.(5.18) of $\mathbf{x}$ for given model

[68] The posterior $f(\boldsymbol{\beta}, \boldsymbol{\phi}|\mathbf{x}, \mathbf{e})$ describes the uncertainty in the model parameters $\boldsymbol{\beta}$ and $\boldsymbol{\phi}$ given the observed data $\mathbf{x} = [\mathbf{x}_j]_{j=1}^{J}$ and the corresponding observed context variables $\mathbf{e} = [\mathbf{e}_j]_{j=1}^{J}$ in a test. The notation $\tilde{\mathbf{e}}$ makes clear that the prediction in Eq. (5.23) is not w.r.t. $f(\boldsymbol{\beta}, \boldsymbol{\phi}|\mathbf{x}, \mathbf{e})$ but to the sampling distribution in Eq. (5.18), i.e. to $f_{\mathbf{X}|\boldsymbol{\theta}}\left(\tilde{\mathbf{x}}|g(\tilde{\mathbf{e}}^{\mathrm{T}} \cdot \boldsymbol{\beta} + \boldsymbol{\varphi})\right)$.

parameters $\boldsymbol{\theta} = g(\tilde{\boldsymbol{e}}^{\mathrm{T}} \cdot \boldsymbol{\beta} + \boldsymbol{\varphi})$. The *(sensor) perception reliability* can be derived from $f(\tilde{\mathbf{x}}|\mathbf{x}, \mathbf{e})$, e.g. in terms of the standard deviation of state quantities or POD and PFA.

Alternatively one could also directly predict the model parameters:

$$\widehat{\boldsymbol{\theta}} = \int f(\boldsymbol{\beta}, \boldsymbol{\phi}|\mathbf{x}, \mathbf{e}) \cdot f_{\boldsymbol{\varphi}|\boldsymbol{\phi}}(\boldsymbol{\varphi}|\boldsymbol{\phi}) \cdot f_{\mathbf{E}}(\tilde{\mathbf{e}}) \cdot g(\tilde{\mathbf{e}}^{\mathrm{T}} \cdot \boldsymbol{\beta} + \boldsymbol{\varphi})\, d\boldsymbol{\phi} d\boldsymbol{\varphi} d\boldsymbol{\beta} d\mathbf{e} \tag{5.24}$$

where $\widehat{\boldsymbol{\theta}}$ is the predicted global mean of the model parameters w.r.t. *higher order uncertainties* over time intervals j. As an alternative to evaluating the mean $\widehat{\boldsymbol{\theta}}$**,** the PDF of $\boldsymbol{\theta}_j$ could be derived with Markov Chain Monte Carlo (MCMC) sampling [114, 115, 266, 267].

5.3.3 Case Study: Quantifying the Influence of Temperature on a Lidar's Accuracy

The learning framework of Section 5.3.1 is exemplarily applied in a simple case study to investigate the effect of increased sensor temperatures on a lidar's accuracy in object position estimations. The utilized lidar controls the angular measurement directions with a mechanically rotating mirror [268]. A heating pad was attached to the sensor casing to control the temperature. Object position deviations ΔX in the direction perpendicular to the sensor are evaluated, i.e. the case study evaluates the lidar's perception reliability in the domain of state uncertainties, see Section 4.1. In this example, temperature is the only context variable we consider.

Experiment Set-Up

To evaluate ΔX, a vehicle equipped with the lidar is parked perpendicular to a wooden wall at a distance of 37.84 m. The simple experiment set-up is depicted in Figure 5.12.

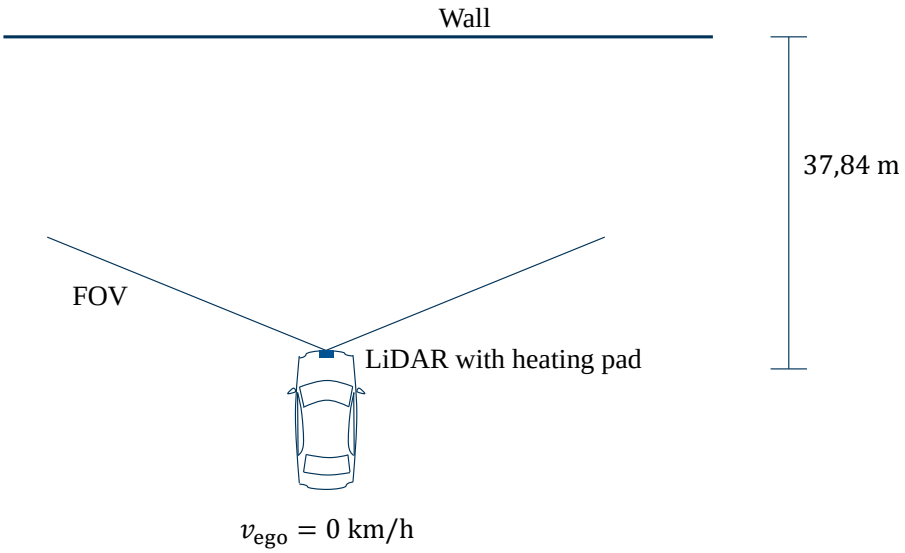

Figure 5.12 Experiment set-up to learn the influence of temperature on a lidar's object position estimation accuracy according to Section 5.3.1.

The distance to the wall was not altered during the test. This means, the ego-velocity of the vehicle is $v_{\text{ego}} = 0\,\text{km/h}$ during the complete test. The weather during the test was cloudy but without precipitation. Due to stable weather conditions, it is not expected that the weather causes variations in the lidar's performance during the test.

In total $24\,\text{min}\,36\,\text{s}$ of measurements were recorded. During the measurements, the heating pad was activated at $t \approx 7\,\text{min}$ to increase the sensor's temperature. The recorded object position deviations Δx of the wall in the direction perpendicular to the sensor are plotted over time in Figure 5.13a). The corresponding temperature of the sensor casing is presented in Figure 5.13b).

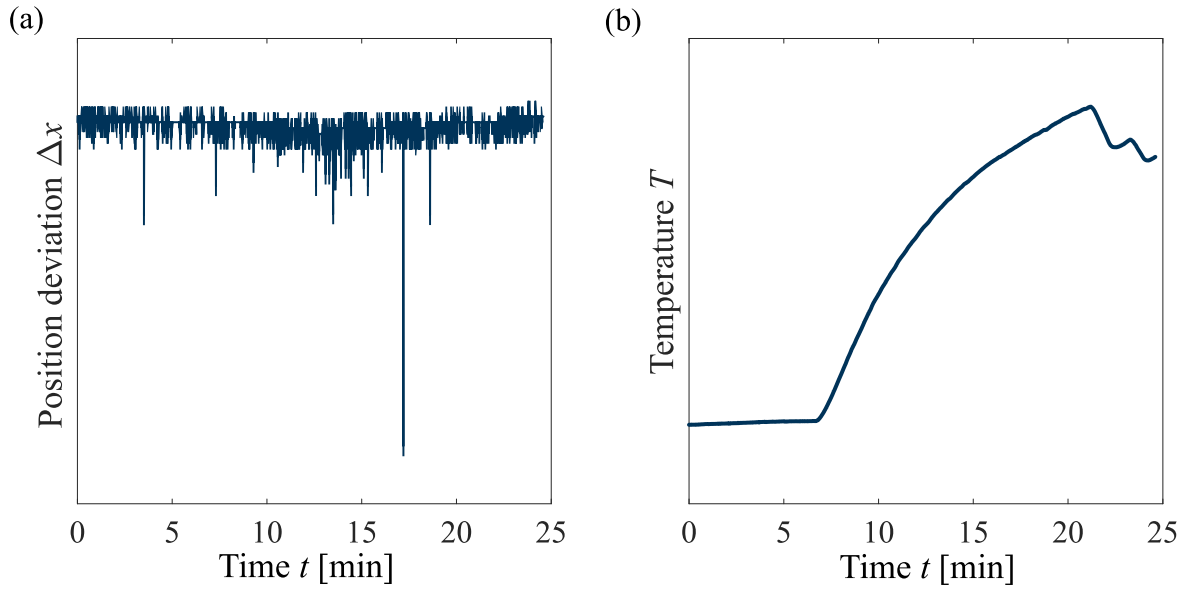

Figure 5.13 (a) Recorded object position deviations Δx of a wall (see Figure 5.12) over time. (b) Corresponding temperature of the lidar casing. The heating pad was activated at $t \approx 7$ min.

It is difficult to draw conclusions on the influence of temperature on the accuracy in Δx by solely visually comparing Figure 5.13a) and Figure 5.13b). A comparatively large value of Δx is observed at time $t \approx 17\,\text{min}$.

Statistical Model

The data is grouped into time intervals of $t = 30\,\text{s}$ length to apply the learning framework of Section 5.3.1, resulting in $J = 50$ time intervals. The position deviations $\Delta x_{j,k}$ in interval j are modeled with a normal distribution for $k = 1, \dots, K$:

$$\Delta x_{j,k} | \mu_j, \sigma_j \sim N(\mu_j, \sigma_j^2) \tag{5.25}$$

where the model parameters $\boldsymbol{\theta}_j = [\mu_j, \sigma_j]$ are the mean and standard deviation of ΔX_j in block j. Eq. (5.25) is the distribution of the data as in Eq. (5.18).

To account for *higher order uncertainties* in the mean of ΔX_j, the μ_j in the different time intervals $j = 1, \dots J$ are assumed to be samples from a normal distribution:

$$\mu_j | \mu_\mu, \sigma_\mu \sim N(\mu_\mu, \sigma_\mu^2) \tag{5.26}$$

$\boldsymbol{\phi}_\mu = [\mu_\mu, \sigma_\mu]$ are hyperparameter as in Eq. (5.20). μ_μ is the global mean, i.e. the mean of μ_j over j. σ_μ is the standard deviation of μ_j over j. With this modeling choice, it is assumed that there is no systematic influence of temperature on the position deviations but that the mean is variable due to random effects.

The standard deviations σ_j of ΔX_j in the different time intervals $j = 1, \dots J$ are modeled in function of the standardized[69] temperature T_{standard}. Additionally, a random effect $\varphi_{\sigma,j}$ accounts for variability in σ_j, which is not accounted for by temperature T:

$$\sigma_j = \sqrt{\exp(-\beta_0 - \beta_1 \cdot T_{\text{standard}} - \varphi_{\sigma,j})} \tag{5.27}$$

The exponential in Eq. (5.27) ensures that $\sigma_j > 0$.

Finally, the random effect $\varphi_{\sigma,j}$ is modeled with a normal distribution:

$$\varphi_{\sigma,j} | \tau_\varphi \sim N(0, 1/\tau_\varphi) \tag{5.28}$$

where the hyperparameters are $\boldsymbol{\phi}_\sigma = [\mu_\sigma, \sigma_\sigma^2] = [0, 1/\tau_\varphi]$.

The prior distribution $f(\boldsymbol{\beta}, \boldsymbol{\phi})$ in Eq. (5.21) is naively selected to be weakly-informative for all model parameters with a uniform prior.

Results

The model parameters are learned from the data in Figure 5.13 in analogy to Eq. (5.21) with MCMC [114, 115, 266, 267] using OpenBugs [269]. The results are presented in arbitrary units.

The inference of the mean of Δx_j in the different time intervals j is presented in Figure 5.14a) to demonstrate the variability in μ_j. Points are a posteriori parameter estimates and lines are 95% credible intervals. A comparison of Figure 5.14a) with empirical values of μ_j reveals that the learning framework of Section 5.3.1 correctly determines the mean values. The credible intervals indicate that estimation uncertainties for μ_j are generally small, except for time interval $j = 35$.

[69] The standardized temperature has a mean of 0 and a standard deviation of 1. Regression analysis and model training is more stable when standardizing covariates (i.e. features) [114, 115].

Figure 5.13a) shows that a comparatively large position deviation Δx occurred in time interval $j = 35$, which translates into the larger uncertainty in μ_{35}.

In units of Figure 5.14a), the global mean is estimated as $\mu_\mu = 0.76$ with a 95% credible interval of $[0.64, 0.88]$. Visually comparing Figure 5.14a) with Figure 5.13b) leads to the hypothesis of a quadratic relationship between μ_j and temperature T. This potential relationship is not further investigated nor modeled because the variability of μ_j is negligible in absolute values. In practice, the influence of T on μ_j should be accounted for.

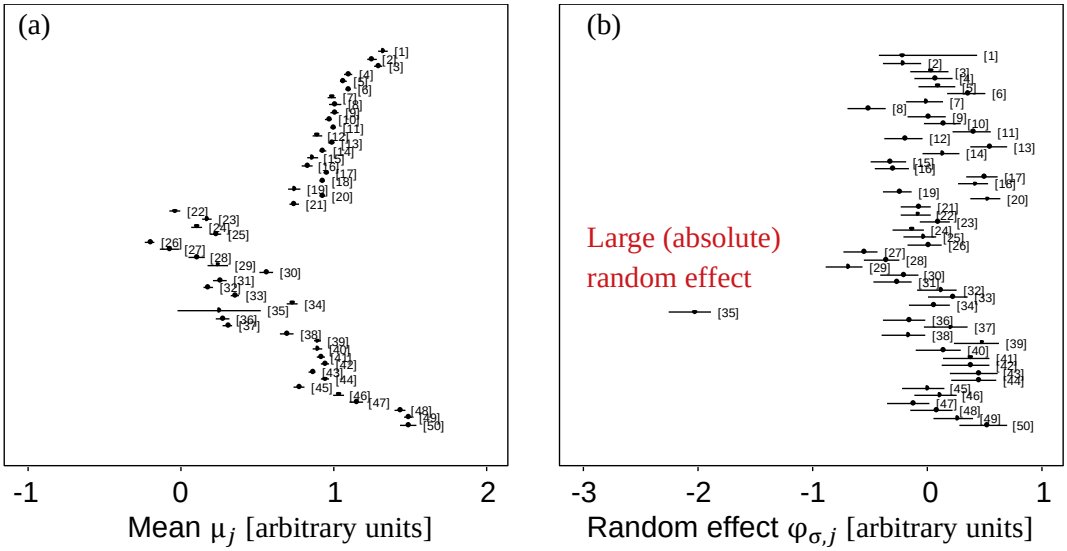

Figure 5.14 Caterpillar plots: (a) mean μ_j of Δx_j and (b) random effect $\varphi_{\sigma,j}$ of the standard deviation σ_j in interval $j = 1, \ldots 50$. Points are a posteriori mean estimates and lines 95% credible intervals.

The estimation of random effects $\varphi_{\sigma,j}$ is presented in Figure 5.14b). The variability in Δx_{35} caused by the large position deviation in time interval $j = 35$ is explained with a large (absolute) random effect of $\varphi_{\sigma,j} = -2.06$ and not by the temperature influence. A random effect of $\varphi_{\sigma,j} = -2.06$ increases the standard deviation σ_j by a factor of 2.79. Further investigating the data in time interval $j = 35$ reveals that the large position deviation is due to an error in the object bounding box and not due to temperature. This is correctly identified by the random effect.

To assess *sensor perception reliability* in the state uncertainty domain (Section 4.1.2), the functional relationship between temperature T and the standard deviation σ_j of ΔX_j according to Eq. (5.27) is investigated. Results of the parameter estimation are summarized in Table 5.2. A central question is, whether the temperature has an influence on the variability of the position

deviations Δx? Because β_1 is with more than 99.99 % credibility smaller than zero, it is concluded that T is correlated with σ_j[70].

Table 5.2 Results of the parameter inference in arbitrary units. From [161].

Parameter	Mean	2.5% quantile	Median	97.5% quantile
β_0	1.00	0.97	1.00	1.04
β_1	−0.32	−0.56	−0.32	−0.10
$\varphi_{\sigma,2}$	−0.20	−0.40	−0.20	0.03
$\varphi_{\sigma,35}$	−2.06	−2.27	−2.05	−1.90
τ_φ	5.51	3.53	5.43	7.95
μ_μ	0.76	0.64	0.76	0.88
σ_μ	0.44	0.38	0.46	0.56

The learned relationship between T and σ_j is presented in Figure 5.15a) together with observed σ_j. The relationship adequately represents observed σ_j. The credible intervals in Figure 5.15a) account for the influence of T on σ_j, i.e. they do not account for statistical uncertainties in random effects $\varphi_{\sigma,j}$ or the hyperparameter τ_φ. Conditional on the a posteriori mean value of τ_φ, the random effects $\varphi_{\sigma,j}$ have with 95% probability a multiplicative effect between 0.66 and 1.52 on σ_j. The scattering of observed σ_j around the relationship in Figure 5.15a) is due to random effects $\varphi_{\sigma,j}$.

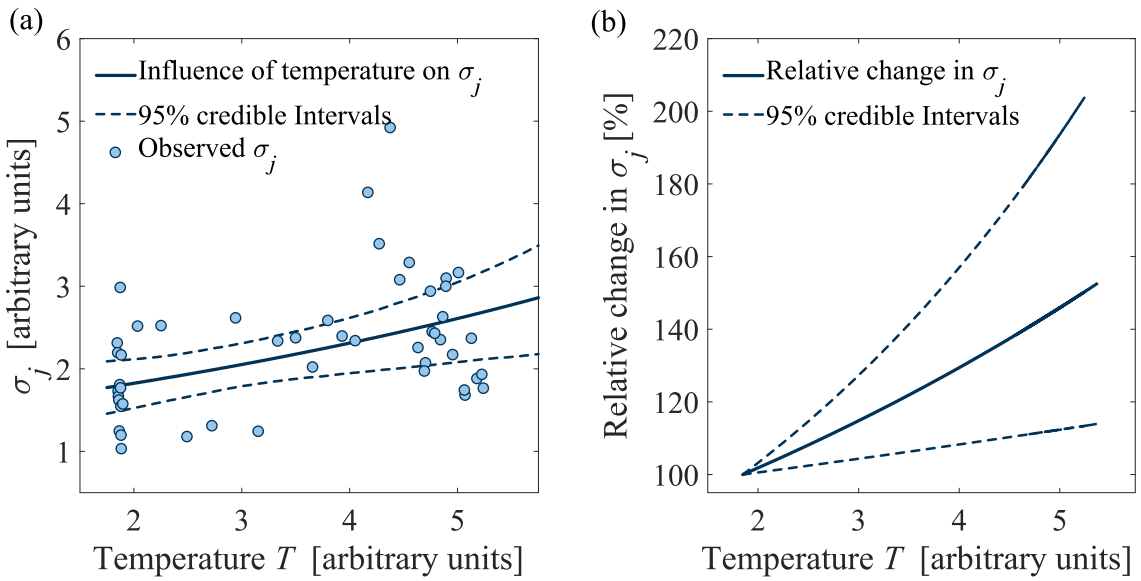

Figure 5.15 (a) Influence of temperature on the standard deviation σ_j of Δx_j. Points are observed σ_j. The 95% credible intervals account for the a posteriori uncertainty in the constant factor β_0 and the weight β_1 of the temperature influence, see Eq. (5.27). (b) Relative change of σ_j with temperature T. The credible intervals only account for the weight β_1. Taken and adapted from [161].

[70] Based on the results, it is possible that an increased temperature causes an increased standard deviation. It could however also be the case that another factor which is correlated with the temperature in Figure 5.13b) actually causes the standard deviation to increase.

Figure 5.15b) shows the relative change of σ_j with increasing temperature T. The standard deviation of position deviations increases in the mean by 46%, when increasing the temperature of the sensor casing from 1.85 to 5.00 [arbitrary units].

Finally, the standard deviation σ_j is predicted. The prediction is performed by sampling τ_φ, β_0, β_1 and $\varphi_{\sigma,j}$ with MCMC and inserting these parameter into Eq. (5.27). The result is displayed in Figure 5.16.

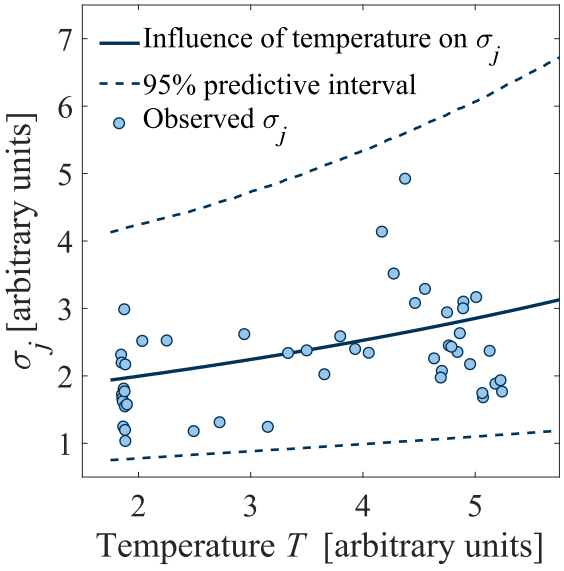

Figure 5.16 Posterior predictive distribution of the standard deviation σ_j. Conditional on the data and the temperature T, σ_j is predicted to lie with 95% probability between the dashed lines.

With 95% probability, the standard deviation is predicted to lie between the dashed lines, conditional on the data in Figure 5.13 and conditional on temperature T. To obtain a prediction not conditional on T, one needs to estimate the exposure $f_T(t)$ towards T. Averaging the prediction in Figure 5.16 over $f_T(t)$ is then a prediction of σ_j accounting for the context variable T and for random effects that are not due to T.

5.3.4 Discussion and Conclusions

The main idea behind the proposed framework is to learn *(sensor) perception reliability* in dependence of context variables. Combining the learned relationship with the probability of exposure towards the context variables then allows to predict the overall *(sensor) perception reliability*.

With this approach, we partly address the concerns of scenario based testing on proving grounds as discussed in Section 3.3: a) the framework provides statistical evidence on *(sensor) perception reliability* over the population of possible scenarios, accounting for context variables **e**, instead of

evaluating a specific test case deterministically as either passed or failed. b) The method (partly) accounts for factors not present in a specific test case with the population of random effects in Eq. (5.20). Hence the method does not require an infinite number of test cases. Both a) and b) combined account for *higher order uncertainties* and allow to efficiently assess *(sensor) perception reliability*.

In the learning framework, the distribution of random effects $f_{\boldsymbol{\varphi}|\boldsymbol{\phi}}(\boldsymbol{\varphi}_j|\boldsymbol{\phi})$ in Eq. (5.20) is informed by factors not included in **E** whose influence is present in the data **x**. The distribution $f_{\boldsymbol{\varphi}|\boldsymbol{\phi}}(\boldsymbol{\varphi}_j|\boldsymbol{\phi})$ however cannot account for the stochastic effect of factors, which are not present in the data **x** *AND* would lead to a different sensor behavior than expected under the model $f_{\boldsymbol{\varphi}|\boldsymbol{\phi}}(\boldsymbol{\varphi}_j|\boldsymbol{\phi})$[71]. For example, one cannot identify the effect of rare shock events (see Section 6.1.4 for a definition of shock events) not present on the proving ground that systematically would lead to perception errors. This is a fact one has to bear in mind if the *perception reliability* analysis relies to a large degree on a scenario based testing.

To apply the learning framework of Section 5.3.1, the context variables have to be measured (and controlled on a proving ground). As already discussed in Section 3.3.2, some context variables such as snowfall are difficult to measure quantitatively or to control on proving grounds. Additionally, relevant context variables **E** have to be known a priori to set up the relationship between **E** and the model parameter in Eq. (5.19). Relevant context variables can for instance be identified with the method of Section 5.1.

A practical problem is however that the interactions between different context variables can – due to the course of dimensionality – only partly be taken into account. The curse of dimensionality refers to an exponential increase in the required data (i.e. test effort) with the number of features (here context variables) in learning relationships such as Eq. (5.19), if interactions between the features are to be considered [115, 189]. In the case study, for example, the object we observed was a wooden wall. Postulating that the effect of temperature on sensor accuracy depends also on the object type (i.e. an interaction between the two context variables temperature and object type) would require to obtain data for the combinations of each object type and each relevant temperature level. The required test effort to account for all possible interactions between context variables hence becomes probably intractably large.

A prerequisite for the prediction of *(sensor) perception reliability* according to Section 5.3.2 is an estimation of the exposure $f_{\mathbf{E}}(\mathbf{e})$ towards context variables. While $f_{\mathbf{E}}(\mathbf{e})$ could in principle be estimated based on additional data and field studies, or through expert knowledge, an accurate assessment of $f_{\mathbf{E}}(\mathbf{e})$ is a challenge in practice. An advantage of the framework is that by altering

[71] This means, the factors not present in the data **x** would lead to a different distribution $f_{\boldsymbol{\varphi}}(\boldsymbol{\varphi}_j|\boldsymbol{\phi})$.

$f_{\mathbf{E}}(\mathbf{e})$ in Eq. (5.23), one is able to account for exposure frequencies in different geographical regions (e.g. a hot and dry region versus a cold and wet region). This allows to evaluate *(sensor) perception reliability* under varying user profiles.

In conclusion, the presented framework allows to preclude an unsatisfactory *(sensor) perception reliability* under the influence of relevant context variables. The results of a case study show that the proposed methodology is for example able to identify and quantify the relationship between temperature and *(sensor) perception reliability* in the state uncertainty domain. The presented framework could therefore contribute to a validation of *(sensor) perception reliability* and improves scenario based testing procedures evaluated with pass /fail criteria. Due to the risk of not accounting for critical factors not present in the tested scenarios on the proving grounds, additional field test are however recommendable for a final validation of *(sensor) perception reliability*.

5.4 Learning Perception Reliability with Field Tests in Real Traffic

This Section is based on our publications in [39, 157] and is partly taken from our publications in [39, 157].

Extensive field tests in real traffic can account for the variability of environmental factors and thus allow to learn and validate *(sensor) perception reliability* under realistic conditions [44, 65, 66]. A challenge is the setting-up of a reference truth, to test representatively, to deal with changes in the perception system and the approval trap (see Section 3.1.4). We address the approval trap in Section 4.2 with methods from system reliability theory by conceptualizing the perception module in terms of its individual sensors (k-out-of-n vote in Figure 4.7). Our contribution in this Section is the presentation of methods to actually learn *sensor perception reliabilities* in field tests.

With the availability of a reference truth to identify perception deficiencies, learning *(sensor) perception reliability* is straightforward. Either the sensor or perception error rate is directly evaluated with a suitable definition of perception errors, or alternatively, the perception reliability metrics according to Section 4.1 are learned from the data. In this Section, we outline with the example of existence uncertainties how to learn *(sensor) perception reliabilities* and dependencies from representative field tests with a reference truth. Representative field tests account for the variability in sensor performance (*higher order uncertainties*) and the resulting estimates of the sensor or perception error rates can be interpreted as global mean error rates (Sections 3.1.4 and 4.1.4). Applying Section 4.3.2 for instance ensures that a test is approximately representative.

In particular, we formulate the problem of learning *perception reliability* of redundant sensors in the context of existence uncertainties in Section 5.4.1. Bayesian inference [114] straightforwardly

enables to learn the underlying reliability metrics, as we outline in Section 5.4.2. Additionally, a statistical model for existence uncertainties in redundant sensors is introduced in Section 5.4.3 under the assumption of statistically independent sensor errors. As is apparent from Figure 4.15, an important aspect for *perception reliability* are potential statistical dependencies among sensor errors. The beta-binomial dependence model introduced in Section 4.2 assumes identical sensor error rates with equi-correlated sensor errors. As a more flexible alternative to the beta-binomial distribution, we propose in Section 5.4.4 to model the dependence among FN or FP sensor errors with a Gaussian copula [115, 270–273]. Finally, we provide a low rank parameterization of the Gaussian copula [115] in Section 5.4.5, with the advantage of having a smaller number of model parameters.

5.4.1 Problem Formulation: Learning (Sensor) Perception Reliability in the Existence Uncertainty Domain

In accordance with Sections 4.1.1 and 4.1.5, we interpret the sensors' object detections binary. The object data of n redundant sensors is analyzed within a certain temporal discretization, e.g. with t_{crit} as in Eq. (4.8). A single sensor is identified with i, for $i = 1, \dots, n$ sensors. A specific discrete point in time is identified with m, for $m = 1,2,3, \dots M$. M is the total number of observed discrete points in time. The output of sensor i at point in time m is described by a binary random variable $D_{i,m}$, with states detection $D_{i,m} = 1$ or no detection $D_{i,m} = 0$. The random variable O_m is introduced, which describes the binary state of the reference truth at point in time m. With $O_m = 1$, an object is present, and with $O_m = 0$ no object is present. $D_{i,m}$ and O_m enable to evaluate the individual sensor output with the confusion matrix of Figure 4.3.

With sensor redundancy, not only the confusion matrix of each individual sensor but also the joint performance of all sensors in the sensor set is relevant, because sensor data fusion combines the output of different sensors [57–59, 110, 118]. We therefore describe the output of the sensor set as a binary random vector $\boldsymbol{D}_m$ with $\boldsymbol{D}_m = [D_{1,m}, \dots, D_{n,m}]$. $\boldsymbol{D}_m$ is termed detection vector, i.e. it is the vector of binary object detections in the different sensors (e.g. referring to a specific area of the FOV). With n sensors, there are 2^n different possible outcomes of $\boldsymbol{D}_m$. The 8 possible outcomes of $\boldsymbol{D}_m$ with $n = 3$ sensors are for example illustrated in Table 5.3. A random variable Y_m is introduced, which identifies a specific combination of $\boldsymbol{D}_m$ and facilitates notation. Y_m takes values between 1,…, 2^n and once defined as in Table 5.3, $Y_m = y_m$ can be written interchangeably for $\boldsymbol{D}_m = [d_{1,m}, \dots, d_{n,m}]$. As an example, with the definitions in Table 5.3, $Y_m = 4$ is identical to $\boldsymbol{D}_m = [0,0,1]$.

Table 5.3 Illustration of the sample space of the detection vector $\boldsymbol{D_m = [D_{1,m}, \dots, D_{n,m}]}$ for $\boldsymbol{n = 3}$ sensors. The variable $\boldsymbol{Y_m}$ identifies a specific detection vector $\boldsymbol{D_m}$.

Variable Y_m	$D_{1,m}$	$D_{2,m}$	$D_{3,m}$
1	0	0	0
2	1	0	0
3	0	1	0
4	0	0	1
5	1	1	0
6	0	1	1
7	1	0	1
8	1	1	1

In the binary setting, the joint performance of the sensors is fully described by the frequencies of the different elements in Table 5.3, once conditional on $O_m = 1$ and once conditional on $O_m = 0$. The assessment of these conditional frequencies is the task of the *sensor perception reliability* analysis. We are interested in global averages of these frequencies, to assess if safety and performance goals are met. This is in contrast to sensor data fusion [57–59, 110, 118], which aims at obtaining instantaneous estimates of these frequencies for an optimal combination of the information provided by different sensors.

We therefore describe the occurrence of $Y_m = y_m$ with its probability in a large number of randomly selected points in time, conditional on O_m. With this description, we implicitly assume exchangeability, i.e. no information is conveyed by the ordering of the different points in time $m = 1,2, \dots, M$. Hence, we drop the index m in the following and the joint performance of the sensors is fully described by the conditional probability mass functions $p_{Y|O}$, i.e. $\Pr(Y = y|O = 1)$ and $\Pr(Y = y|O = 0)$, for any value y. The implications of assuming exchangeability are discussed in Section 4.2.4.

Let $\boldsymbol{\theta}$ denote the model parameters that describe the probability mass functions $p_{Y|O}$. In this Section, we consider the model parameters $\boldsymbol{\theta}$ to be constant and not a function of any context variables (covariates). The probability of an object being present in a specific area of the FOV is described by its a priori probability $\Pr(O = 1) = p_{\text{obj}}$. With these definitions, Figure 5.17 summarizes the model graphically by means of a Bayesian network [116].

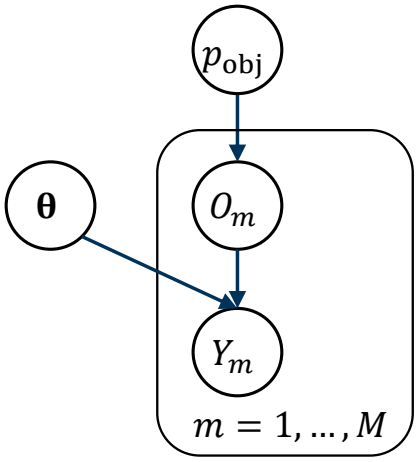

Figure 5.17 Bayesian network representation of the model for learning sensor perception reliability in the existence uncertainty domain. p_{obj} and $\boldsymbol{\theta}$ are the model parameters to be learned, Y_m identifies the detection vector as e.g. defined in Table 5.3 for $n = 3$ and O_m indicates if an object is present. Taken and adapted from [157].

5.4.2 Learning Perception Reliability with a Reference Truth

With a reference truth, one obtains labeled data $\mathbf{x}_{\mathrm{labeled}} = [y_m, o_m]_{m=1}^{M}$, i.e. whether an object is actually present or not, described by o_m, is known for each m. From $\mathbf{x}_{\mathrm{labeled}}$, one can directly derive method of moment or maximum likelihood estimates of the individual sensors' POD and PFA as well as of the frequencies of $Y = y$, conditional on an object being present or not. As an alternative, we utilize Bayesian analysis for the inference of $\boldsymbol{\theta}$ and p_{obj} to account for estimation uncertainties [114, 115]. In Bayesian statistics, a probability expresses a degree of belief that can change as new information is gathered. The posterior of $\boldsymbol{\theta}$ and p_{obj} is readily obtained based on the data. In case of p_{obj} it is:

$$f\left(p_{\mathrm{obj}} \middle| \mathbf{x}_{\mathrm{labeled}}\right) \propto f\left(p_{\mathrm{obj}}\right) \cdot \prod_{m=1}^{M} {p_{\mathrm{obj}}}^{\mathbb{I}(o_m=1)} \cdot \left(1 - p_{\mathrm{obj}}\right)^{\mathbb{I}(o_m=0)} \tag{5.29}$$

$f\left(p_{\mathrm{obj}} \middle| \mathbf{x}_{\mathrm{labeled}}\right)$ is the posterior probability density function (PDF) of p_{obj} given the vector of observed data $\mathbf{x}_{\mathrm{labeled}}$, $f\left(p_{\mathrm{obj}}\right)$ is the prior PDF and $\prod_{m=1}^{M} {p_{\mathrm{obj}}}^{\mathbb{I}(o_m=1)} \cdot \left(1 - p_{\mathrm{obj}}\right)^{\mathbb{I}(o_m=0)}$ is the likelihood describing the data. $\mathbb{I}(o_m = 1)$ is the indicator function, which equals one if $o_m = 1$ is true and zero otherwise. By choosing a beta distributed prior $f\left(p_{\mathrm{obj}}\right)$, the posterior is also beta distributed, hence this is a convenient choice (the beta distribution is the conjugate prior) [114].

The posterior of $\boldsymbol{\theta}$ is:

$$f(\boldsymbol{\theta} | \mathbf{x}_{\mathrm{labeled}}) \propto f(\boldsymbol{\theta}) \cdot \prod_{m=1}^{M} p_{Y|\boldsymbol{\theta},O}(y_m | \boldsymbol{\theta}, o_m) \tag{5.30}$$

$f(\boldsymbol{\theta})$ is the prior PDF of $\boldsymbol{\theta}$ and $\prod_{m=1}^{M} p_{Y|\boldsymbol{\theta},O}(y_m | \boldsymbol{\theta}, o_m)$ is the likelihood. If $p_{Y|\boldsymbol{\theta},O}$ is described by a multinomial distribution, the Dirichlet PDF is the conjugate prior, which leads to a closed form

solution [115]. Alternatively, the posterior can be estimated based on Markov Chain Monte Carlo (MCMC) methods [114, 115, 266, 267].

When $p_{Y|\boldsymbol{\theta},O}$ is described with the multinomial distribution, each of the conditional distributions $p_{Y|\boldsymbol{\theta},O=1}$ and $p_{Y|\boldsymbol{\theta},O=0}$ has $2^n - 1$ parameters, resulting in a total of $2^{n+1} - 2$ free parameters $\boldsymbol{\theta}$ that need to be learned (2^n parameters for the different values of y, where one parameter is redundant because of the constraint $\sum_{y=1}^{y=2^n} \Pr(Y = y|O = o) = 1$). The parameters $\boldsymbol{\theta}$ are the relative frequencies of the different values of Y. The following Sections specify alternatives to the multinomial distribution for $p_{Y|\boldsymbol{\theta},O}$ to reduce the number of model parameters.

5.4.3 Model for Statistically Independent Sensor Errors

In accordance with Section 4.1.1, the *sensor perception reliability* of an individual sensor $i = 1, \dots, n$ in the existence uncertainty domain is described with POD_i and PFA_i. When sensor errors are statistically independent, the probability $p_{Y|\boldsymbol{\theta},O}(y|\boldsymbol{\theta}, o)$ is fully determined by the parameters $\boldsymbol{\theta}_{\mathrm{indep}} = [\mathrm{POD}_1, \dots, \mathrm{POD}_n, \mathrm{PFA}_1, \dots, \mathrm{PFA}_n]$. The probability $p_{Y|\boldsymbol{\theta}_{\mathrm{indep}},O}\left(y\middle|\boldsymbol{\theta}_{\mathrm{indep}}, O = 1\right)$ given an object is present is:

$$p_{Y|\boldsymbol{\theta}_{\mathrm{indep}},O}\left(y\middle|\boldsymbol{\theta}_{\mathrm{indep}}, O = 1\right) = \prod_{i=1}^{n} (\mathrm{POD}_i)^{\mathbb{I}(d_i=1)} \cdot (1 - \mathrm{POD}_i)^{\mathbb{I}(d_i=0)} \tag{5.31}$$

The values of $d_i,\ i = 1, \dots, n$ are those corresponding to y, in analogy to Table 5.3. The probability $p_{Y|\boldsymbol{\theta}_{\mathrm{indep}},O}\left(y\middle|\boldsymbol{\theta}_{\mathrm{indep}}, O = 0\right)$ given no object is present is:

$$p_{Y|\boldsymbol{\theta}_{\mathrm{indep}},O}\left(y\middle|\boldsymbol{\theta}_{\mathrm{indep}}, O = 0\right) = \prod_{i=1}^{n} (\mathrm{PFA}_i)^{\mathbb{I}(d_i=1)} \cdot (1 - \mathrm{PFA}_i)^{\mathbb{I}(d_i=0)} \tag{5.32}$$

With a reference truth, information on o_m is available, hence one can directly insert Eqs. (5.31)-(5.32) for each m into Eq. (5.30) and learn $\boldsymbol{\theta}_{\mathrm{indep}}$. The advantage of this model over the use of the multinomial distribution is that the number of parameters is only $2n$ as opposed to $2^{n+1} - 2$ in the case of a multinomial distribution.

5.4.4 Gaussian Copula for Statistically Dependent Sensor Errors

The model of Section 5.4.3 does not account for potential statistical dependencies in FN or FP errors among different sensors. The statistical error dependence is fully accounted for by the probabilities of $\Pr(y|O = o)$. Describing $\Pr(y|O = o)$ with a multinomial distribution leads to $2^n - 1$ free parameters that need to be learned once for $O = 1$ and once for $O = 0$. To reduce the number of parameters relative to the multinomial distribution, we introduce a model of statistical dependence for *sensor perception reliabilities* in the existence uncertainty domain based on the Gaussian copula, which is the most common copula model [115, 270–273].

The Gaussian copula for correlated binary data is also known as a multivariate probit in the context of regression analysis [115, 273]. The Gaussian copula model is based on transforming the marginal (Bernoulli) random variables D_i, describing the individual sensor detections, to random variables U_i in standard normal space. One then imposes that the transformed D_i of the different sensors i in standard normal space jointly have the multi-normal distribution, whose correlation matrix describes the statistical dependence among the sensors.

Transformation of D_i to Standard Normal Space

The transformation to standard normal space enables a straightforward description of the dependence among the random variables D_i with a manageable number of free parameters. The random variable D_i conditional on an object being present, describing TP and FN sensor detections, is transformed into standard normal space by the probability conserving transformation:

$$\mathrm{POD}_i = \Pr(D_i = 1|O = 1) = \Pr(U_{\mathrm{FN},i} \leq \Phi^{-1}(\mathrm{POD}_i)|O = 1) \tag{5.33}$$

where $U_{\mathrm{FN,i}}$ is a standard normal random variable, Φ^{-1} the inverse standard normal cumulative distribution function (CDF) and $u_{\mathrm{POD},i} = \Phi^{-1}(\mathrm{POD}_i)$. Consequently, it is $\Pr(D_i = 0|O = 1) = \Pr(U_{\mathrm{FN},i} > \Phi^{-1}(\mathrm{POD}_i)|O = 1)$. The inverse transformation is illustrated in Figure 5.18.

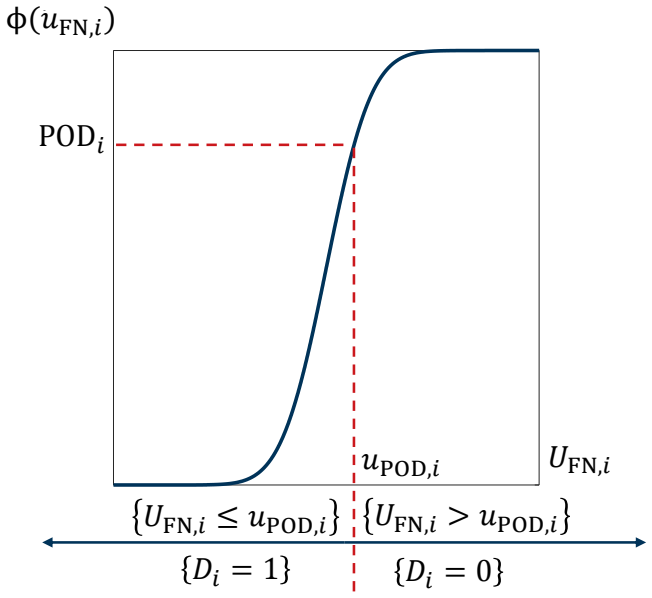

Figure 5.18 Inverse transformation of TP and FN probabilities $\Pr(D_i = 1|O = 1)$ and $\Pr(D_i = 0|O = 1)$ to a standard normal random variable $U_{\mathrm{FN},i}$. The solid blue line is the standard normal CDF. Taken and adapted from [157].

The transformation for FP and TN sensor outcomes is equivalent, with $u_{\mathrm{PFA},i} = \Phi^{-1}(\mathrm{PFA}_i)$, resulting in a standard normal random variable $U_{\mathrm{FP},i}$.

Gaussian Copula

The statistical dependence among the sensors i and j is represented by the correlation coefficient $\rho_{\mathrm{U}_{\mathrm{FN},i,j}}$ among the random variables $U_{\mathrm{FN},i}$ and $U_{\mathrm{FN},j}$, and by the correlation coefficient $\rho_{\mathrm{U}_{\mathrm{FP},i,j}}$ among the random variables $U_{\mathrm{FP},i}$ and $U_{\mathrm{FP},j}$. As an example, the joint probability of a FN sensor error in both sensors i and j under this model is:

$$\Pr\left(D_i = 0, D_j = 0 | O = 1\right) = \Phi_2\left(\begin{bmatrix} -u_{\mathrm{POD},i} \\ -u_{\mathrm{POD},j} \end{bmatrix}, \begin{bmatrix} 1 & \rho_{\mathrm{U}_{\mathrm{FN},i,j}} \\ \rho_{\mathrm{U}_{\mathrm{FN},i,j}} & 1 \end{bmatrix}\right) \tag{5.34}$$

where Φ_2 is the bivariate standard normal CDF with zero mean vector and covariance matrix defined by $\rho_{\mathrm{U}_{\mathrm{FN},i,j}}$. Eq. (5.34) is derived from $\Pr\left(D_i = 0, D_j = 0 | O = 1\right) = \Pr(U_{\mathrm{FN},i} > u_{\mathrm{POD},i}, U_{\mathrm{FN},j} > u_{\mathrm{POD},j} | O = 1)$.

Note that $\rho_{\mathrm{U}_{\mathrm{FN},i,j}}$ and $\rho_{\mathrm{U}_{\mathrm{FP},i,j}}$ are not identical to $\rho_{\mathrm{FN},i,j}$ and $\rho_{\mathrm{FP},i,j}$ (as in Eq. (4.17)), respectively. The relationship between e.g. $\rho_{\mathrm{FP},i,j}$ and $\rho_{\mathrm{U}_{\mathrm{FP},i,\mathrm{j}}}$ is [274]:

$$\rho_{\mathrm{FP},i,j} = \int_{-\infty}^{\infty}\int_{-\infty}^{\infty} \frac{\mathbb{I}\left(u_{\mathrm{FP},i} \le u_{\mathrm{PFA},i}\right) - \mathrm{PFA}_i}{\sqrt{\mathrm{PFA}_i \cdot (1 - \mathrm{PFA}_i)}} \cdot \frac{\mathbb{I}\left(u_{\mathrm{FP},j} \le u_{\mathrm{PFA},j}\right) - \mathrm{PFA}_j}{\sqrt{\mathrm{PFA}_j \cdot (1 - \mathrm{PFA}_j)}} \cdot \varphi_2\left(\begin{bmatrix} u_{\mathrm{FP},i} \\ u_{\mathrm{FP},j} \end{bmatrix}, \begin{bmatrix} 1 & \rho_{\mathrm{U}_{\mathrm{FP},i,j}} \\ \rho_{\mathrm{U}_{\mathrm{FP},i,j}} & 1 \end{bmatrix}\right) du_{\mathrm{FP},i} du_{\mathrm{FP},j} \tag{5.35}$$

where φ_2 is the bivariate standard normal PDF with zero mean vector and covariance matrix defined by $\rho_{\mathrm{U}_{\mathrm{FP},i,j}}$.

In the general case, more than two sensors are present. With the Gaussian copula, the dependence among n sensors is described by the joint distribution of $\mathbf{U}$, which is multivariate normal. Its PDF is defined as:

$$f_{\mathbf{U}|O}(\mathbf{U} = \mathbf{u} | O = o) = \begin{cases} \varphi_{\mathrm{n}}\left(\mathbf{u}, \mathbf{R}_{\mathbf{U},\mathrm{FN}}\right), & o = 1 \\ \varphi_{\mathrm{n}}\left(\mathbf{u}, \mathbf{R}_{\mathbf{U},\mathrm{FP}}\right), & o = 0 \end{cases} \tag{5.36}$$

where $\varphi_{\mathrm{n}}(\mathbf{u}, \mathbf{R})$ is the n-dimensional multivariate correlated standard normal PDF with argument $\mathbf{u} = [u_1, \ldots, u_n]$ and covariance matrix $\mathbf{R}$. If $o = 1$, the correlation matrix in standard normal space is $\mathbf{R}_{\mathbf{U},\mathrm{FN}}$, whose elements are $\rho_{\mathrm{U}_{\mathrm{FN},i,j}}$; if $o = 0$, the correlation matrix is $\mathbf{R}_{\mathbf{U},\mathrm{FP}}$, whose elements are $\rho_{\mathrm{U}_{\mathrm{FP},i,j}}$. The model parameters are $\boldsymbol{\theta}_{\mathrm{dep}} = [\mathrm{POD}_1, \ldots, \mathrm{POD}_n, \mathrm{PFA}_1, \ldots, \mathrm{PFA}_n, \mathbf{R}_{\mathbf{U},\mathrm{FN}}, \mathbf{R}_{\mathbf{U},\mathrm{FP}}]$.

The conditional probability $p_{Y|\boldsymbol{\theta}_{\text{dep}},O}\left(y\middle|\boldsymbol{\theta}_{\text{dep}},O=o\right)$ is determined based on Eq. (5.36). We demonstrate this here for $O=1$. Each Y corresponds to a combination of the detection vector $\mathbf{D}$, i.e. $p_{Y|\boldsymbol{\theta}_{\text{dep}},O}\left(y\middle|\boldsymbol{\theta}_{\text{dep}},O=o\right)=p_{\mathbf{D}|\boldsymbol{\theta}_{\text{dep}},O}(\mathbf{d}|\boldsymbol{\theta}_{\text{dep}},O=o)$, whereby the elements of $\mathbf{d}$ follow Table 5.3 (or its alternative for a different number of sensors). It is reminded that $\{D_i=1\}$ is equivalent to $\left\{U_{\text{FN},i}\le u_{\text{POD},i}\right\}$ and $\{D_i=0\}$ is equivalent to $\left\{U_{\text{FN},i}>u_{\text{POD},i}\right\}$. To compute the joint probability $p_{\mathbf{D}|\boldsymbol{\theta}_{\text{dep}},O}(\mathbf{d}|\boldsymbol{\theta}_{\text{dep}},O=o)$, one therefore needs to integrate Eq. (5.36) for each combination of $\mathbf{d}=[d_1,\dots,d_n]$ over a n-dimensional hypercube defined by $\left\{U_{\text{FN},i}\le u_{\text{POD},i}\right\}$ if $D_i=1$ or $\left\{U_{\text{FN},i}>u_{\text{POD},i}\right\}$ if $D_i=0$ for all $i=1,\dots,n$. Figure 5.19 graphically represents the integration of Eq. (5.36) over $U_{\text{FN},1}>u_{\text{POD},1}$ and $U_{\text{FN},2}>u_{\text{POD},2}$ by the probability density in the red shaded area with the example of $n=2$. Note that the probability density in the red shaded area in Figure 5.19 is schematically identical to Eq. (5.34).

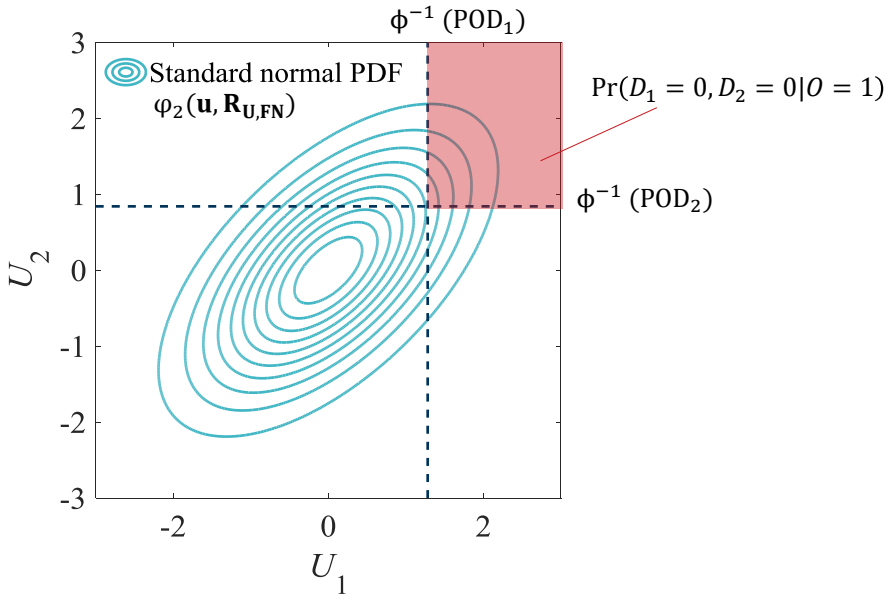

Figure 5.19 Graphical illustration of Gaussian copula dependence model for correlated sensor errors. Contour lines represent the bivariate standard normal PDF $\boldsymbol{\varphi_2}$. The probability density in the red shaded area is the probability of $\mathbf{Pr}(\boldsymbol{D_1=0,D_2=0|O=1})$. Likewise, $\mathbf{Pr}(\boldsymbol{D_1=d_1,D_2=d_2|O=1})$ are defined analogously for any combination of $\boldsymbol{d_1}$ and $\boldsymbol{d_2}$.

In the general case, the result of integrating Eq. (5.36) over the different n-dimensional hypercubes is:

$$p_{Y|\boldsymbol{\theta}_{\text{dep}},O}\left(y\middle|\boldsymbol{\theta}_{\text{dep}},O=1\right)=p_{\mathbf{D}|\boldsymbol{\theta}_{\text{dep}},O}\left(\mathbf{d}\middle|\boldsymbol{\theta}_{\text{dep}},O=1\right)=$$
$$=\Phi_{\text{n}}\left(\begin{bmatrix}s(d_1)\cdot\phi^{-1}(\text{POD}_1)\\ \vdots\\ s(d_n)\cdot\phi^{-1}(\text{POD}_n)\end{bmatrix},\mathbf{R}_{\mathbf{U,FN},+}(\mathbf{d})\right) \tag{5.37}$$

with Φ_{n} the n dimensional multivariate correlated standard normal CDF and

$$s(d_i) = \begin{cases} +1, & d_i = 1 \\ -1, & d_i = 0 \end{cases} \tag{5.38}$$

regulates over which part of the hyperspace to integrate. $\mathbf{R}_{\mathbf{U},\mathrm{FN},+}(\mathbf{d})$ is an adapted version of $\mathbf{R}_{\mathbf{U},\mathrm{FN}}$, whose elements have different signs. The elements $\rho_{\mathrm{U}_{\mathrm{FN},+,i,j}}$ are defined by:

$$\rho_{\mathrm{U}_{\mathrm{FN},+,i,j}}(d_i, d_j) = \begin{cases} 1, & i = j \\ s(d_i) \cdot s(d_j) \cdot \rho_{\mathrm{U}_{\mathrm{FN},i,j}}, & i \neq j \end{cases} \tag{5.39}$$

In analogy, the probability of sensor outcomes given $O = 0$ is:

$$\begin{aligned} & p_{Y|\boldsymbol{\theta}_{\mathrm{dep}},O}\left(y \middle| \boldsymbol{\theta}_{\mathrm{dep}}, O = 0\right) = p_{\mathbf{D}|\boldsymbol{\theta}_{\mathrm{dep}},O}\left(\mathbf{d} \middle| \boldsymbol{\theta}_{\mathrm{dep}}, O = 0\right) = \\ & = \Phi_{\mathrm{n}}\left(\begin{bmatrix} s(d_1) \cdot \Phi^{-1}(\mathrm{PFA}_1) \\ \vdots \\ s(d_n) \cdot \Phi^{-1}(\mathrm{PFA}_n) \end{bmatrix}, \mathbf{R}_{\mathbf{U},\mathrm{FP},+}(\mathbf{d})\right) \end{aligned} \tag{5.40}$$

The correlation matrices each have $n \cdot (n-1)/2$ correlation coefficients and define the pairwise sensor dependence in FP and FN errors. Therefore, there are a total of $2n + \ n \cdot (n-1) = n^2 + n$ free parameters with this model. For larger number of sensors n, this is a significantly smaller number than the $2^{n+1} - 2$ free parameters of the multinomial distributions. E.g. for 5 sensors, the number of parameters are 30 instead of 62.

Eqs. (5.37)-(5.40) are inserted into Eq. (5.30) to set up the posterior distribution of the model parameters $\boldsymbol{\theta}_{\mathrm{dep}}$ and p_{obj} with a reference truth. If the off-diagonal elements in $\mathbf{R}_{\mathbf{U},\mathrm{FN}}$ and $\mathbf{R}_{\mathbf{U},\mathrm{FP}}$ are set to zero, the model is identical to the one of Section 5.4.3.

The resulting posterior in Eq. (5.30) can for instance be evaluated with MCMC methods [114, 115, 266, 267]. This is however computationally costly and numerically instable for small probabilities because of the involved evaluation of the multinormal CDF. Furthermore, the still significant number of free parameters to estimate reduces the convergence rate of the MCMC algorithms. In the next Section, we therefore report an alternative formulation of the Gaussian copula for correlated sensor errors, based on a low rank approximation of the correlation matrix. This alternative formulation is identical to Eqs. (5.37)-(5.40) for a special class of correlation matrices, but is computationally cheaper and MCMC based inference is easier to converge.

5.4.5 Gaussian Copula with Low Rank Correlation Matrices for Statistically Dependent Sensor Errors

In this Section we describe a low rank parameterization of the correlated multivariate normal distribution [115], applied to the Gaussian copula. This low rank parameterization is a compromise between the flexibility of capturing generic dependence structures, computational costs and difficulties in MCMC convergence. The presentation in this Section is based on [115, 242, 275, 276].

We prescribe that the correlation matrix $\mathbf{R}_{\mathbf{U},\mathrm{FN}}$ is of the Dunnet-Sobel class, which are rank one correlation matrices, whose entries are constructed as follows:

$$\rho_{\mathrm{U}_{\mathrm{FN},i,j}} = \lambda_{\mathrm{U}_{\mathrm{FN},i}} \cdot \lambda_{\mathrm{U}_{\mathrm{FN},j}} \tag{5.41}$$

wherein the $\lambda_{\mathrm{U}_{\mathrm{FN},i}}$ are the model parameters to be learnt. The construction of $\mathbf{R}_{\mathbf{U},\mathrm{FP}}$ is equivalent.

The use of this rank one correlation matrix reduces the number of free parameters to learn from $n^2 + n$ to $4n$. The second advantage of model is that, instead of the n-dimensional integration required to evaluate Eqs. (5.37)-(5.40), it allows to compute $p_{\mathbf{D}|\boldsymbol{\theta}_{\mathrm{dep}},O}\left(\mathbf{d}\middle|\boldsymbol{\theta}_{\mathrm{dep}}, o\right)$ with a one dimensional integration:

$$p_{\mathbf{D}|\boldsymbol{\theta}_{\mathrm{dep,DS}},O}\left(\mathbf{d}\middle|\boldsymbol{\theta}_{\mathrm{dep,DS}}, o\right) = \int_{-\infty}^{\infty} \varphi(u_{\mathrm{c}}) \cdot p_{\mathbf{D}|\mathrm{U_c},\boldsymbol{\theta}_{\mathrm{dep,DS}},O}\left(\mathbf{d}\middle|u_{\mathrm{c}}, \boldsymbol{\theta}_{\mathrm{dep,DS}}, o\right) du_{\mathrm{c}} \tag{5.42}$$

where U_{c} is a standard normal distributed auxiliary random variable and $\varphi(u_{\mathrm{c}})$ its PDF. Given $U_{\mathrm{c}} = u_{\mathrm{c}}$, the detection events in the different sensors are conditionally independent. Conditional on $U_{\mathrm{c}} = u_{\mathrm{c}}$ and $O = 1$, the probability $p_{\mathbf{D}|\mathrm{U_c},\boldsymbol{\theta}_{\mathrm{dep,DS}},O}\left(\mathbf{d}\middle|u_{\mathrm{c}}, \boldsymbol{\theta}_{\mathrm{dep,DS}}, o\right)$ of the detection vector is:

$$p_{\mathbf{D}|U_{\mathrm{c}},\boldsymbol{\theta}_{\mathrm{dep,DS}},O}\left(\mathbf{d}\middle|u_{\mathrm{c}}, \boldsymbol{\theta}_{\mathrm{dep,DS}}, O = 1\right) = \prod_{i=1}^{n} \left[d_i \cdot \phi\left(\frac{\phi^{-1}(\mathrm{POD}_i) - u_{\mathrm{c}} \cdot \lambda_{\mathrm{U}_{\mathrm{FN},i}}}{\sqrt{1 - \lambda_{\mathrm{U}_{\mathrm{FN},i}}{}^2}} \right) + (1 - d_i) \cdot \left[1 - \phi\left(\frac{\phi^{-1}(\mathrm{POD}_i) - u_{\mathrm{c}} \cdot \lambda_{\mathrm{U}_{\mathrm{FN},i}}}{\sqrt{1 - \lambda_{\mathrm{U}_{\mathrm{FN},i}}{}^2}} \right) \right] \right] \tag{5.43}$$

Where Φ is the standard normal CDF.

Equivalently, given $U_{\mathrm{c}} = u_{\mathrm{c}}$ and $O = 0$, the probability of the detection vector $p_{\mathbf{D}|U_{\mathrm{c}},\boldsymbol{\theta}_{\mathrm{dep,DS}},O}(\mathbf{d}|u_{\mathrm{c}},\boldsymbol{\theta}_{\mathrm{dep,DS}},o)$ writes

$$p_{\mathbf{D}|U_{\mathrm{c}},\boldsymbol{\theta}_{\mathrm{dep,DS}},\mathrm{O}}(\mathbf{d}|u_{\mathrm{c}},\boldsymbol{\theta}_{\mathrm{dep,DS}},O=0) = \prod_{i=1}^{n}\left[d_i\cdot\phi\left(\frac{\phi^{-1}(\mathrm{PFA}_i)-u_{\mathrm{c}}\cdot\lambda_{\mathrm{U_{FP},}i}}{\sqrt{1-\lambda_{\mathrm{U_{FP},}i}{}^2}}\right)+ (1-d_i)\cdot\left[1-\phi\left(\frac{\phi^{-1}(\mathrm{PFA}_i)-u_{\mathrm{c}}\cdot\lambda_{\mathrm{U_{FP},}i}}{\sqrt{1-\lambda_{\mathrm{U_{FP},}i}{}^2}}\right)\right]\right] \tag{5.44}$$

Inserting Eqs. (5.42)-(5.44) into Eq. (5.30) results in the posterior distribution of $\boldsymbol{\theta}_{\mathrm{dep,DS}}$ and p_{obj}. The Dunnet-Sobel class correlation matrix, as defined by (5.41) is not as flexible as a generic correlation matrix.

5.4.6 Discussion

Due to the limited realism of other test methods, field tests are the most important method for a final validation of *(sensor) perception reliability*. When a reference truth is available, estimation of *(sensor) perception reliability* and sensor error rates is straightforward.

In case of existence uncertainties, perception reliabilities of redundant sensors are for instance estimated with Eq. (5.30). In the binary case of existence uncertainties, frequencies of different combinations of sensor detections **d** (e.g. the 8 combinations in Table 5.3**)** conditional on an object being present or not fully describe *sensor perception reliability* and dependencies. From these frequencies, the sensors' POD_i and PFA_i are readily derived, which are related to the FP and FN sensor error rates (Section 4.1.1).

Our contribution in this Section is the introduction of dependence models to learn FN and FP sensor error rates. The overall goal of a *perception reliability* analysis is however not to demonstrate acceptable FP and FN sensor error rates but to demonstrate an acceptable rate of safety-critical perception errors $\lambda_{\mathrm{per}} \leq \lambda_{\mathrm{TLS_{per}}}$. The required test effort to validate *perception reliability* based on the individual sensors was initially estimated in Section 4.3. Once *sensor perception reliabilities* and dependencies have been learned with the models and methods presented in this Section, the perception module's actual *perception reliability*, quantified by λ_{per}, can be estimated either with the k-out-of-n model (Section 4.2), or another possibly more detailed and realistic representation of the sensor data fusion and object detection decision process.

For instance, a detailed statistical simulation framework that incorporates the actual software code and timing behavior of sensor data fusion could be employed, see also Section 5.2. Such a statistical simulation framework enables to simultaneously account for *perception reliabilities* in

all uncertainty domains (Section 4.1), which need to be learned in analogy to this Section. Such an approach would allow to estimate *perception reliability* on the level of the fused environment model from learned *sensor perception reliabilities*, and hence, to validate $\lambda_{\text{per}} \leq \lambda_{\text{TLS}_{\text{per}}}$.

Learning *sensor perception reliabilities* with Eq. (5.30) is flexible w.r.t. the statistical model of sensor errors. For example, directly employing the frequencies of the different combinations of sensor detections $\mathbf{d}$ results in a multinomial model. Due to the large number of parameters in the multinomial model that need to be learned, we proposed the Gaussian copula dependence model as an alternative. The Gaussian copula does neither assume identical sensor error rates nor equi-correlated sensor errors, which are the assumptions of the beta-binomial model. Hence, compared with the beta-binomial model introduced in Section 4.2, used to derive initial *sensor perception reliability* requirements, the Gaussian copula is more flexible in actually learning TP and FP sensor error rates from data. We additionally proposed a low rank approximation to the Gaussian copula to reduce the computational costs in evaluating the posterior distribution of the model parameters of interest. In principle, one could however also use any other suitable dependence model.

It is pointed out that the assumption of exchangeability also applies to Eq. (5.30). For a discussion on exchangeability we refer to Section 4.2.4. Another assumption of this Section was that a reference truth is available, i.e. labeled data $\mathbf{x}_{\text{labeled}}$ enables the identification of perception deficiencies. Setting up a reference truth in real driving environments can be demanding and the post processing of the data time consuming. In case suitable data with a reference truth is available, Bayesian parameter inference with the proposed models is straightforward, e.g. with MCMC sampling [114, 115, 266, 267].

In the future, the proposed models must be evaluated with data to further examine dependence structures of FP and FN errors among different sensor technologies (e.g. radar, camera and lidar sensors). As in any data analysis, the data then enables to improve and adjust the proposed models. Because of its high flexibility in modeling pairwise dependencies, and because it accounts for varying error rates in different sensors, the Gaussian copula can be used as an initial dependence model for an analysis of *sensor perception reliabilities* in field tests. The challenge and substantial effort of setting up a reference truth in field tests however leads to the question: can one learn *sensor perception reliabilities* without a reference truth?

6 Assessing Perception Reliability by Exploiting Redundancy

This Section is based on our publications in [156, 157] and partly taken from our publications in [156, 157].

Chapter 5 outlined a variety of methods to assess *(sensor) perception reliability*. Field tests allow testing under realistic conditions. Depending on the number of redundant sensors and the statistical dependence among sensor errors, the effort in field tests to demonstrate $\lambda_{\mathrm{per}} \leq \lambda_{\mathrm{TLS}_{\mathrm{per}}}$ under the k-out-of-n representation of sensor data fusion is feasible (Section 4.3). In these tests, a reference truth needs to be derived either with suitable reference sensors in combination with human data labeling, from online sensor data fusion [57, 58, 110, 118], or from automatic offline labeling algorithms [158]. Employing reference sensors requires considerable efforts and online sensor data fusion as well as offline automatic data labeling is imperfect.

Furthermore, statistical dependencies among sensor errors are unknown prior to a *sensor perception reliability* assessment, and thus have to be learned as well. The stronger the statistical error dependence, the stricter are the reliability requirements on individual sensors to comply with $\lambda_{\mathrm{per}} \leq \lambda_{\mathrm{TLS}_{\mathrm{per}}}$ (see Figure 4.12). Consequently, the stronger the statistical error dependence among sensors, the larger test effort is required to be able to demonstrate $\lambda_{\mathrm{per}} \leq \lambda_{\mathrm{TLS}_{\mathrm{per}}}$ (see Figure 4.15).

An idea to overcome the limitations of traditional empirical tests is to run automated driving functionalities in the background of a fleet of end-user vehicles without executing the output, while human drivers control the vehicles [68, 201, 277–279]. Discrepancies between human driving and the intended automated driving serve as a pseudo reference for the safety validation of automated driving functionalities jointly with their perception. These approaches are sometimes termed Shadow Mode [68], Trojan horse [40, 279] or open loop tests [277]. One can combine empirical testing and virtual simulations in these approaches [277]. In the following, we refer to these ideas as Shadow Mode. In addition to human driving, other types of information such as a priori knowledge of static obstacles could be used to assess *sensor perception reliability* through fleet learning [280].

Implementing a Shadow Mode in a fleet allows to cover a large number of situations with little effort. A challenge is the prediction of how a situation would have developed, if the automated driving functionality had been in control of the vehicle. Furthermore, some sensor errors might be difficult to identify solely based on a comparison of human and intended automated driving [201]. A limitation of the Shadow Mode is that if the output of the automated driving functionality was actually executed, a different sensing behavior would be observed in a specific situation due to an

altered vehicle dynamics. For example, an emergency brake would alter the sensor's vertical field of view, which in turn could alter the subsequent automated driving function's behavior.

Motivated by the challenges, we investigate theoretically in Section 6.1, if in principle it is possible to learn *sensor perception reliabilities* without a reference truth, by solely comparing the output of redundant sensors. In light of the opportunities of Shadow Mode on the one hand, and the difficulties of jointly evaluating the automated driving functionality with its perception by means of the Shadow Mode on the other hand, we develop a testing framework in Section 6.1, which enables to learn sensor error rates as well as sensor error dependencies without a reference truth. The resulting framework could be implemented as a Shadow Mode for perception. We demonstrate on a theoretical basis that the framework correctly determines *sensor perception reliabilities* if an adequate statistical model for sensor errors and dependencies is employed.

Based on these findings, we extend the learning framework in Section 6.2 to address the main simplifications made in Section 6.1. The proposed framework could facilitate *sensor perception reliability* demonstrations because it enables the learning of *sensor perception reliability* from a fleet of vehicles equipped with standard series sensors. The idea is to quasi automatically generate big data online in end-user vehicles, transfer this data to a backend and learn *sensor perception reliabilities* offline from the aggregated data of the whole fleet in the backend.

6.1 Does One Need a Reference Truth?

In classical reliability assessments and testing [150, 238, 239, 281–288], failure events are well defined and are directly observable in tests. When assessing *sensor perception reliability*, this only holds if a reference truth (ground truth) and a suitable sensor error definition is known. Generating the reference truth is however cumbersome and a comprehensive definition of perception errors difficult (Section 3.1.4). The aim of this Section is to investigate on a theoretical basis, if it is possible to demonstrate *sensor perception reliability* without a reference truth, by solely comparing the output of multiple redundant sensors.

To this end, we simplify the problem of inferring *sensor perception reliabilities* in Section 6.1.1. We formulate likelihood functions in Section 6.1.2 that describe the sensors' observations without reference truth, by comparing the outputs of multiple sensors. We combine the learnable parameters of these likelihood functions in Section 6.1.3 with a k-out-of-n system representation to enable a demonstration of *perception reliability*. Due to the importance of the statistical dependence among sensor errors (Section 3.1.4), we additionally discuss in Section 6.1.4 the challenges associated with selecting an adequate statistical model when learning without reference truth. Synthetic case studies in Section 6.1.5 demonstrate that it is in principle possible to correctly

learn *sensor perception reliabilities* without reference truth by exploiting sensor redundancy. The redundancy provides a statistical reference truth, but without the need to manually generate and verify it.

6.1.1 Simplified Problem Formulation

We consider n redundant environment perceiving sensors. The sensors perform the same type of task, such as the detection of relevant objects. The data of the sensors is analyzed in discretized time e.g. with temporal discretization t_{crit} as in Eqs. (4.7)-(4.8). A specific discrete point in time is identified with $m = 1,2,3, \dots M$. M is the total number of observed discrete points in time. We focus here on the situation in which all n sensors are identical.

It is assumed that a definition of perception errors exists, which – if perfect knowledge of the reference truth is available – would allow to classify the output of a sensor $i = 1, \dots, n$ at a specific point in time either as correct ($x_i = 1$) or erroneous ($x_i = 0$). For example, FP and FN errors in a specific area of the FOV are defined as perception errors. Here, we purposely do not distinguish between different types of errors (e.g. FP and FN errors) to enhance the clarity of the presentation.

It follows that the error occurrence in one sensor is described with a binary random variable X_i. $K = \sum_{i=1}^{n}(1 - X_i)$ is the total number of sensors providing deficient information (i.e. making errors) at a given discrete point in time. Due to the potential statistical dependence among sensor errors, $\Pr(X_i = 0 | X_j = 0)$ is not equal to $\Pr(X_i = 0)$. Additionally, due to the variable perception performance (see Section 3.1.4), $\Pr(X_i = 0)$ is not constant but variable itself. We account for both aspects with the beta-binomial model introduced in Section 4.2.2.

The parameters of the model are the average probability p_{av} of a sensor error ($X_i = 0$) in a large number of randomly selected points in time and the sensor error correlation coefficient ρ. The correlation coefficient is here identical among all pairs of sensors (equi-correlation), because we assume all n sensors to be identical. The beta-binomial model implies equi-correlation, which justifies its selection as the statistical model. Inserting p_{av} and ρ into Eqs. (4.26)-(4.27) defines the beta-binomial model, without distinction in FP and FN errors. The assumption of exchangeability discussed in Section 4.2.4 is here applicable as well.

Depending on the temporal discretization t_{crit} [h], a given p_{av} is related to the sensor error rate $\lambda_{\mathrm{sensor}}$ [1/h]:

$$\lambda_{\mathrm{sensor}} = \frac{p_{\mathrm{av}}}{t_{\mathrm{crit}}} \tag{6.1}$$

$\lambda_{\mathrm{sensor}}$ is the expected number of sensor errors per time.

To summarize, with the presented problem formulation, the complement of *sensor perception reliability* is p_{av}. In the next section, it is described how to infer p_{av} and ρ without reference truth.

6.1.2 Sensor Perception Reliability Assessment without a Reference Truth

With only a single sensor and without additional information, it is not possible to identify whether a sensor error has occurred at point in time m or not. Consequently, the sensor error probability cannot be learned. However, when data from multiple redundant sensors is available, the sensor outputs can be compared with each other. Consider the following example: $n = 3$ sensors with overlapping field of views are implemented to detect relevant objects in a certain limited area in front of a vehicle. Failure to detect a relevant object in this area (FN) and also a wrong detection in this area (FP) are defined as sensor errors. With these definitions and the situation outlined in Section 6.1.1, only four relevant outcomes exist at point in time m: a) all sensors are correct $k = 0$; b) one sensor makes an error $k = 1$; c) two sensors make an error $k = 2$ and d) all sensors make an error $k = 3$. The different permutations of the outcomes b) and c) are irrelevant because it does not matter which particular sensor makes an error with identical sensors.

Without a reference truth, the outcomes $k = 0$ and $k = 3$ cannot be distinguished. For example, if all sensors indicate an object in the area of interest, it could be that this object exists in reality, thus all sensors would be correct. This means $k = 0$ has occurred. However, it could also be that the detected object does not exist. Then all sensors would make an error and $k = 3$ has occurred.

While K is unknown without reference truth, it is known how many of the sensors have matching output. If the output of all sensors matches – denoted with $z = 0$ – either $k = 0$ or $k = 3$ must have occurred. If the output of two sensors matches, denoted with $z = 1$, either $k = 1$ or $k = 2$ must have occurred. In other words, what can be observed without a reference truth is the random variable Z, a measure of how many sensors agree on the output. This random variable reflects the inability to discriminate between $k = 0$ and $k = 3$ or $k = 1$ and $k = 2$. The sample spaces of K and Z are summarized for $n = 3$ in Figure 6.1.

K-out-of-n errors:	0	1	2	3
Observation: Z	0	1	1	0

Figure 6.1 **With $n = 3$ sensors, the number of sensors K to make an error at point in time m can take values 0, 1, 2, 3. Without a reference truth, one cannot observe K but one can observe how many of the sensors have matching output. For a binary sensor task, if $k = 0$ or $k = 3$, the output of all sensors matches (observation $z = 0$). If $k = 1$ or $k = 2$, the output of two sensors matches (observation $z = 1$). Taken and adapted from [156].**

The observation Z has been introduced with the example of $n = 3$ sensors. In case of an arbitrary number of sensors n, the sample space is $Z \in \{0,1, \dots, \left\lceil \frac{n-1}{2} \right\rceil\}$. $Z = z$ if $\{K = z \cup K = n - z\}$.

For illustrative purposes, Figure 6.2 exemplarily shows how the probability of the observation $\Pr(Z = z|p_{\text{av}}, \rho)$ in a system of $n = 7$ sensors (Figure 6.2b) is related to the probability $\Pr(K = k|p_{\text{av}}, \rho)$ of exactly K sensor errors (Figure 6.2a) with the beta-binomial model. As will be demonstrated, one can estimate the model parameters and thus the full distribution in Figure 6.2a) solely by knowledge of Figure 6.2b) i.e. the frequency of the observations Z. Note that the error frequencies in Figure 6.2 are unrealistically large, they are selected for demonstrative purposes only.

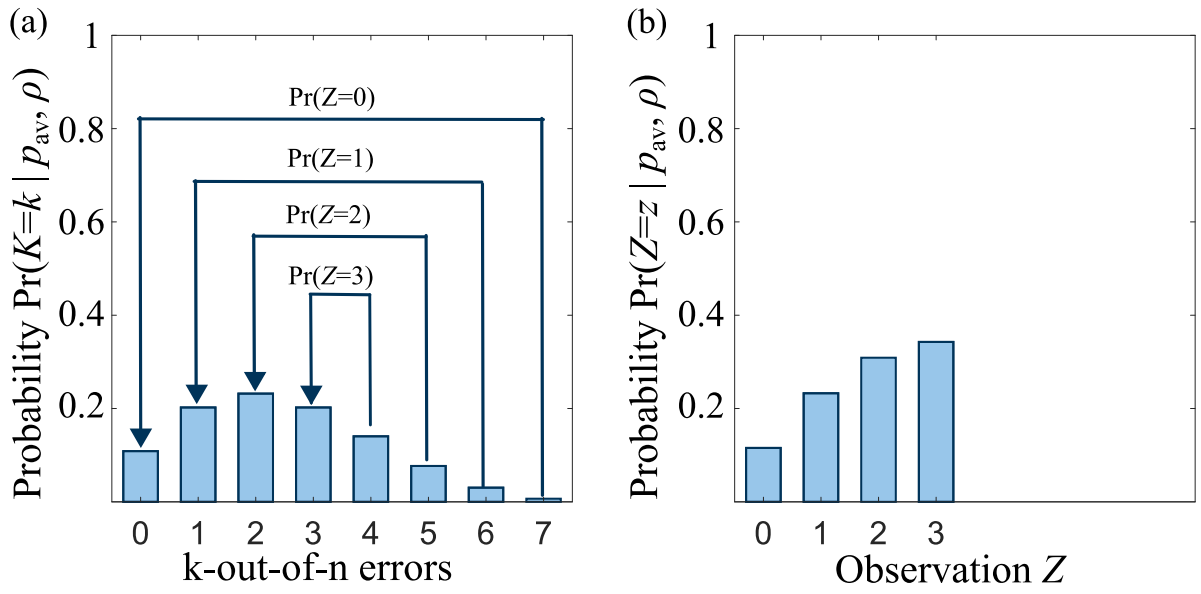

Figure 6.2 **(a) The probability of k-out-of-n sensor errors with the beta-binomial distribution and for exemplary purposes selected n=7, $p_{\text{av}} = 0.35$ and $\rho = 0.1$. These frequencies would be observed in a long test with a reference truth. (b) The corresponding statistically expected frequencies of the observations Z without a reference truth. Taken and adapted from [156].**

Bayes theorem is applied to learn the individual sensors' average error probability p_{av} as well as the sensors' error correlation coefficient ρ from the observations $\mathbf{z} = [z_1, z_2, \dots, z_M]$ [114]:

$$f(p_{\text{av}}, \rho|\mathbf{z}) \propto f(p_{\text{av}}, \rho) \cdot L_z(p_{\text{av}}, \rho) \tag{6.2}$$

where $f(p_{\text{av}}, \rho|\mathbf{z})$ is the joint posterior distribution of p_{av} and ρ given the test outcome $\mathbf{z}$, $f(p_{\text{av}}, \rho)$ is the joint prior distribution and

$$L_z(p_{\text{av}}, \rho) = \prod_{m=1}^{M} \Pr(Z = z_m|p_{\text{av}}, \rho) \tag{6.3}$$

the likelihood of (p_{av}, ρ). The likelihood $\Pr(Z = z_m|p_{\mathrm{av}}, \rho)$ for the single observation z_m in measurement cycle m is the probability of either exactly $K = z_m$ or exactly $K = n - z_m$ errors occurring:

$$\Pr(Z = z_m|p_{\mathrm{av}}, \rho) = \Pr(K = z_m|p_{\mathrm{av}}, \rho) + \Pr(K = n - z_m|p_{\mathrm{av}}, \rho) \tag{6.4}$$

The probabilities $\Pr(K = z_m|p_{\mathrm{av}}, \rho)$ and $\Pr(K = n - z_m|p_{\mathrm{av}}, \rho)$ are defined with Eqs. (4.26)-(4.27). For a given z_m, the formulation in Eq. (6.4) does not require to make a deterministic decision on whether $K = z_m$ or $K = n - z_m$ has occurred, instead it probabilistically accounts for the fact that both are possible. The likelihood in Eq. (6.4) could equivalently be formulated with any other suitable statistical model instead of the beta-binomial distribution in Eqs. (4.26)-(4.27).

In case of an even number of sensors n, there is one exception to Eq. (6.4). If half of the sensors agree on the output, $z_m = n/2$, the number of sensor errors is with certainty $k_m = n/2$. Consequently, the likelihood of the observation is in this case:

$$\Pr(Z = n/2|p_{\mathrm{av}}, \rho) = \Pr(K = n/2\ |p_{\mathrm{av}}, \rho) \tag{6.5}$$

To simplify Eq. (6.2), the number n_z is introduced. It denotes the number of observations with $Z = z$ in a test with a total of M observations. It is:

$$M = n_{z=0} + n_{z=1} + \cdots + n_{z=\max(Z)} \tag{6.6}$$

where $\max(Z) = \left\lceil \frac{n-1}{2} \right\rceil$. Eq. (6.3) is a multinomial likelihood, which can be expressed with sufficient statistics $n_{z=0}, n_{z=1}, \ldots, n_{z=\max(Z)}$ [115]. Therefore, Eq. (6.3) can be reformulated to:

$$L_z(p_{\mathrm{av}}, \rho) = \prod_{z=0}^{\max(z)} \Pr(Z = z|p_{\mathrm{av}}, \rho)^{n_z} \tag{6.7}$$

The product is now only of size $\left\lceil \frac{n+1}{2} \right\rceil$ compared to size M in Eq. (6.3).

If the number of measurement cycles M becomes large, it is more convenient to work with the log-posterior distribution, to prevent underflow, i.e. Eq. (6.2) is replaced with:

$$\ln[f(p_{\mathrm{av}}, \rho|\mathbf{z})] \propto \ln[f(p_{\mathrm{av}}, \rho)] + \sum_{z=0}^{\max(z)} n_z \cdot \ln[\Pr(Z = z|p_{\mathrm{av}}, \rho)] \tag{6.8}$$

The log-posterior Eq. (6.8) can readily be evaluated on a numerical grid. Alternatively, Markov Chain Monte Carlo (MCMC) [114, 115, 266, 267] allows to obtain samples from the joint posterior $f(p_{\mathrm{av}}, \rho|\mathbf{z})$.

Eqs. (6.2) and (6.8) describe the uncertainty in the average sensor error probability p_{av} and the sensor error correlation coefficient ρ for a specific test outcome without a reference truth. If the interest is in the individual average sensor error probability, the marginal posterior PDF of p_{av} is obtained from Eq. (6.2) by integrating over the uncertainty in the error correlation coefficient ρ:

$$f(p_{\mathrm{av}}|\mathbf{z}) = \int_0^1 f(p_{\mathrm{av}}, \rho|\mathbf{z})\, d\rho \tag{6.9}$$

$f(\rho|\mathbf{z})$ is obtained analogously. From Eq. (6.9), the posterior mean $\hat{p}_{\mathrm{av}}$ of the average sensor error probability and similarly, from $f(\rho|\mathbf{z})$ the posterior mean of the sensor error correlation coefficient $\hat{\rho}$ are readily derived. These are point estimates of the sensor perception (un-)reliability and dependence for the given test outcome $\mathbf{z}$. Alternatively, point estimates based on the median, on the posterior mode or a specific quantile could be selected to summarize the individual *sensor perception reliability*.

By utilizing Bayesian parameter inference [114], we also quantify the uncertainty associated with the parameter estimation. The uncertainty in the estimation of p_{av} or ρ is expressed with credible intervals as the (central) intervals that contain γ% of the probability density in $f(p_{\mathrm{av}}|\mathbf{z})$ or $f(\rho|\mathbf{z})$.

6.1.3 Demonstrating Perception Reliability by Exploiting Sensor Redundancy

To demonstrate *perception reliability* without reference truth by exploiting sensor redundancy, the k-out-of-n model of Section 4.2.1 is employed to represent the fusion model. We assume that the perception module fails to provide correct information at a given point of time when more than half of the individual sensors make an error, i.e. the perception is validated by means of a majority vote of the individual sensors. The k-out-of-n model is, however, not necessary to learn p_{av} and ρ of the individual sensors without the reference truth. The methodology presented in Section 6.1.2 is valid without the assumption of a majority vote for the perception module.

With the majority vote, the perception module fails if:

$$K \geq \left\lfloor \frac{n}{2} + 1 \right\rfloor \tag{6.10}$$

where K is the number of sensor errors in a sensor system consisting of n redundant sensors, as introduced in Section 6.1.1.

With the probability of exactly k-out-of-n sensor errors being defined by Eq. (4.26) and its parameters by Eq. (4.27), the probability $p_{\mathrm{per}}(p_{\mathrm{av}}, \rho)$ of the perception module to fail at a randomly selected point in time is the probability that the majority of individual sensors provide incorrect information [65]:

$$p_{\mathrm{per}}(p_{\mathrm{av}}, \rho) = \sum_{k=\left\lfloor \frac{n}{2}+1 \right\rfloor}^{n} \Pr(K = k | p_{\mathrm{av}}, \rho) \tag{6.11}$$

$p_{\mathrm{per}}(p_{\mathrm{av}}, \rho)$ implies that the perception module's failure probability p_{per} is a function of the individual average sensor error probability p_{av} as well as the sensor error correlation coefficient ρ.

Let the perception module's TLS be $\lambda_{\mathrm{TLS_{per}}}$, which can be transformed to a TLS $p_{\mathrm{TLS_{per}}}$ on p_{per} with Eq. (6.1). Hence, it has to be demonstrated that the probability p_{per} of the perception module to fail complies with $p_{\mathrm{TLS_{per}}}$. In a deterministic setting, when p_{av} and ρ are known with certainty, one would accept the system if

$$p_{\mathrm{per}}(p_{\mathrm{av}}, \rho) \leq p_{\mathrm{TLS_{per}}} \tag{6.12}$$

otherwise the system would be rejected. In reality, the parameters p_{av} and ρ cannot be known with certainty. They have to be inferred e.g. according to Section 6.1.2 from a limited amount of data obtained through tests. The methodology to jointly account for uncertainties in multiple parameters is subsequently presented for the simplified problem interpretation of Section 6.1.1 in case no reference truth is available, but it is analogous for another problem formulation such as Section 5.4 for tests with reference truth.

To decide whether a test summarized by Eq. (6.2) indicates compliance with $p_{\mathrm{TLS_{per}}}$, the domain of all acceptable combinations of (p_{av}, ρ) is identified. Dividing the outcome space of the involved model parameters into a failure and a safe domain (here, the safe domain is termed acceptable domain) is common in structural reliability [274] and is transferred to the problem of demonstrating *perception reliability*. Based on Eqs. (6.11)-(6.12) the acceptable domain is formulated as:

$$\left\{ p_{\mathrm{per}}(p_{\mathrm{av}}, \rho) - p_{\mathrm{TLS_{per}}} \leq 0 \right\} \tag{6.13}$$

The surface $p_{\mathrm{per}}(p_{\mathrm{av}}, \rho) - p_{\mathrm{TLS_{per}}} = 0$ divides all acceptable combinations of (p_{av}, ρ) from all inacceptable combinations and is termed limit state surface (LSS). An exemplary LSS is schematically illustrated in Figure 6.3 for an arbitrary value of $p_{\mathrm{TLS_{per}}}$.

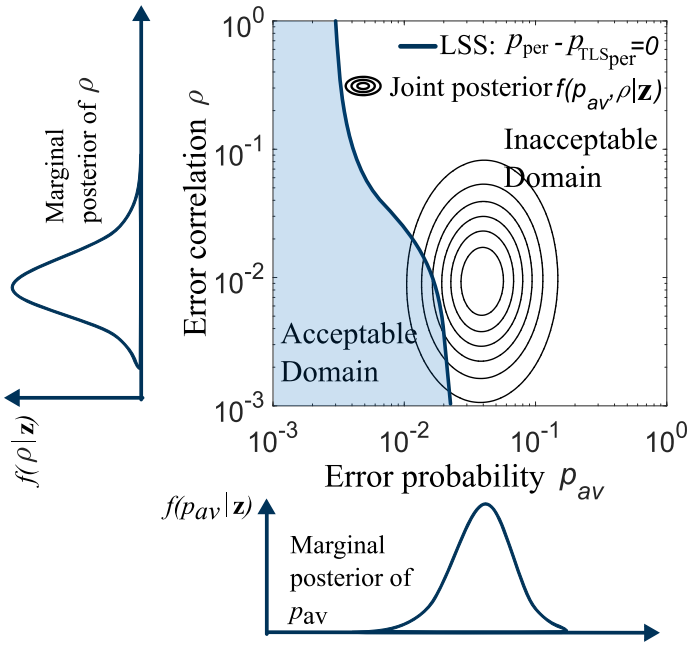

Figure 6.3 Illustration of the perception reliability validation: The limit state surface (LSS) divides all combinations of (p_{av}, ρ) leading to satisfactory system safety from all combinations of (p_{av}, ρ) that would lead to violations of desired system safety. After a reliability demonstration test, the probability of complying with the target level of safety $p_{TLS_{per}}$ is defined by the integral over the posterior PDF $f(p_{av}, \rho|\mathbf{z})$ in the acceptable domain (shaded area). Taken and adapted from [156].

All combinations of (p_{av}, ρ) on the left side of the LSS lead to failure probabilities complying with the target level of safety, i.e. $p_{per}(p_{av}, \rho) \leq p_{TLS_{per}}$. In contrast, all combinations of (p_{av}, ρ) on the right side of the LSS lead to violations of the target level of safety, hence they constitute the inacceptable domain. The ellipses in the Figure are exemplary contours of the joint posterior $f(p_{av}, \rho|\mathbf{z})$ according to Eq. (6.2) for a given test outcome $\mathbf{z}$.

The Bayesian probability of the system to comply with its target level of safety $p_{TLS_{per}}$ is quantified by the posterior probability density $f(p_{av}, \rho|\mathbf{z})$ in the acceptable domain. Given the outcome of the test $\mathbf{z}$, the system's compliance probability is therefore the integral (shaded area in Figure 6.3).

$$\Pr\left(p_{per} \leq p_{TLS_{per}}|\mathbf{z}\right) = \int_{p_{per}(p_{av},\rho)-p_{TLS_{per}}\leq 0} f(p_{av}, \rho|\mathbf{z})\, dp_{av}\, d\rho \tag{6.14}$$

This integral can be evaluated numerically or through MCMC.

Sampling p_{av}, ρ from the posterior distribution $f(p_{av}, \rho|\mathbf{z})$, Eq. (6.2), and inserting into (6.11) allows to derive the posterior PDF of the perception module's failure probability $f(p_{per}|\mathbf{z})$. To obtain a point estimate of p_{per}, the posterior mean $\hat{p}_{per}$ is derived from $f(p_{per}|\mathbf{z})$. Alternatively, the median or an upper credible bound on p_{per} could be selected as point estimate.

The probability of complying with the target level of safety *after a specific* test, given by Eq. (6.14) is the aim of the *perception reliability* assessment. If for a given test outcome $\Pr\left(p_{\text{per}} \leq p_{\text{TLS}_{\text{per}}} | \mathbf{z}\right)$ is sufficiently large, meaning one has high credibility in the system complying with its reliability requirements, the system is accepted. The reliability test decision rule is in analogy to Eq. (4.43)[72]:

$$\text{Accept the system if } \Pr\left(p_{\text{per}} \leq p_{\text{TLS}_{\text{per}}} | \mathbf{z}\right) \geq \gamma \tag{6.15}$$

Reject otherwise. γ is a pre-specified level of credibility. This procedure is equivalent to a lower Bayesian credible bound of the perception module's reliability complying with the TLS. In other words, the applied reliability demonstration test plan is the Bayesian analogue to the frequentist lower confidence bound of the system reliability complying with the TLS (see e.g. [285]).

6.1.4 Challenges Associated with Statistical Dependence among Sensor Errors

In this Section, we discuss the challenges associated with the dependence among sensor errors in an analysis without reference truth. Firstly, strong error correlations among sensors makes it difficult to learn *sensor perception reliabilities* by exploiting sensor redundancies. Secondly, and more critical, it is much more challenging to identify whether or not the correct statistical model for sensor error dependence is employed.

With a single sensor, it is not possible to identify the sensor error probability without a reference truth. Therefore, if all sensors were fully dependent, $\rho = 1$, the error probability of the sensors could not be learned from the data without a reference truth, as the observation would be that the output of the sensors matches at all times ($z = 0$). It is however possible to correctly find p_{av} even for correlations close to one ($\rho \to 1$) without a reference truth, if sufficient data is available. Only the case $\rho = 1$ makes it strictly impossible to learn p_{av}.

A larger challenge is the choice of the statistical model for sensor error dependence. In Section 6.1.1 we utilize the beta-binomial model. As with any statistical analysis, if the selected statistical model is not adequately representing the real statistical process, spurious inference might be the result. The problem of the analysis without a reference truth is that it is much more difficult to identify if an inadequate statistical model has been selected. To illustrate this challenge and to

[72] This decision rule limits the probability of a specific released system not to comply with the safety targets to 1- γ for any given test outcome. An individual decision maker following this rule therefore is not wrong more often than $(1 - \gamma)$ in the long run (provided the model assumptions are adequate). This decision rule is not optimal. An optimum decision in a Bayesian sense – and from a cost optimization point of view – would be to accept a system if the posterior mean system failure probability $\hat{p}_{\text{per}}$ complies with the target level of safety ($\hat{p}_{\text{per}} \leq p_{\text{TLS}_{\text{per}}}$). Then, for a large number of released systems, the (average) failure frequency of all systems is complying with the (average) target level of safety. This cost optimal decision rule would however expose the individual decision maker to a larger risk of being wrong in a specific decision (≤ 50 % if the posterior system failure probability is symmetrically distributed).

examine the consequences of choosing an inadequate statistical model, we additionally consider an alternative dependence model.

In addition to the common failure events described by the beta-binomial model, the alternative dependence model accounts for critical shock events that occur with a probability p_s. A shock event causes all the sensors to make an error simultaneously. The term shock event is utilized by the reliability engineering community in the context of common cause failures [289] and can be understood as an abstract collection of causes leading to a joint failure event. Given a critical shock event occurs, all components fail deterministically. The probability of k failures is then:

$$\Pr(K = k|p_\mathrm{s}, p_\mathrm{av}, \rho) = p_\mathrm{s} \cdot \delta_{k,n} + (1 - p_\mathrm{s}) \cdot \Pr(K = k|p_\mathrm{av}, \rho) \tag{6.16}$$

where $\delta_{k,n}$ is the Kronecker delta, which is $\delta_{k,n} = 1$ if $k = n$ and $\delta_{k,n} = 0$ otherwise. $\Pr(K = k|p_\mathrm{av}, \rho)$ is the beta-binomial model in Eqs. (4.26)-(4.27). The model in Eq. (6.16) is essentially a mixture model between beta-binomial distributed common failure events and shocks. A similar general formulation of dependence structures with mixture models and dirac functions can be found in [290]. The model of Eq. (6.16) in discretized time can be interpreted in analogy to the beta-factor model [291] and its generalization, the binomial failure rate (BFR) model [222, 292, 293] in continuous time.

If the true statistical error process is accurately described by Eq. (6.16), but one assumes Eqs. (4.26)-(4.27) as the basis for the estimation without reference truth according to Section 6.1.2, one potentially severely underestimates the probability of all sensors to make an error simultaneously ($K = n$). The reason for the underestimation lies in the inability to distinguish the case of no sensor errors ($k = 0$) from all sensors to make an error ($k = n$) without a reference truth, as these both lead to the observation $z = 0$. For example, when p_av and ρ are small but p_s is large, then $\Pr(K = n)$ is large. Eqs. (4.26)-(4.27) however do not allow for a large $\Pr(K = n)$ if p_av and ρ are small. With this specific example, the combination of applying Eqs. (4.26)-(4.27) with the inability of distinguishing ($k = n$) from ($k = 0$) without reference would lead to an underestimation of $\Pr(K = n)$. The consequences of choosing an inadequate statistical model with the presented framework are further studied in case studies and discussed in Section 6.1.6.

6.1.5 Case Studies

Synthetic case studies are conducted to illustrate the presented ideas. The perception module of the case studies consists of $n = 7$ identical redundant sensors. For demonstrative purposes, the perception module's TLS is selected exemplary as $p_{\mathrm{TLS}_\mathrm{per}} = 10^{-4}$. This requirement is not in accordance with common risk acceptance criteria (see Section 3.1.1), but is here selected nevertheless, as the goal is to demonstrate the validity of the presented framework.

The aim of the first three case studies is to show that Eq. (6.2) allows to estimate the individual sensors' average error probability p_{av} and the sensor error correlation coefficient ρ under the model of Eqs. (4.26)-(4.27) without a deterministic reference truth. To this end, different virtual truth scenarios are selected for p_{av} and ρ, which are examined in the case study: 1) a combination of (p_{av}, ρ) that leads to compliance with the target level of safety $p_{\mathrm{per}} \leq p_{\mathrm{TLS_{per}}} = 10^{-4}$, 2) a combination of (p_{av}, ρ) that does not comply with the target level of safety $p_{\mathrm{per}} > p_{\mathrm{TLS_{per}}} = 10^{-4}$ and 3) sensors with almost full error dependence. This means the correlation coefficient is $\rho \approx 1$.

In a fourth case study, the effect of selecting an inadequate statistical sensor error dependence model is studied. To this end, critical shock events are simulated to occur according to Eq. (6.16) i.e. the virtual truth is $p_{\mathrm{s}} \neq 0$, but we purposely assume $p_{\mathrm{s}} = 0$. The parameters of the different virtual truth scenarios are summarized in Table 5.2.

Table 6.1 Case study: Virtual truth of the average sensor error probability p_{av}, the error correlation coefficient ρ, the shock probability p_{s} and the corresponding perception module's failure probabilities p_{per} under a majority vote. From [156].

Case	Sensor error probability p_{av}	Error correlation ρ	Shock probability p_{s}	Perception failure probability p_{per}
1)	10^{-4}	0.01	0	$1.89 \cdot 10^{-8}$
2)	10^{-2}	0.20	0	$2.8 \cdot 10^{-3}$
3)	10^{-3}	0.99	0	10^{-3}
4)	10^{-2}	0.20	0.1	0.103

Synthetic datasets are derived with the true parameter values $(p_{\mathrm{av}}, \rho, p_{\mathrm{s}})$ of Table 5.2 by randomly sampling the number of sensors k_m to make an error at different points in time $m = 1,2, \ldots, M$. The sampling distribution of k_m is the beta-binomial distribution defined in Eqs. (4.26)-(4.27) with parameters (p_{av}, ρ) for cases 1-3 and the shock model of Eq. (6.16) for case 4. The resulting datasets $\mathbf{k}$ of the four cases are transformed into the observation z_m if $\{k_m = z_m \cup k_m = n - z_m\}$, which simulates that the exact number k_m of sensors making an error cannot be observed without a reference truth, as explained in Section 6.1.2. With this approach, it is simulated that only $\mathbf{z}$ and not $\mathbf{k}$ are known after the test. Figure 6.4 schematically illustrates the structure of the synthetic datasets.

Discretized time

Index of discretized time	**1**	**2**	**3**	**4**	**5**	**6**	**7**	**8**	**…**	***m***	**…**
k_m-out-of-n errors	**0**	**1**	**2**	**0**	**7**	**5**	**4**	**3**	**…**	**6**	**…**
Observation: z_m	**0**	**1**	**2**	**0**	**0**	**2**	**3**	**3**	**…**	**1**	**…**

Figure 6.4 Schematic illustration of the synthetic datasets. For each point in time m, the number of sensor errors k_m are sampled from Eqs. (4.26)-(4.27), or Eq. (6.16) respectively, with known parameters according to Table 5.2. To simulate that no reference truth is available, k is transformed into the observations z without a reference truth. Taken and adapted from [156].

The prior distribution $f(p_{\text{av}}, \rho)$ in Eq. (6.8) is selected as a uniform prior in the domain $p_{\text{av}} < 0.5$:

$$f(p_{\text{av}}, \rho) = \begin{cases} 1, & if\ p_{\text{av}} < 0.5 \\ 0, & otherwise \end{cases} \tag{6.17}$$

The restriction to $p_{\text{av}} < 0.5$ is necessary, since the likelihood is symmetric around $p_{\text{av}} = 0.5$, i.e. the learning process cannot distinguish between e.g. $p_{\text{av}} = 0.01$ and $p_{\text{av}} = 0.99$. However it is safe to assume that no sensor is considered for a safety-relevant task, if it commits an error more often than every second discrete point in time ($p_{\text{av}} \geq 0.5$). Such a high frequency of errors would be detected in the development and the sensor would thus be rejected before the reliability analysis. The prior is here naïvely selected to be weakly informative, a different prior such as Jeffreys prior could be derived following [243, 244]. The idea behind the presented framework is to learn *sensor perception reliabilities* from a large fleet of human controlled vehicles without a reference truth, leading to a large amount of data. As the choice of the prior becomes irrelevant with a finite number of model parameters in the limit of $M \to \infty$ [114], the modeling of the prior is not further investigated.

For virtual truth cases 1-3, it is checked whether the parameter estimates based on the framework in Section 6.1.2 converge against the true values as given in Table 5.2. Moreover, the probability of the perception module to comply with the target level of safety $p_{\text{TLS}_{\text{per}}}$ is derived as described in Section 6.1.3. In virtual truth case 4, the effect of learning the parameters of Eqs. (4.26)-(4.27) without reference truth is studied, despite $p_s \neq 0$.

Results case study 1: $p_{\text{av}} = 10^{-4}$, $\rho = 10^{-2}$

First, the posterior distribution $f(p_{\text{av}}, \rho|\mathbf{z})$ is set up with Eq. (6.8) and the first $M = [10^2, \dots, 10^7]$ (synthetic) observations z. The marginal posteriors $f(p_{\text{av}}|\mathbf{z})$ and $f(\rho|\mathbf{z})$ are calculated with Eq. (6.9) for each M. Additionally, to demonstrate compliance with $p_{\text{TLS}_{\text{per}}} = 10^{-4}$, the limit state surface (LSS) is derived according to Section 6.1.3. Results of these evaluations are presented in Figure 6.5 which corresponds to $M = 10^3$.

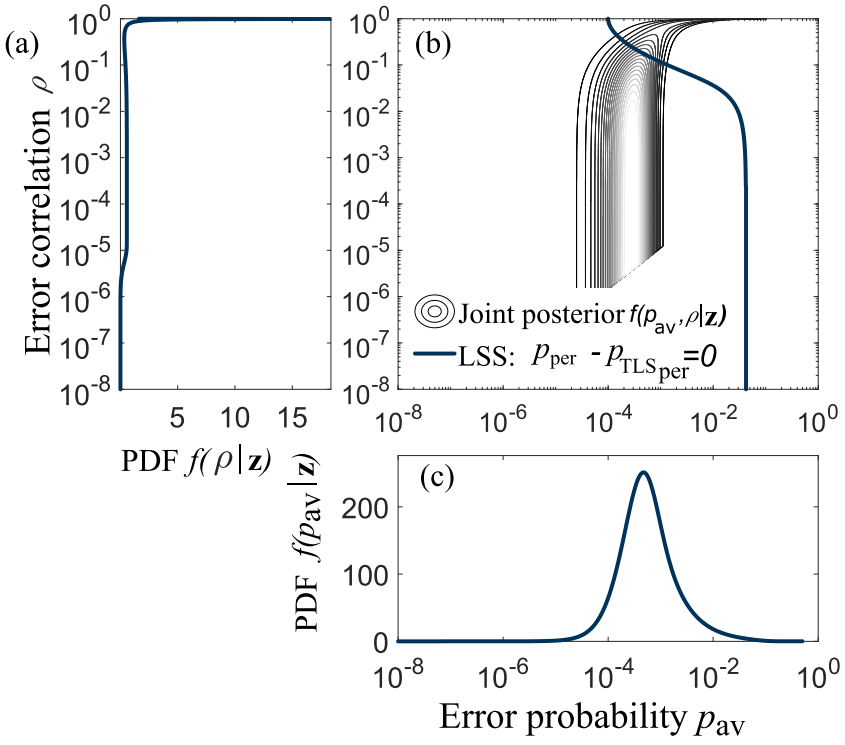

Figure 6.5 **Virtual truth case 1 after $M = 10^3$ observations z. (a) marginal posterior of sensor error correlation coefficient $f(\rho|\mathbf{z})$; (b) joint posterior $f(p_{av}, \rho|\mathbf{z})$. (c) marginal posterior of sensor error probability $f(p_{av}|\mathbf{z})$. Also shown in (b) is the limit state surface (LSS) that separates the acceptable (left) from the inacceptable (right) domain with the given system target level of safety $p_{TLS_{per}} = 10^{-4}$. Taken and adapted from [156].**

In Figure 6.5b), the contour lines reflect the uncertainty in the average sensor error probability p_{av} and the sensor error correlation coefficient ρ after $M = 10^3$ observations. Both the average sensor error probability p_{av} and the sensor error correlation coefficient ρ are still subject to substantial uncertainty. As visible in Figure 6.5a), ρ is assigned a marginal posterior probability density between 10^{-6} and one, with a spike close to one. The spike in the density close to one is due to the fact that up to $M = 10^3$, the sensors had 998 times matching observations $z = 0$. Figure 6.5c) illustrates that the marginal posterior density of p_{av} is confined between 10^{-5} and 0.5 with its mode around 10^{-3}. It is pointed out that in the domain of $p_{av} \approx 10^{-4}$ and $\rho \leq 10^{-6}$ the beta-binomial distribution is here not defined because of numerical reasons.

Due to the logarithmic scale of Figure 6.5, it is not directly visible that most of the joint posterior density is concentrated in the unacceptable domain on the right side of the LSS, leading to a compliance probability of $\Pr\left(p_{per} \leq p_{TLS_{per}}|\mathbf{z}\right) = 0.08$ with Eq. (6.14). Because of the uncertainties in the parameters, the compliance probability is not yet conclusive.

Figure 6.6 presents the joint posterior distribution $f(p_{av}, \rho|\mathbf{z})$ after $M = 10^6$ observations. The posterior means $\hat{p}_{av} = 1.03 \cdot 10^{-4}$ and $\hat{\rho} = 0.014$ are close to their true values and the remaining

uncertainty is small. All the posterior density is in the acceptable domain, thus the probability of compliance is $\Pr\left(p_{\text{per}} \leq p_{\text{TLS}_{\text{per}}} | \mathbf{z}\right) = 1$.

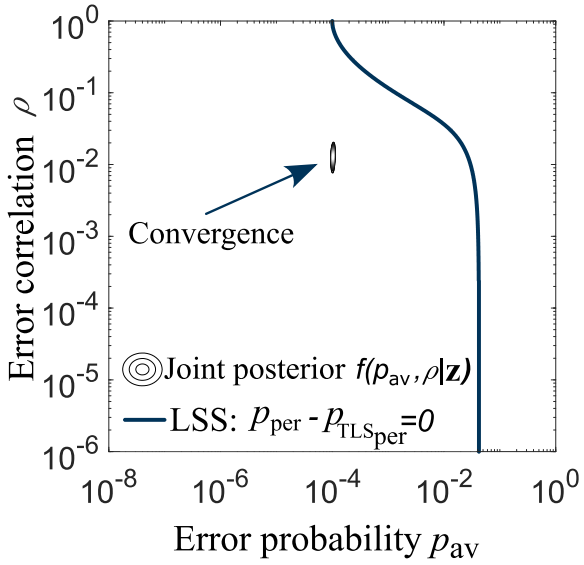

Figure 6.6 Joint posterior distribution $f(p_{\text{av}}, \rho|\mathbf{z})$ of virtual truth case 1 after $M = 10^6$ observations z. $f(p_{\text{av}}, \rho|\mathbf{z})$ has converged towards the true values $p_{\text{av}} = 10^{-4}$ and $\rho = 10^{-2}$. Taken and adapted from [156].

To obtain a clearer picture of the effect of data size, the posterior means $\hat{p}_{\text{av}}$ and $\hat{\rho}$ are tracked in Figure 6.7 over an increasing number of observations M. Figure 6.7a) displays that $\hat{p}_{av}$ converges to its true value of $p_{\text{av}} = 10^{-4}$ within approximately $M = 3 \cdot 10^4$ observations. Figure 6.7b) demonstrates that ρ is more difficult to learn, $\hat{\rho}$ requires between $M = 10^5$ and $M = 10^6$ observations to approach its true value of $\rho = 10^{-2}$.

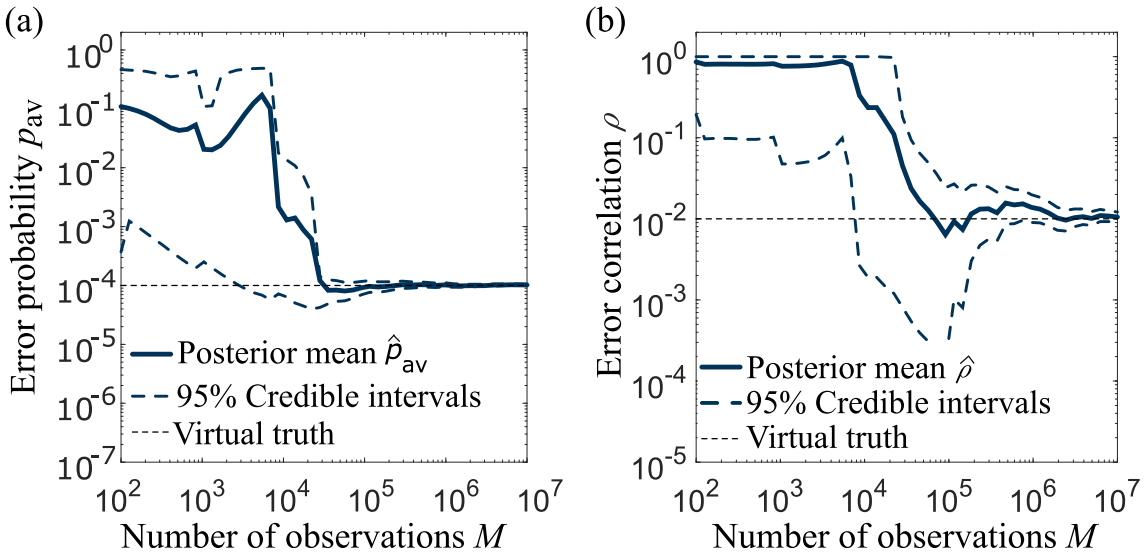

Figure 6.7 Virtual truth case 1 with increasing number of observations M. (a) Posterior mean $\hat{p}_{\text{av}}$ and (b) posterior mean $\hat{\rho}$ with corresponding 95% credible intervals. Taken and adapted from [156].

To demonstrate the uncertainty in p_{per} at a specific point of time, samples (p_{av}, ρ) are drawn from the posterior distribution $f(p_{\text{av}}, \rho|\mathbf{z})$. These samples are inserted into Eq. (6.11) to obtain samples of the perception failure probability p_{per}. These samples define the posterior PDF $f(p_{\text{per}}|\mathbf{z})$ of the perception failure probability as described in Section 6.1.3. For $M = 10^6$ observations, the posterior PDF $f(p_{\text{per}}|\mathbf{z})$ of the perception failure probability is presented in Figure 6.8 which corresponds to the posterior $f(p_{\text{av}}, \rho|\mathbf{z})$ in Figure 6.6.

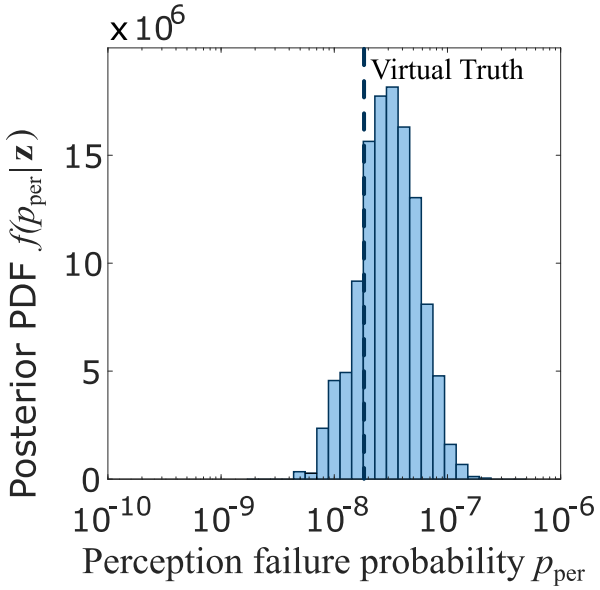

Figure 6.8 Virtual truth case 1. Sampled posterior PDF $f(p_{\text{per}}|\mathbf{z})$ of the perception failure probability after $M = 10^6$ observations z. The virtual truth $p_{\text{per}} = 1.89 \cdot 10^{-8}$ is indicated by the dashed line. Taken and adapted from [156].

$f(p_{\text{per}}|\mathbf{z})$ is approximately centered around the virtual truth of $p_{\text{per}} = 1.89 \cdot 10^{-8}$ with a posterior mean of $\hat{p}_{\text{per}} = 4.9 \cdot 10^{-8}$ and a 95% credible interval of $[p_{\text{per}_{\text{low}}} = 1.3 \cdot 10^{-8}, p_{\text{per}_{\text{up}}} = 1.2 \cdot 10^{-7}]$. Hence, the true system failure probability $p_{\text{per}} = 1.89 \cdot 10^{-8}$ is identified with limited uncertainty. Furthermore, it is clear that the requirement $p_{\text{per}} \leq p_{\text{TLS}_{\text{per}}} = 10^{-4}$ is complied with.

To obtain an idea of how much testing is required to identify that the system complies with its target level of safety $p_{\text{TLS}_{\text{per}}} = 10^{-4}$, the compliance probability $\Pr\left(p_{\text{per}} \leq p_{\text{TLS}_{\text{per}}}|\mathbf{z}\right)$ is plotted against the number of observations M in Figure 6.9a). Starting around $M = 4.5 \cdot 10^4$, the compliance probability is $\Pr\left(p_{\text{per}} \leq p_{\text{TLS}_{\text{per}}}|\mathbf{z}\right) \geq 0.99999$ with which one has near certainty of complying with $p_{\text{TLS}_{\text{per}}}$. In Figure 6.9b) the corresponding posterior mean $\hat{p}_{\text{per}}$ is plotted in function of the number of observations M. Figure 6.9b) is derived analogous to Figure 6.8 but with varying numbers of observations M. The 95% credible intervals of the perception failure probability illustrate the uncertainty in the estimate. It takes around $M = 10^5$ observations for $\hat{p}_{\text{per}}$

to be near its true value (dotted line). After $M = 10^7$ observations, the uncertainty is negligible and the true value has been identified.

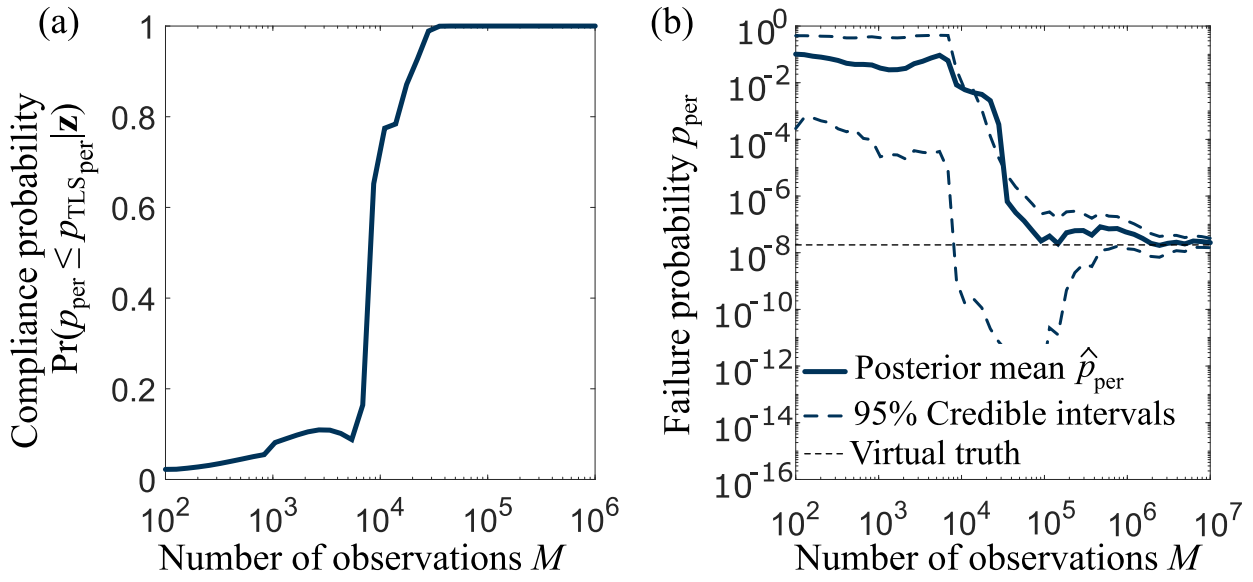

Figure 6.9 Virtual truth case 1 with increasing number of observations M. (a) Compliance probability $\Pr\left(p_{per} \leq p_{TLS_{per}}|\mathbf{z}\right)$. (b) Posterior mean $\hat{p}_{per}$ of the perception failure probability with the 95% credible intervals. Taken and adapted from [156].

Results case study 2: $p_{av} = 10^{-2}$, $\rho = 0.2$

The analysis presented for case 1 is repeated for virtual truth case 2. To summarize the results of the parameter inference, Figure 6.10 shows the posterior mean sensor error probability $\hat{p}_{av}$ and sensor error correlation coefficient $\hat{\rho}$ over the number of observations M. Additionally, the estimated perception failure probability p_{per} is presented in Figure 6.10c).

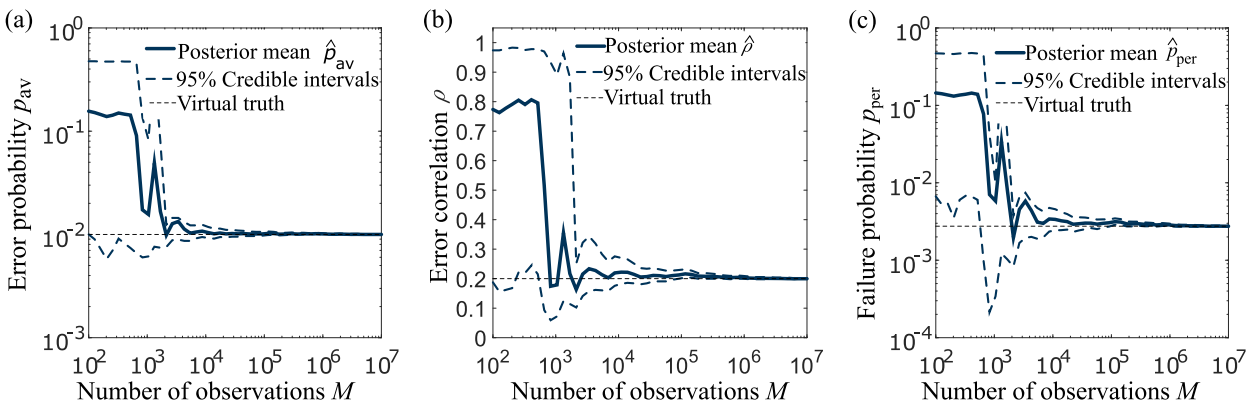

Figure 6.10 Virtual truth case 2 with increasing number of observations M. (a) Posterior mean $\hat{p}_{av}$, (b) posterior mean $\hat{\rho}$ and (c) posterior mean $\hat{p}_{per}$ with corresponding 95% credible intervals. Taken and adapted from [156].

Similar to case 1, the virtual truth $p_{\mathrm{av}} = 10^{-2}$ (Figure 6.10a) and $\rho = 0.2$ (Figure 6.10b) are learned correctly. Compared to virtual truth case 1, the parameters are learned faster because of the high error probability p_{av}. Figure 6.10c) illustrates that the correct failure probability of the perception module $p_{\mathrm{per}} = 2.8 \cdot 10^{-3}$ is identified after less than $M = 10^4$ observations. The compliance probability is $\Pr\left(p_{\mathrm{per}} \leq p_{\mathrm{TLS_{per}}} | \mathbf{z}\right) \approx 0$ after $M = 3.4 \cdot 10^3$ observations.

Results case study 3: $p_{\mathrm{av}} = 10^{-3}$, $\rho = 0.99$

The previous case studies have numerically demonstrated the validity of the framework presented in Sections 6.1.2 and 6.1.3. Here, the case of almost fully statistically dependent sensor errors with $\rho = 0.99$ is examined. Again, the analysis according to Section 6.1.2 is repeated. The resulting joint posterior distribution $f(p_{\mathrm{av}}, \rho | \mathbf{z})$ with $M = 10^7$ observations is reported in Figure 6.11.

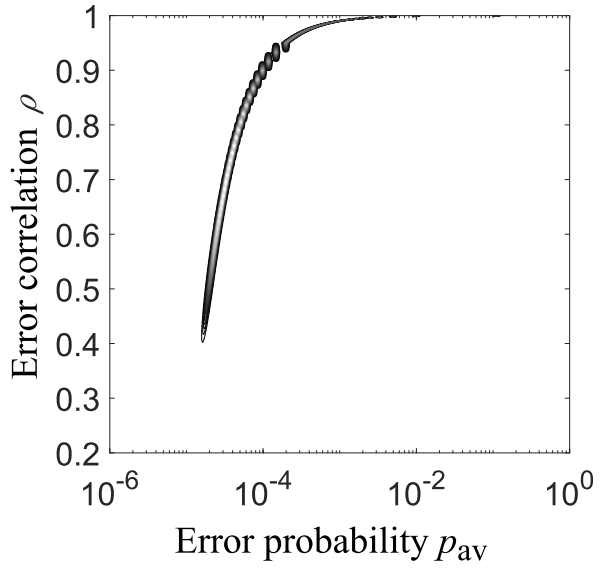

Figure 6.11 Joint posterior distribution $f(p_{\mathrm{av}}, \rho | \mathbf{z})$ of virtual truth case 3 after $M = 10^7$ observations. The posterior has not yet converged against the virtual truth of $p_{\mathrm{av}} = 10^{-3}$ and $\rho = 0.99$.

The parameter inference presented in Figure 6.11 has not yet converged after $M = 10^7$ observations.

The $M = 10^7$ synthetic observations $\mathbf{z}$ utilized in this case study are based on sampling from the virtual truth, as explained under Section 6.1.5. To demonstrate that convergence is ultimately achieved with more data, the sampled dataset is substituted with the expected number $E[N_{Z=z} | p_{\mathrm{av}}, \rho, M]$ of observations $Z = z$, given the virtual truth and the number of observations M. The expected number of observations $E[N_{Z=z} | p_{\mathrm{av}}, \rho, M]$ is derived by multiplying M with Eq. (6.4). This approach still allows to evaluate whether convergence is to be expected with a large

number of observations without relying on sampling (which becomes computationally too expensive for $M \gg 10^7$).

The parameter inference based on the expected number of observations is illustrated in Figure 6.12. It is verified that eventually the correct parameters can be learned.

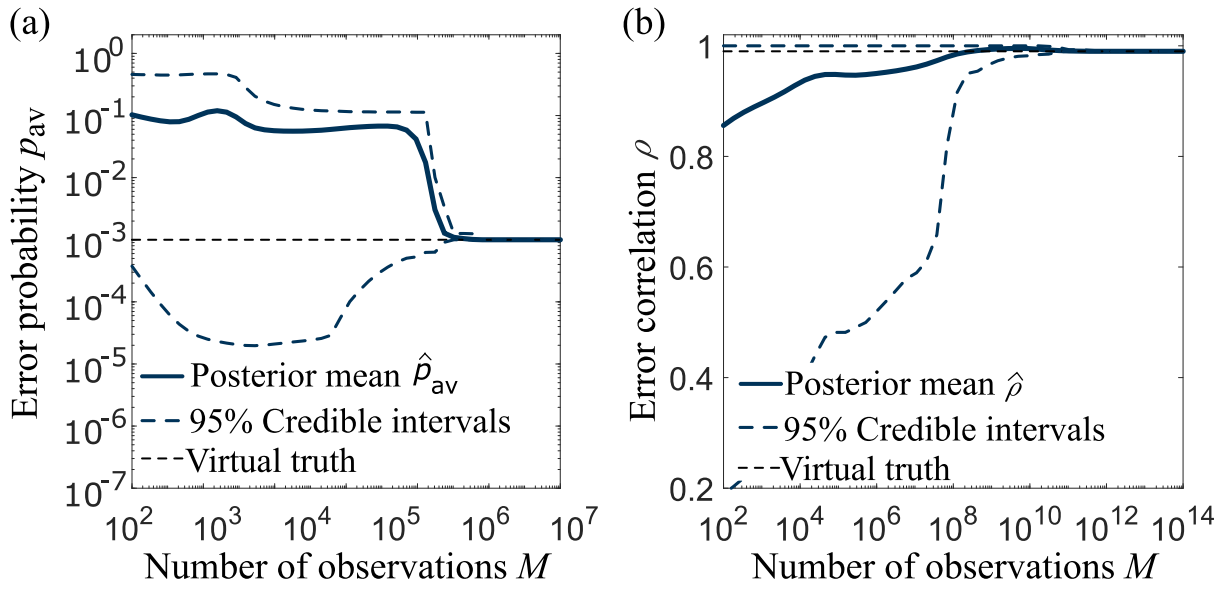

Figure 6.12 Virtual truth case 3 with increasing number of observations M. (a) Posterior mean $\hat{p}_{\text{av}}$, (b) posterior mean $\hat{\rho}$ with the corresponding 95% credible intervals. The underlying observations are equal to their expected values. Taken and adapted from [156].

The behavior in Figure 6.12 with $\rho \to 1$ can be explained with the beta distribution of the error probabilities p_m in the different points in time m according to Eq. (4.25). First, when $\rho = 1$, then the sensor error probability at a randomly selected point in time is $p_m = 1$ with a probability of p_{av} and $p_m = 0$ otherwise. This is equivalent to a Bernoulli trial in which either all or none of the sensors make an error, therefore, the observation is $z = 0$ for all m. As mentioned in Section 6.1.4, it is therefore not possible to learn the sensor error probability with fully dependent sensors. Still ρ can be correctly identified.

In case ρ is slightly smaller than one, in rare instances p_m takes values substantially different from zero and one, which could lead to observations $z \neq 0$. This is illustrated in Figure 6.13 by the beta CDF of p_m with virtual truth case 3. There is a small probability of p_m to be between $10^{-20} \leq p_m \leq 1$. Eventually, from these observations, the correct p_{av} and ρ are learned, as shown in Figure 6.12.

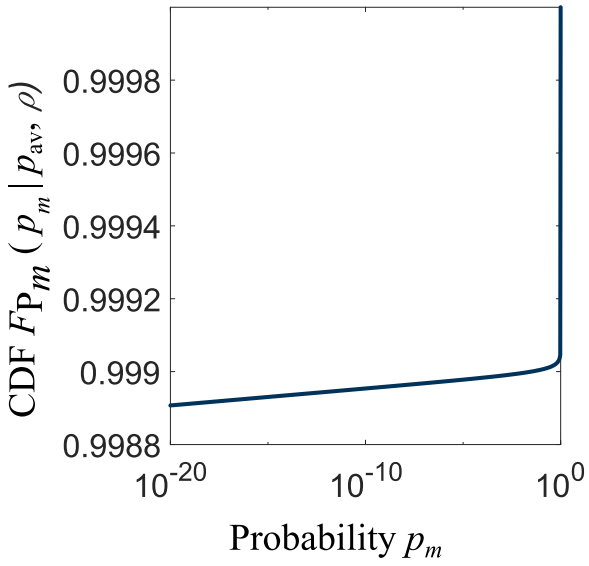

Figure 6.13 Beta CDF $F_{P_m}(p_m|p_{av} = 10^{-3}, \rho = 0.99)$ of the sensor error probability p_m in randomly selected points of time m. With $\rho \to 1$, p_m is either one with probability p_{av} or zero otherwise. Because here ρ is not exactly one, a small probability exists for p_m to be different from zero and one. Taken and adapted from [156].

Results case study 4: $p_{av} = 10^{-2}, \rho = 0.2, p_s = 0.1$

To study the effect of selecting an inadequate statistical model when learning *sensor perception reliabilities* without reference truth, we additionally consider the dependence model of Section 6.1.4. In this example, the virtual truth is as in case 2, but with additional critical shocks occurring approximately ten times as often as common error events. The resulting frequency of sensor errors is unrealistically large, but this choice facilitates the illustration of the problem associated with a wrong statistical dependence model.

Under the virtual truth with Eq. (6.16) in the limit as $M \to \infty$, the frequencies in Figure 6.14a) would be observed when conducting a long test with a reference truth. Likewise, the corresponding frequencies of the observations $Z = z$ illustrated in Figure 6.14b) would be observed in a long test without a reference truth. If one now selects the beta-binomial model, Eqs. (4.26)-(4.27), to describe the statistical sensor error process and one learns the parameters (p_{av}, ρ) from the observations Z without a reference truth in Figure 6.14b), one learns the sensor error frequencies illustrated in Figure 6.14c).

Comparing Figure 6.14a) and Figure 6.14c) reveals that the probability of $K = 7$ sensor errors would be underestimated with the beta-binomial model. This is due to the additional critical shocks occurring with $p_s = 0.1$ that are not represented in the beta-binomial model and the fact that $k = 7$ and $k = 0$ cannot be distinguished without reference truth, as already explained in Section 6.1.4.

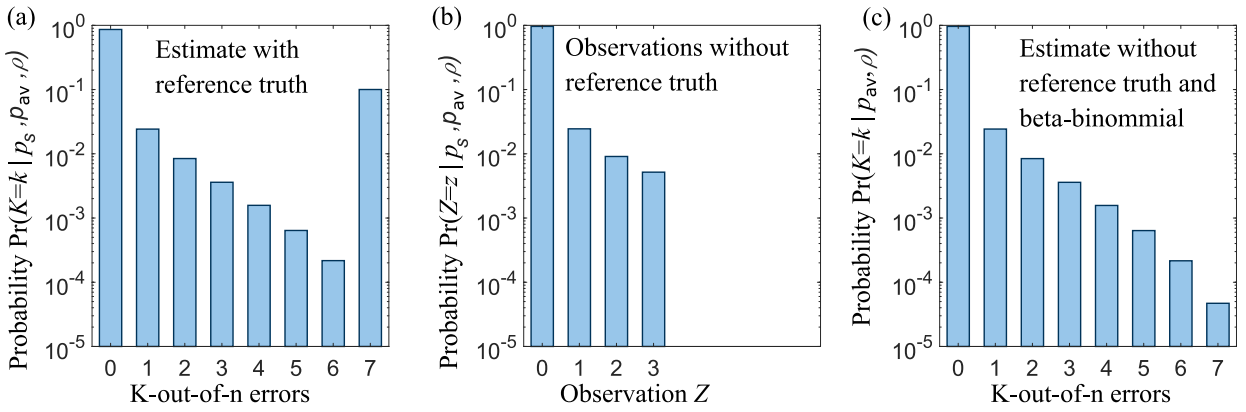

Figure 6.14 (a) Probability of $K = k$ sensors to make an error with virtual truth case 4 according to Eq. (6.16). These frequencies would be obtained in a long test with reference truth. (b) Corresponding frequencies of the observations $Z = z$ in a test without reference truth. (c) Resulting frequencies of sensor errors if one learns the model parameters of the beta-binomial distribution (without critical shocks), Eqs. (4.26)-(4.27), from the observations in (b). Taken and adapted from [156].

It might be suspected that the probability p_s of the critical shocks cannot in general be learned without reference truth, as any number of the observations in which all of the sensors agree on the output, $z = 0$, could in reality be $k = 7$. However, when applying the correct dependence model after Eq. (6.16) to the framework outlined in Section 6.1.2, even the probability of the shocks p_s can be learned without reference truth.

To demonstrate this, the model parameters p_s, p_{av} and ρ are learned from the expected virtual truth (similar to case 3) with $M = 10^9$. The software OpenBUGS based on MCMC sampling is utilized to obtain samples from the joint posterior distribution $f(p_s, p_{av}, \rho|\mathbf{z})$ [269]. The resulting marginal posterior distributions of the model parameters are illustrated in Figure 6.15.

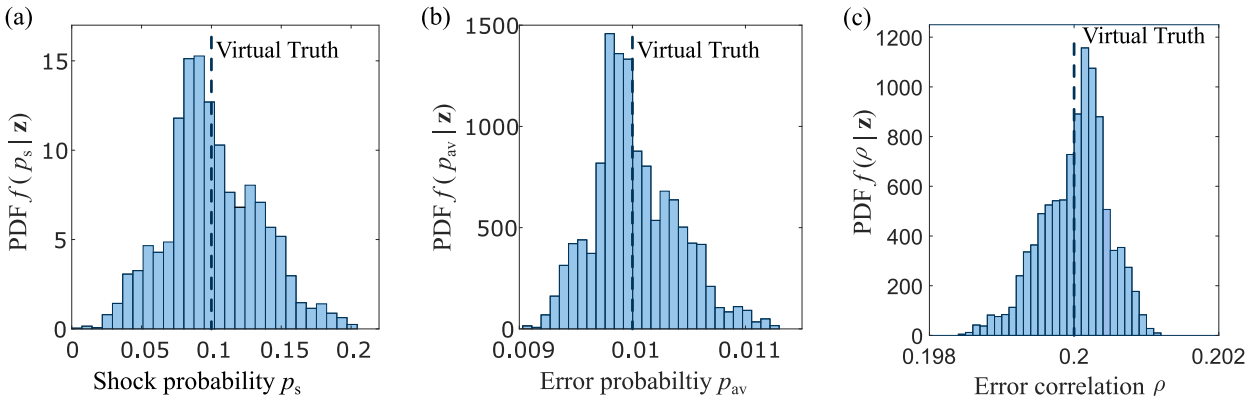

Figure 6.15 (a), (b), (c) Marginal posterior distributions of the model parameters p_s, p_{av} and ρ. Even without reference truth, all model parameters including the shock probability p_s converge to their virtual truth of $p_s = 0.1$, $p_{av} = 0.01$, $\rho = 0.2$. Taken and adapted from [156].

Figure 6.15a) illustrates that the posterior PDF $f(p_s|\mathbf{z})$ is converging. The posterior mean is $\hat{p}_s = 0.101$ and the 95% credible interval $[p_{s_{low}} = 0.039, p_{s_{up}} = 0.174]$, meaning that there is still some uncertainty associated with the estimate. The posterior mean is $\hat{p}_s \approx p_s$ also because the data is equal to the expected values, however, it still can be concluded that in the long term convergence is to be expected. The average sensor error probability p_{av} and the sensor error correlation coefficient ρ are also correctly identified as is apparent from Figure 6.15b) and Figure 6.15c).

It is pointed out, to learn without a reference truth that all sensors make a (shock event type) error once in ten measurement cycles, a large number of observations are necessary (here $M = 10^9$ is used). Additionally, the model for the (common, non-shock event type) errors needs to be perfect, which is however not realistic in practice.

6.1.6 Discussion and Conclusion

The aim of this Section is to investigate on a theoretical basis, if it is possible to learn *sensor perception reliabilities* without a reference truth. Such an approach would have several advantages: First, the significant effort associated with setting up a reference truth is not required. Second, technical difficulties, which make it hard to set up the reference truth in real driving environments, would be overcome. Third, it is an opportunity to parallelize the testing to a large extent and to generate the large amount of data required to achieve credibility in strict target levels of safety. The large amount of data could for instance be collected by a fleet of end-user vehicles equipped with the required sensing hardware, without having automated driving functionalities activated. The data preprocessing could be done online in the vehicles, and the *sensor perception reliabilities* could be demonstrated offline with the combined data from all vehicles in the fleet. With this approach, the test is naturally representative (see Section 3.1.4).

We showed in synthetic case studies with different virtual truth scenarios that when one employs an adequate statistical model for sensor errors, the correct sensor error rates and dependencies can potentially be learned without a reference truth by exploiting sensor redundancy. From these error rates and dependencies, the target level of safety of the perception module can be demonstrated e.g. with the k-out-of-n model assumption or another suitable representation of sensor data fusion. Hence we conclude on a theoretical basis that the concept is applicable, but a number of challenges and simplifications must be addressed before implementing it in practice.

The Challenge of Selecting an Adequate Sensor Error and Dependence Model

Learning *sensor perception reliability* with the presented framework is enabled by Eqs. (6.2)-(6.4). The expression of the likelihood in Eq. (6.4) is central to the approach and can be combined with any suitable statistical model. This means, the learning framework is flexible with respect to the selection of a statistical dependence model. The number of redundant sensors n however constrain the number of free parameters in the statistical model that can be learned without reference truth, which we further discuss in Section 6.2.3.

The main challenge is that the selection of an inadequate statistical model can lead to spurious inference and statistically biased estimates of the sensor error rates, as exemplified in Figure 6.14. That particular case reflects the most critical situation: that the model does not adequately capture the possibility of all sensors to make an error simultaneously. The actual limitation of working without a reference truth is not the error in the parameter estimation, but the difficulty in detecting from data without reference truth that an inadequate statistical model has been selected. The learned beta-binomial model in Figure 6.14c) almost perfectly explains the observations of Figure 6.14b), which in reality are due to the virtual truth in Figure 6.14a). Therefore, one has to make sure the utilized statistical model is robust with respect to the dependence of sensor errors in the different sensors. A model in analogy to Eq. (6.16) conceptually seems to be adequate because it accounts for common failure events and critical shocks. However, considering the large test effort required to learn the shock frequency (see case 4), learning a shock model purely from data without reference truth might not be practicable.

Future research should therefore investigate on the basis of data which statistical models are adequately representing sensor errors and dependencies to enable the learning of *sensor perception reliabilities* without reference truth and to demonstrate the practical feasibility of the approach. Additionally, the potential underestimation of all sensors making an error simultaneously should in practice be precluded with other test methods such as virtual simulations, tests on proving grounds and limited field tests with reference truth.

Simplifications and Future Work

To enhance the clarity of the presentation and for mathematical simplicity, several simplifying assumptions have been made. For instance, we assumed exchangeability in Eq. (6.3) and did not model a potential statistical dependence of sensor errors over subsequent points in time. Future research could try to address both, the dependence among sensors and the dependence over time, in one model.

We further assume that all sensors are identical, but in ADSs, lidar, radar and camera sensors are employed (see Section 2.1). These sensors exploit different physical measurement principles, e.g.

by actively probing the environment with radio waves (radar), coherent light (laser) or by passively capturing environment information in the light spectrum (camera). The different sensors are complementary in their measurement principles, which purposely reduces the sensor error dependence. On the one hand, different physical measurement principles should therefore partly mitigate the discussed problem of shock events and dependencies in the extremes. On the other hand, it is obvious that these sensors have different error rates and interdependencies.

Moreover, this Section did not distinguish between FP and FN sensor errors. In practice, these must be separated, as the mechanisms leading to these errors and their consequences are different [289].

Conclusions

We demonstrated that it is theoretically possible to learn *sensor perception reliabilities* and dependencies without reference truth, when utilizing an adequate statistical model for sensor errors. When not utilizing an adequate statistical model, the reliability estimates might be wrong. On its own, this is not crucial, wrong inference occurs in any statistical analysis with inadequate statistical models. However, without a reference truth, one cannot detect that the statistical model is inadequate. In contrast, with full knowledge of the reference truth, the data allows to identify whether the employed statistical model is inappropriate. Thus, one has to ensure that the selected statistical model is adequate when estimating *sensor perception reliabilities* without reference truth.

6.2 Exploiting Sensor Redundancy for Learning False Positive and False Negative Error Rates

The framework in Section 6.1 is restricted to the situation that all sensors are identical, and it does not distinguish between different error types, e.g. between false positive (FP) and false negative (FN) sensor errors. Here, we overcome these limitations.

We focus on the sensors' task of detecting objects, i.e. existence uncertainty (Section 4.1.1). In Section 6.2.1 we formulate the problem of jointly learning FP and FN error rates of non-identical sensors without reference truth, solely by exploiting sensor redundancy. In Section 6.2.2 we develop the likelihood function to estimate FP and FN sensor error rates, which probabilistically accounts for the missing reference truth. This is in contrast to the generation of a reference truth by automatic label generation algorithms or through sensor data fusion, which typically make a deterministic decision on the reference truth (e.g. on object existence) that can be wrong.

Due to the unknown state of the reference truth, which enters the likelihood as a latent variable, the resulting likelihood function is a mixture of two categorical distributions [115]. The likelihood function enables to estimate average sensor POD and PFA as well as statistical error dependencies with Bayesian inference [114] and more generally machine learning [115, 119] from data without a reference truth. The framework can be applied with any adequate dependence model.

Thereafter, in Section 6.2.3, we discuss difficulties and solution strategies in evaluating the posterior distribution when learning without reference truth. We demonstrate in Section 6.2.4 with synthetic binary (sensor) data, that under adequate dependence models, the framework correctly estimates the POD and PFA as well as sensor error dependencies despite the missing reference truth. In numerical examples, we also investigate the effect of an incorrect dependence model on the accuracy of predictions, which in analogy to the discussion of Section 6.1.6 is the main challenge to the approach.

6.2.1 Problem Formulation: Learning Sensor Perception Reliability in the Existence Uncertainty Domain without Reference Truth

The idea is to learn *sensor perception reliabilities* from a probabilistic reference truth provided by sensor redundancy. The framework is therefore restricted to redundant sensors with overlapping field of view (FOV). We focus in this Section on the sensors' indication of objects, i.e. on existence uncertainties. Other types of sensor information such as object velocities or object classes are not considered. In accordance with Section 4.1.1, we further restrict the presentation here to binary object detections. The binary object detection, e.g. in a certain limited area of the field of view, can be facilitated by an object association derived from sensor data fusion [57–59, 110] based on the individual sensors' objects to track associations (see Section 2.2.2). Such a binary representation is applicable to static and dynamic objects. The detailed specification of a binary representation is not the scope of this Section. We assume the sensors' task of indicating objects is reduced to a binary description.

To learn the sensors' perception reliabilities in the existence uncertainty domain, we apply the formulation of the problem in Section 5.4.1. Let the number of redundant sensors be n. The sample space of the sensors' object indications consists of the 2^n possible outcomes of the detection vector $\mathbf{D} = [D_1, \dots, D_n]$ at points in time m. The detection vectors are identified by $Y = y$, for instance as defined in Table 5.3. The random variable O indicates the existence of an object in reality. The task is to learn $\Pr(Y = y|O = 1)$ and $\Pr(Y = y|O = 0)$ for any value y.

The model of Figure 5.17 forms the basis for the estimation of the parameters $\boldsymbol{\theta}$. The difficulty of learning the model parameters $\boldsymbol{\theta}$ lies in the fact that O cannot be observed without reference truth. This means, no labeled data pairs $[\mathrm{y}_m, o_m]$ are available for the inference of $\boldsymbol{\theta}$. In this case, one

has to consider the uncertainty in O_m for each m. To this end, O_m is included in the model as a latent variable, whose true state is unknown. It is described by its a priori probability $\Pr(O = 1) = p_{\text{obj}}$, which is unknown and should be learned from the data. The estimation of $\boldsymbol{\theta}$ and p_{obj} from data $\mathbf{y} = [\mathrm{y}_m]_{m=1}^{M}$ without reference truth is described next.

6.2.2 Learning Sensor Perception Reliability without a Reference Truth

Without reference truth, the state of O_m is unknown and inference based on Eqs. (5.29)-(5.30) is not possible. In this case, one has to consider the uncertainty in the state of O_m for each observation m. To this end, O_m is included as a latent variable in the model, following Figure 5.17. The likelihood function is formulated by applying the total probability theorem with respect to O_m. The resulting likelihood function is a mixture of two categorical distributions:

$$\begin{aligned} &p_{Y|\boldsymbol{\theta},p_{\text{obj}}}(y_m \,|\boldsymbol{\theta}, p_{\text{obj}}) = \\ &= p_{\text{obj}} \cdot p_{Y|\boldsymbol{\theta},O}(y_m \,|\boldsymbol{\theta}, O_m = 1) + (1 - p_{\text{obj}}) \cdot p_{Y|\boldsymbol{\theta},O}(y_m \,|\boldsymbol{\theta}, O_m = 0) \end{aligned} \tag{6.18}$$

Eq. (6.18) is the likelihood of the model parameters $\boldsymbol{\theta}$ and p_{obj}, given a single observation y_m without reference truth.

The posterior $f(\boldsymbol{\theta}, p_{\text{obj}}|\mathbf{y})$ of the model parameters $\boldsymbol{\theta}$ and p_{obj}, given all the observations $\mathbf{y} = [y_m]_{m=1}^{M}$ without reference truth, is:

$$f(\boldsymbol{\theta}, p_{\text{obj}}|\mathbf{y}) \propto f(\boldsymbol{\theta}, p_{\text{obj}}) \cdot \prod_{m=1}^{M} p_{Y|\boldsymbol{\theta},p_{\text{obj}}}(y_m \,|\boldsymbol{\theta}, p_{\text{obj}}) \tag{6.19}$$

$f(\boldsymbol{\theta}, p_{\text{obj}})$ is the joint prior and $p_{Y|\boldsymbol{\theta},p_{\text{obj}}}(y_m \,|\boldsymbol{\theta}, p_{\text{obj}})$ is the likelihood of a single observation in Eq. (6.18).

Eq. (6.19) assumes exchangeability, i.e. no information is conveyed by the ordering of $[y_m]_{m=1}^{M}$. Strategies to relax the assumption of exchangeability are discussed in Section 4.2.4.

In the general case, $p_{Y|\boldsymbol{\theta},p_{\text{obj}}}(y_m \,|\boldsymbol{\theta}, p_{\text{obj}})$ has the multinomial distribution. As part of the exponential family, multinomial data can be summarized with its sufficient statistics [115] which are the number of occurrence $\mathbf{n}_y = [n_{y=1}, \dots, n_{y=2^n}]$ of each value y in $\mathbf{y}$. With this notation, the log posterior corresponding to Eq. (6.19) is:

$$\log f(\boldsymbol{\theta}, p_{\text{obj}}|\mathbf{n}_y) \propto \log f(\boldsymbol{\theta}, p_{\text{obj}}) + \sum_{y=1}^{2^n} n_y \cdot \log p_{Y|\boldsymbol{\theta},p_{\text{obj}}}(y_m \,|\boldsymbol{\theta}, p_{\text{obj}}) \tag{6.20}$$

$p_{Y|\boldsymbol{\theta},O}(y_m \,|\boldsymbol{\theta}, O_m = o)$ in Eq. (6.18) has to be specified to apply the learning framework. When modeling $p_{Y|\boldsymbol{\theta},O}(y_m \,|\boldsymbol{\theta}, O_m = o)$ with the multinomial distribution, the number of free parameters

in $\boldsymbol{\theta}$ that need to be learned is $2^{n+1}-2$, as pointed out in Section 5.4. In Section 6.2.3, it is discussed that a number of $2^{n+1}-2$ free model parameters cannot be learned without reference truth. Less general alternatives that depending on the number of sensors can be learned without reference truth are the model under independence (Section 5.4.3) and the Gaussian copula dependence model (Section 5.4.4), which have $2n$ and n^2+n free model parameters, respectively. The low rank approximation to the Gaussian copula in Section 5.4.5 requires to learn $4n$ free model parameters. Strategies to evaluate the posterior in Eq. (6.19) or Eq. (6.20), related challenges and requirements on the number of free model parameters are discussed next.

6.2.3 Evaluating the Posterior Distribution without Reference Truth: Challenges and Solution Strategies

The Label-Switching Problem

One problem with inference involving mixtures such as Eq. (6.18) is that the model parameters are not uniquely identified due to the unknown state of the latent variables O_m, which results in the likelihood having multiple modes [115]. This behaviour is known as the label-switching problem because the likelihood is invariant to relabeling the mixture components [294]. The potential number of modes in the likelihood is equal to the number of possible permutations of the mixture components [115, 294], here two.

This is illustrated with the example of $n=3$ sensors and assuming statistical independence between sensor errors (as in Section 5.4.3). In reference to Table 5.3, consider the case that the observations consist 20% of the time of $y=1$ and 80% of $y=8$. Then, according to Eqs. (5.31)-(5.32) the following two explanations are equally likely: Either $p_{\text{obj}}=0.8$, the POD of all sensors is one and the PFA of all sensors is zero, or $p_{\text{obj}}=0.2$, the PFA of all sensors is one and the POD of all sensors is zero. This means, the labels of POD and PFA can be switched.

We solve this problem by regulating the corresponding parameters by a suitable prior $f\left(\boldsymbol{\theta}, p_{\text{obj}}\right)$. As the sensors are being considered for a safety relevant task, at the time of the reliability assessment they have already been tested considerably during development. A priori, POD should be close to one and PFA close to zero. Therefore, one can safely construct a prior which excludes $\text{POD} < 0.5$ and $\text{PFA} > 0.5$. In the illustrative example above this would lead to $p_{\text{obj}}=0.8$, $\text{POD}=1$ and $\text{PFA}=0$ for all sensors.

With this prior, the full posterior distribution of the parameters in Eq. (6.20) can be identified with a suitable MCMC algorithm [114, 115, 266, 267]. Alternatively, the maximum a posteriori parameter (MAP) estimate can be found with standard optimization algorithms [115] or the expectation maximization (EM) algorithm [119, 295].

Selecting a Prior Distribution

In Eq. (6.19) it is required to select a prior distribution $f(\boldsymbol{\theta}, p_{\mathrm{obj}})$ for the model parameters. Recall that the main idea behind the proposed framework is to learn *sensor perception reliabilities* from a large number of driver controlled vehicles, which leads to a large amount of data. It is well known that the influence of the prior on the posterior reduces with increasing amount of data, if the number of model parameters is finite [114]. Hence, in the limit of $M \to \infty$, the choice of the prior becomes irrelevant. The derivation of a prior distribution is therefore not in the focus of this thesis. In case no prior information is available, the prior can be selected to be weakly informative with Jeffreys prior [115, 243] or alternatively, one can simply select a uniform prior for the model parameters. The only information that should be encoded in the prior is the restriction of the support due to the label switching problem.

Markov Chain Monte Carlo (MCMC)

As no closed form analytical solutions can be obtained for the posterior distributions, MCMC is applied to evaluate the posterior [114, 115, 266, 267]. An expectation maximization (EM) algorithm under the assumption of independence between sensor errors (Section 5.4.3) allows us to find accurate maximum a posteriori (MAP) estimates. These are utilized as MCMC starting points for POD_i and PFA_i in the sensors $i = 1, \dots, n$ [119].

While common MCMC algorithms work well with the posterior in the independent case (Section 5.4.3), inference with a multivariate probit model (i.e. a Gaussian copula for correlated binary data, Section 5.4.4) is known to be difficult and computationally demanding [273]. Specialized MCMC algorithms have been developed for probit models [273], these are here however not applicable because our likelihood when inserting Eqs. (5.37)-(5.40) into Eqs. (6.18)-(6.20) is a mixture of two probit models. In particular, the model parameters $\boldsymbol{\theta}_{\mathrm{dep}}$ can exhibit strong dependence. This motivates the use of (gradient-based) Hamiltonian MCMC [114, 296] to improve convergence.

With the combination of EM for finding initial MAP estimates and the Hamiltonian MCMC, convergence can be achieved for the models presented in Sections 5.4.3 and 5.4.5, as we demonstrate in Section 6.2.4.

Required Number of Sensors without Reference Truth

When evaluating *sensor perception reliability* with reference truth, a single sensor can be analyzed individually. In contrast, the proposed approach without reference truth relies on redundant sensors identifying the same objects. In this Section, we briefly discuss the minimum number of sensors necessary to apply the proposed learning framework. To this end, we interpret the inference of the model parameters $\boldsymbol{\theta}$ and p_{obj} as a deterministic inverse problem.

We consider the limiting case of $M \to \infty$ observations, in which the probabilities $p_y,\ y = 1, \dots, 2^n$ are known deterministically. According to the models of Section 5.4, the probabilities p_y are a function of the parameters $\boldsymbol{\theta}$ and p_{obj}, i.e., $p_y = g_y(\boldsymbol{\theta}, p_{\mathrm{obj}})$. With this interpretation of the problem, $p_y = g_y(\boldsymbol{\theta}, p_{\mathrm{obj}})$ for $y = 1, \dots, 2^n$ defines a nonlinear system of $2^n - 1$ equations (one equation is redundant because of the constraint $\sum_{y=1}^{y=2^n} p_y = 1$). This would allow obtaining $\boldsymbol{\theta}$ and p_{obj} from $\mathbf{p}_{\mathrm{y}} = [p_1, \dots, p_{2^n}]$ by the solution of the inverse problem $[\boldsymbol{\theta}, p_{\mathrm{obj}}] = g_y^{-1}(\mathbf{p}_y)$. Note that the constraints POD < 0.5 and PFA > 0.5 following the label switching problem must be included to obtain a unique solution to this inverse problem.

With this interpretation of the problem, we conclude that $\boldsymbol{\theta}$ can consist of up to $2^n - 2$ free parameters (one free parameter is p_{obj}). Otherwise, there is no unique solution for $\boldsymbol{\theta}$ in the inverse problem, which leads to likelihood invariance under different parameter combinations. The model with independent sensors (Section 5.4.3) has $2n$ free parameters. It follows that $n \geq 3$ sensors are required to learn this model. For dependent sensors, the model of Section 5.4.4 has $n^2 + n$ parameters and the model of Section 5.4.5 $4n$, excluding p_{obj}. Hence, both proposed models for dependent sensors require $n \geq 5$ sensors.

6.2.4 Numerical Examples

A series of synthetic numerical experiments demonstrate the proposed methodology. Several synthetic data sets $\mathbf{y} = [y_m]_{m=1}^{M}$ containing M samples of the detection vectors are generated with fixed sensor reliabilities and dependencies $\boldsymbol{\theta}$ and p_{obj}. In case of independence, the data set $\mathbf{y}$ is sampled based on Eqs. (5.31)-(5.32) inserted into Eq. (6.18) and in case of dependence, based on Eqs. (5.37)-(5.40) inserted into Eq. (6.18). From the samples $\mathbf{y} = [y_m]_{m=1}^{M}$, the sufficient test statistics $\mathbf{n}_y$ are derived (see Section 6.2.2).

As outlined in Section 6.2.3, the minimum number of sensors to learn the parameters is $n = 3$ in case of independence and $n = 5$ in case of dependence, which we adapt in the following. Four different numerical investigations are performed: (1) We estimate $\boldsymbol{\theta}_{\mathrm{indep}}$ under the assumption of independence (following Sections 5.4.3 and 6.2.2) from statistically independent data. (2) We

estimate $\boldsymbol{\theta}_{\text{indep}}$ under the assumption of independence (following Sections 5.4.3 and 6.2.2) from data generated with a model including statistical dependence, to investigate the effect of wrong model assumptions. (3) We estimate $\boldsymbol{\theta}_{\text{dep}}$ under the assumption of dependence (following Sections 5.4.5 and 6.2.2) with data whose dependence is described by correlation matrices of the Dunnet-Sobel class. (4) We estimate $\boldsymbol{\theta}_{\text{dep}}$ under the assumption of dependence (following Sections 5.4.5 and 6.2.2) where the error dependence structure used to generate the data does not follow the Dunnet-Sobel class assumed in the estimation. Therefore, in case studies 1) and 3), the models are aligned to the data, and in case studies 2.) and 4.), the data does not follow the selected models. The synthetic data does not reproduce real data but allows to study the properties of the proposed inference algorithms.

The number of sensors n, the selected probabilities of detection POD_i, the probabilities of false alarm PFA_i and the number of synthetic observations M for case studies 1-4 are summarized in Table 6.2. $p_{\text{obj}} = 0.8$ in all case studies.

Table 6.2 Case study: virtual truth of the probabilities of detection POD_i and probabilities of false alarm PFA_i. From [157].

Case	Number of sensors n	$\text{POD}_1, \dots, \text{POD}_n$	$\text{PFA}_1, \dots, \text{PFA}_n$ [x10^{-3}]	Number of data points M
1)	3	[0.999, 0.9999, 0.99999]	[2, 0.2, 0.02]	10^7
2)	3	[0.999, 0.9999, 0.99999]	[2, 0.2, 0.02]	10^7
3)	5	[0.995,0.999,0.9995,0.9999,0.99995]	[25,20,2.5,2.0,1.0]	10^8
4)	5	[0.995,0.999,0.9995,0.9999,0.99995]	[25,20,2.5,2.0,1.0]	10^8

The correlation matrices $\mathbf{R}_{\mathbf{U},\text{FN}}$ and $\mathbf{R}_{\mathbf{U},\text{FP}}$ of FN and FP sensor errors in standard normal space for case studies (2-4) are summarized in Table 6.3. In case study 2, the detection vectors are sampled based on $\boldsymbol{\theta}_{\text{dep}}$, therefore the virtual truth includes the correlation matrices. The elements of the correlation matrices of case (2) are randomly selected between zero and one. The correlation matrices of case study (3) belong to the Dunnet-Sobel class (Section 5.4.5) with randomly selected underlying values $\boldsymbol{\lambda}_{\text{U}_{\text{FN}}} = [0.50, 0.61, 0.00, 0.58, 0.78]$ and $\boldsymbol{\lambda}_{\text{U}_{\text{FP}}} = [0.66, 0.53, 0.83, 0.96, 0.31]$. The values of $\mathbf{R}_{\mathbf{U},\text{FN}}$ and $\mathbf{R}_{\mathbf{U},\text{FP}}$ in case study (4) are randomly selected between zero and 0.7.

Table 6.3 Virtual truth correlation matrices $\mathbf{R}_{U,FN}$ and $\mathbf{R}_{U,FP}$ of FN and FP sensor errors. From [157].

Case	$\mathbf{R}_{U,FN}$	$\mathbf{R}_{U,FP}$
2)	$\begin{pmatrix} 1 & 0.71 & 0.72 \\ 0.71 & 1 & 0.91 \\ 0.72 & 0.91 & 1 \end{pmatrix}$	$\begin{pmatrix} 1 & 0.66 & 0.43 \\ 0.66 & 1 & 0.42 \\ 0.43 & 0.42 & 1 \end{pmatrix}$
3)	$\begin{pmatrix} 1 & 0.30 & 0 & 0.29 & 0.39 \\ 0.30 & 1 & 0 & 0.36 & 0.47 \\ 0 & 0 & 1 & 0 & 0 \\ 0.29 & 0.36 & 0 & 1 & 0.46 \\ 0.39 & 0.47 & 0 & 0.46 & 1 \end{pmatrix}$	$\begin{pmatrix} 1 & 0.34 & 0.54 & 0.63 & 0.21 \\ 0.34 & 1 & 0.44 & 0.51 & 0.17 \\ 0.54 & 0.44 & 1 & 0.79 & 0.26 \\ 0.63 & 0.51 & 0.79 & 1 & 0.30 \\ 0.21 & 0.17 & 0.26 & 0.30 & 1 \end{pmatrix}$
4)	$\begin{pmatrix} 1 & 0.61 & 0.67 & 0.64 & 0.53 \\ 0.61 & 1 & 0.20 & 0.23 & 0.33 \\ 0.67 & 0.20 & 1 & 0.40 & 0.42 \\ 0.64 & 0.23 & 0.40 & 1 & 0.17 \\ 0.53 & 0.33 & 0.42 & 0.17 & 1 \end{pmatrix}$	$\begin{pmatrix} 1 & 0.40 & 0.26 & 0.19 & 0.69 \\ 0.40 & 1 & 0.45 & 0.27 & 0.08 \\ 0.26 & 0.45 & 1 & 0.22 & 0.03 \\ 0.19 & 0.27 & 0.22 & 1 & 0.27 \\ 0.69 & 0.08 & 0.03 & 0.27 & 1 \end{pmatrix}$

For reproducibility, the randomly generated sufficient statistics $\mathbf{n}_y$ of the number of observations with $Y = y$ (as in Table 5.3) are summarized in Table 6.4 for case studies (1-2).

Table 6.4 Case studies (1-2): Number $\mathbf{n}_y$ of synthetic observations with $Y = y$. From [157].

Case	$n_{y=1}$	$n_{y=2}$	$n_{y=3}$	$n_{y=4}$	$n_{y=5}$	$n_{y=6}$	$n_{y=7}$	$n_{y=8}$
1)	1996720	4043	395	37	89	8038	801	7989877
2)	1994624	4131	284	302	127	7769	493	7992270

$\mathbf{n}_y$ for case studies (3-4) is given in Table 6.5.

Table 6.5 Corresponding detection vectors D=[$D_1, \dots, D_5$] and observations $Y = y$ in case of $n = 5$ sensors together with the number of synthetic observations n_y in case studies (3) and (4).

Variable Y	D_1	D_2	D_3	D_4	D_5	Case 3) n_y	Case 4) n_y
1	0	0	0	0	0	19088692	19077268
2	1	0	0	0	0	428716	423982
3	0	1	0	0	0	342185	330087
4	0	0	1	0	0	21789	34777
5	0	0	0	1	0	9950	32831
6	0	0	0	0	1	16689	6123
7	1	1	0	0	0	37953	53385
8	1	0	1	0	0	8669	3193
9	1	0	0	1	0	9052	2112
10	1	0	0	0	1	1326	12703
11	0	1	1	0	0	4240	8932
12	0	1	0	1	0	3705	3264
13	0	1	0	0	1	901	305
14	0	0	1	1	0	3615	396
15	0	0	1	0	1	195	88
16	0	0	0	1	1	101	418
17	1	1	1	0	0	2284	2446
18	1	1	0	1	0	3864	857
19	1	1	0	0	1	174	702
20	1	0	1	1	0	4958	74
21	1	0	1	0	1	293	44
22	1	0	0	1	1	161	233
23	0	1	1	1	0	2208	1244
24	0	1	1	0	1	504	3432
25	0	1	0	1	1	250	15537
26	0	0	1	1	1	4043	19636
27	1	1	1	1	0	5832	2743
28	1	1	1	0	1	7144	4310
29	1	1	0	1	1	39732	23755
30	1	0	1	1	1	75500	59802
31	0	1	1	1	1	393521	358226
32	1	1	1	1	1	79481754	79517095

Following Section 6.2.3, we select a uniform prior. To solve the label-switching problem, we require that $f(\boldsymbol{\theta}, p_{\text{obj}}) = 0$ when $\text{POD}_i \leq 0.5$ or $\text{PFA}_i \geq 0.5$ for all $i = 1, \dots, n$.

In all case studies, samples $[\boldsymbol{\theta}, p_{\text{obj}}]$ from the respective posterior distributions $f(\boldsymbol{\theta}, p_{\text{obj}} | \mathbf{n}_y)$ were generated with Hamiltonian MCMC with starting points randomly perturbed around the initial EM estimate, as discussed in Section 6.2.3 [114, 296]. Convergence of the MCMC chains was checked with the Gelman-Rubin convergence statistic [114], which was close to one for all parameters and case studies.

Results: Case (1) Independent Sensors

Based on the observations $\mathbf{n}_y$ of case (1), as in Table 6.4, and the prior as defined under 6.2.4, the log-posterior distribution $\log f\left(\boldsymbol{\theta}_{\text{indep}}, p_{\text{obj}} \middle| \mathbf{n}_y\right)$ is set up with Eq. (6.20) and the model described in Section 5.4.3. Note that it holds $\log f\left(\boldsymbol{\theta}_{\text{indep}}, p_{\text{obj}} \middle| \mathbf{n}_y\right) = \log f\left(\boldsymbol{\theta}_{\text{indep}}, p_{\text{obj}} \middle| \mathbf{y}\right)$ where $\mathbf{y} = [y_m]_{m=1}^{M}$ are the detection vectors underlying $\mathbf{n}_y$.

To illustrate the joint posterior, we exemplarily show 1604 generated samples of selected parameters $[\text{POD}_3, \text{PFA}_3, p_{\text{obj}}]$ in Figure 6.16. POD_3 is the probability of detection in sensor 3 and PFA_3 is the probability of false alarm in sensor 3.

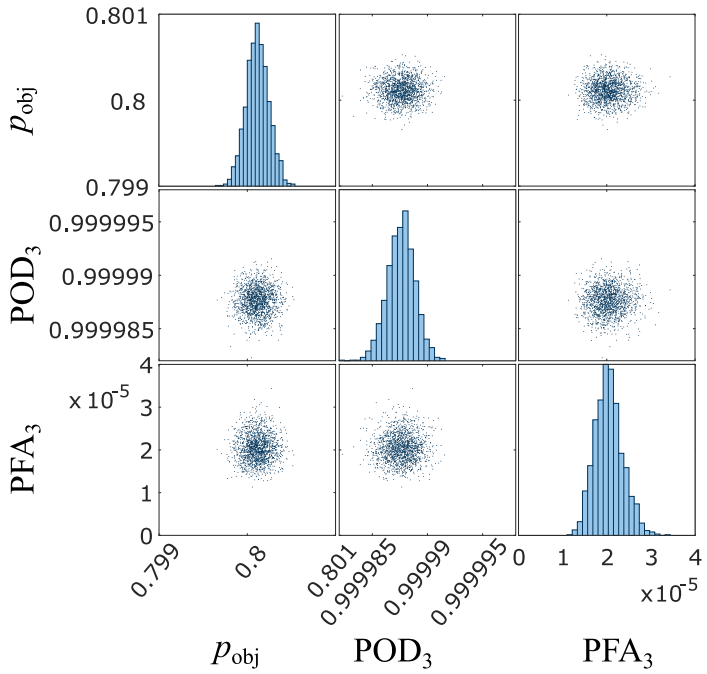

Figure 6.16 Samples of the joint posterior $f\left(\text{POD}_3, \text{PFA}_3, p_{\text{obj}} \middle| \mathbf{y}\right)$ for case (1). Taken and adapted from [157].

The samples spread around the virtual truths of POD_3 and PFA_3 given in Table 6.2 and around the correct probability of an object being present, $p_{\text{obj}} = 0.8$. Therefore, the correct POD_3 and PFA_3 as well as the correct p_{obj} are learned, even though it is unknown if a particular observation in Table 6.4 is actually a TP or FP sensor detection. As also visible in Figure 6.16, the parameters are independent in their posterior distribution, which facilitates convergence of the MCMC chains.

The marginal posterior distributions $f(\text{POD}_i|\mathbf{y})$ and $f(\text{PFA}_i|\mathbf{y})$ of the remaining parameters are presented in Figure 6.17 to demonstrate that also these parameters are correctly learned. The dashed lines indicate the virtual truths. $f(\text{PFA}_2|\mathbf{y})$ is shifted compared to the virtual truth due to sampling uncertainty.

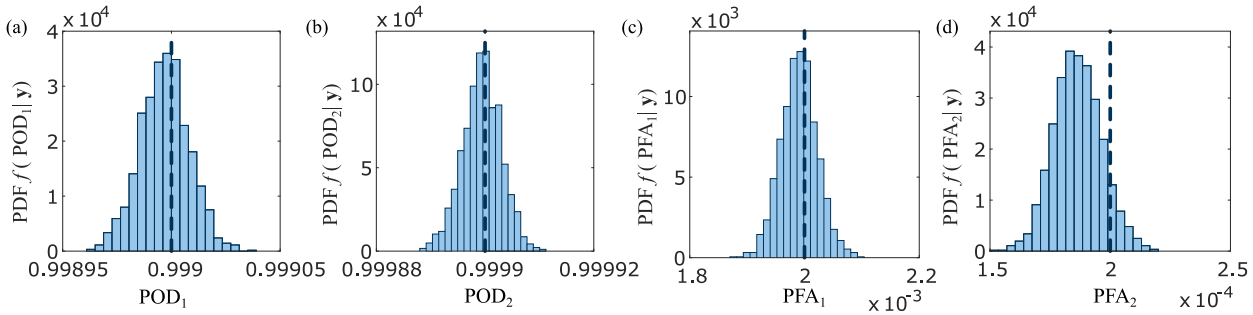

Figure 6.17 Marginal posterior distributions of case (1): (a) $f(\mathbf{POD_1}|\mathbf{y})$ probability of detection in sensor 1 (b) $f(\mathbf{POD_2}|\mathbf{y})$ probability of detection in sensor 2 (c) $f(\mathbf{PFA_1}|\mathbf{y})$ probability of false alarm in sensor 1 (d) $f(\mathbf{PFA_2}|\mathbf{y})$ probability of false alarm in sensor 2. The dashed lines indicate the virtual truth. From [157].

Results: Case (2) Incorrectly Assuming Independence

In case (2), the FP and FN detections underlying the observations $\mathbf{n}_y$ in Table 6.4 include dependence according to the correlation matrices in Table 6.3. Nevertheless, the posterior distribution $f(\boldsymbol{\theta}_{\text{indep}}, p_{\text{obj}}|\mathbf{y})$ is estimated based on the assumption of independence with Eq. (6.20) and the model under independence described in Section 5.4.3, as in case (1).

Figure 6.18 presents the estimated marginal posterior distribution of the model parameters after convergence of the MCMC chains, based on 1604 MCMC samples.

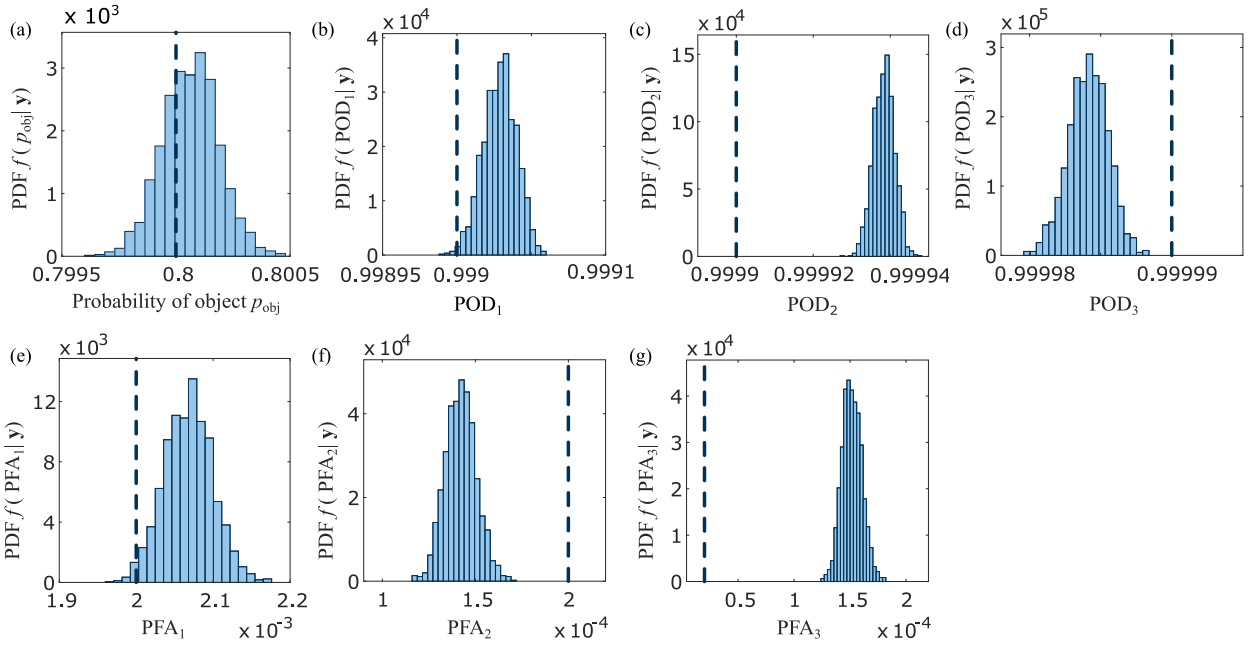

Figure 6.18 Marginal posterior distributions of case (2): (a) $f(p_{\text{obj}}|\mathbf{y})$ probability of object being present; (b-d) $f(\mathbf{POD}_i|\mathbf{y})$ probability of detection in sensor i; (e-f) $f(\mathbf{PFA}_i|\mathbf{y})$ probability of false alarm in sensor i. The dashed lines indicate the virtual truth. Taken and adapted from [157].

Due to the dependence, which is not captured in the model of Section 5.4.3, the parameters are not correctly learned. Nevertheless, the posterior means are approximately of the same order of magnitude as the virtual truths.

Results: Case (3) Dependent Sensors

In case (3), the dependence model is as described in Section 5.4.5. The posterior distribution $f\left(\boldsymbol{\theta}_{\text{dep,DS}}, p_{\text{obj}} \middle| \mathbf{y}\right)$ is based on Eqs. (5.42)-(5.44) inserted into Eq. (6.20). The data $\mathbf{n}_{\text{y}}$ underlying this case study are summarized in Table 6.5. The model parameters are $\boldsymbol{\theta}_{\text{dep,DS}} = [\text{POD}_1, \dots, \text{POD}_5, \text{PFA}_1, \dots, \text{PFA}_5, \lambda_{\text{U}_{\text{FN},1}}, \dots, \lambda_{\text{U}_{\text{FN},5}}, \lambda_{\text{U}_{\text{FP},1}}, \dots, \lambda_{\text{U}_{\text{FP},5}}]$.

Figure 6.19 shows the posterior $f\left(\text{PFA}_3, \text{PFA}_4, \text{PFA}_5, \rho_{\text{U}_{\text{FP},3,4}}, \rho_{\text{U}_{\text{FP},3,5}}, \rho_{\text{U}_{\text{FP},4,5}} \middle| \mathbf{y}\right)$ of selected parameters. The correlation coefficients $\rho_{\text{U}_{\text{FP},i,j}}$ are calculated with Eq. (5.41).

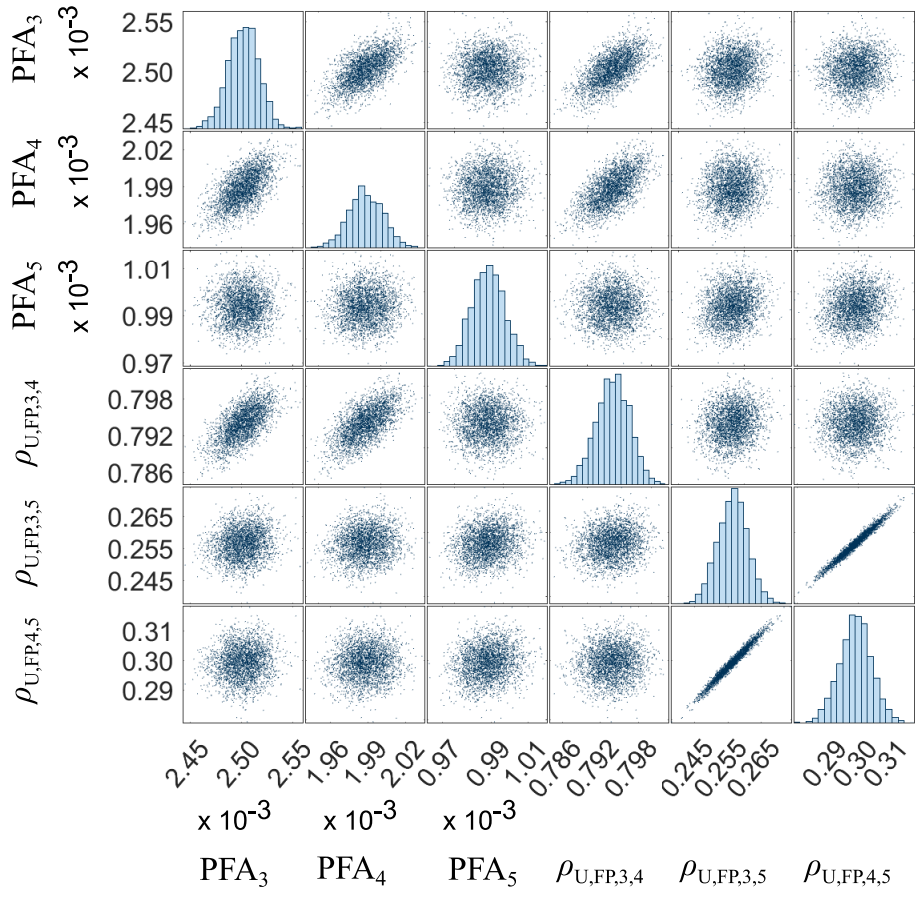

Figure 6.19 The joint posterior $f\left(\text{PFA}_3, \text{PFA}_4, \text{PFA}_5, \rho_{\text{U}_{\text{FN},3,4}}, \rho_{\text{U}_{\text{FP},3,5}}, \rho_{\text{U}_{\text{FP},4,5}} \middle| \mathbf{y}\right)$ of selected parameters for case (3). Taken and adapted from [157].

Unlike in Figure 6.16, not all posterior parameters are independent. In particular, $\rho_{\mathrm{U_{FP,3,5}}}$ and $\rho_{\mathrm{U_{FP,4,5}}}$ show a strong linear dependence. The dependence in the posterior distribution makes MCMC-based inference more challenging, but the applied Hamiltonian MCMC is well suited for such problems because it is gradient based.

The posterior mean estimates of POD_i and PFA_i are summarized in Table 6.6. Similar to Figure 6.17, the remaining posterior uncertainty in POD_i and PFA_i is negligible and is therefore not reported. The posterior mean of p_{obj} is 0.80 with negligible uncertainty. Comparing Table 6.6 with Table 6.2 reveals that all probabilities of detection and probabilities of false alarm are estimated accurately.

Table 6.6 Case study 3): Posterior mean parameter estimates of POD_i and PFA_i. From [157].

Posterior mean $\mathrm{POD}_1, \dots, \mathrm{POD}_5$	Posterior mean $\mathrm{PFA}_1, \dots, \mathrm{PFA}_5 [\cdot 10^{-3}]$
[0.995015, 0.998998, 0.999501, 0.999901, 0.999950]	[25.00, 20.00, 2.50, 1.99, 0.99]

The marginal posteriors $f(\lambda_{\mathrm{U_{FN},}i}|\mathbf{y})$ and $f(\lambda_{\mathrm{U_{FP},}i}|\mathbf{y})$ of the coefficients $\lambda_{\mathrm{U_{FN},}i}$ and $\lambda_{\mathrm{U_{FP},}i}$, which according to Eq. (5.41) fully define the dependence structure in the model of Section 5.4.5, are summarized in Figure 6.20. With the exception of $\lambda_{\mathrm{U_{FN},3}}$, all marginal posterior distributions are centred on their underlying virtual truth. The deviation of the estimate of $\lambda_{\mathrm{U_{FN},3}}$ from the true value is because both values are close to zero. This slight deviation has no noticeable impact on the resulting dependence structure. The deviation of $f(\lambda_{\mathrm{U_{FP},1}}|\mathbf{y})$ from the true value is caused by sampling uncertainty.

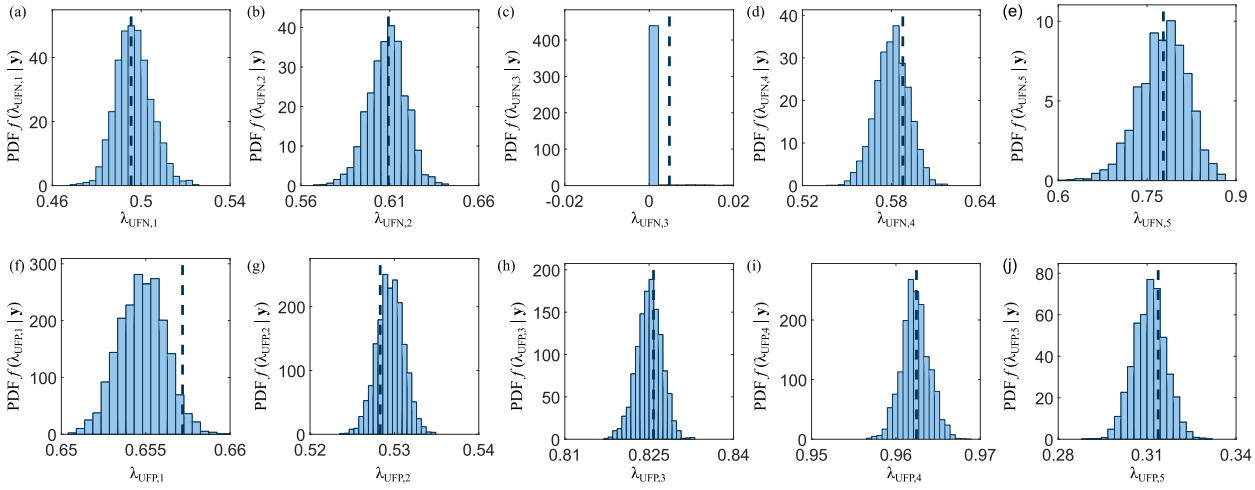

Figure 6.20 The marginal posteriors (a-e) $f(\lambda_{\mathrm{U_{FN},}i}|\mathbf{y})$ and (f-j) $f(\lambda_{\mathrm{U_{FP},}i}|\mathbf{y})$ in sensor i for case (3). The dashed lines indicate the virtual truth. Taken and adapted from [157].

We (jointly) transform all samples in Figure 6.20 into samples of the correlation coefficients of sensor errors in standard normal space $\rho_{U_{FN},i,j}$ and $\rho_{U_{FP},i,j}$ with Eq. (5.41). The matrices containing the resulting posterior means of $\rho_{U_{FN},i,j}$ and $\rho_{U_{FP},i,j}$ are summarized in Table 6.7. These matrices are essentially identical to the underlying truth, presented in Table 6.3. Hence the dependence structure in both FP and FN sensor errors is correctly identified.

Table 6.7 Posterior mean correlation matrices $\widehat{\mathbf{R}}_{\mathbf{U,FN}}$ and $\widehat{\mathbf{R}}_{\mathbf{U,FP}}$ of FN and FP sensor errors in standard normal space for case (3). From [157].

$\widehat{\mathbf{R}}_{\mathrm{U,FN}}$	$\widehat{\mathbf{R}}_{\mathrm{U,FP}}$
$\begin{pmatrix} 1 & 0.30 & 0 & 0.29 & 0.39 \\ 0.30 & 1 & 0 & 0.36 & 0.47 \\ 0 & 0 & 1 & 0 & 0 \\ 0.29 & 0.36 & 0 & 1 & 0.45 \\ 0.39 & 0.47 & 0 & 0.45 & 1 \end{pmatrix}$	$\begin{pmatrix} 1 & 0.35 & 0.54 & 0.63 & 0.20 \\ 0.35 & 1 & 0.44 & 0.51 & 0.17 \\ 0.54 & 0.44 & 1 & 0.79 & 0.27 \\ 0.63 & 0.51 & 0.79 & 1 & 0.30 \\ 0.20 & 0.17 & 0.27 & 0.30 & 1 \end{pmatrix}$

Results: Case (4) Inadequate Dependence Model

In case (4), the virtual truth of the dependence in FN and FP sensor errors does not follow the Dunnet-Sobel class model of Section 5.4.5. Nevertheless, the model is applied to the data of case study (4) in Table 6.5, analogous to case 3.The estimates of the sensors' POD_i and PFA_i (posterior means from 3600 samples) are summarized in Table 6.8. Comparing with Table 6.3 reveals that they are close to their true values.

Table 6.8 Case study (4): Posterior mean parameter estimates of $\mathbf{POD}_i$ and $\mathbf{PFA}_i$. From [157].

Posterior mean $\mathbf{POD_1, \dots, POD_5}$	Posterior mean $\mathbf{PFA_1, \dots, PFA_5 [x10^{-3}]}$
[0.995168, 0.998997, 0.999500, 0.999906, 0.999945]	[25.01, 20.02, 2.52, 1.98, 1.04]

Analogue to Table 6.7, estimates of $\widehat{\mathbf{R}}_{\mathbf{U,FN}}$ and $\widehat{\mathbf{R}}_{\mathbf{U,FP}}$ are derived from the posterior mean correlation coefficients $\hat{\rho}_{U_{FN},i,j}$ and $\hat{\rho}_{U_{FP},i,j}$. The absolute error of these estimates with respect to their underlying virtual truth of $\rho_{U_{FN},i,j}$ and $\rho_{U_{FP},i,j}$ is summarized in Table 6.9. As is apparent, the estimated correlation matrices are partly subject to errors. However, in this specific case the absolute error is small for most correlation coefficients.

Table 6.9 **Absolute error of the posterior mean correlation coefficients $\widehat{\rho}_{\mathrm{U_{FN},\mathit{i},\mathit{j}}}$ and $\widehat{\rho}_{\mathrm{U_{FP},\mathit{i},\mathit{j}}}$ of FN and FP sensor errors in standard normal space w.r.t. the underlying virtual truth of $\rho_{\mathrm{U_{FN},\mathit{i},\mathit{j}}}$ and $\rho_{\mathrm{U_{FP},\mathit{i},\mathit{j}}}$. From [157].**

$(\widehat{\boldsymbol{\rho}}_{\mathbf{U_{FN},\mathit{i},\mathit{j}}} - \boldsymbol{\rho}_{\mathbf{U_{FN},\mathit{i},\mathit{j}}})$					$(\widehat{\boldsymbol{\rho}}_{\mathbf{U_{FP},\mathit{i},\mathit{j}}} - \boldsymbol{\rho}_{\mathbf{U_{FP},\mathit{i},\mathit{j}}})$				
$\begin{pmatrix} 0 \\ 0.00 \\ 0.00 \\ -0.01 \\ 0.04 \end{pmatrix}$	$\begin{matrix} 0.00 \\ 0 \\ 0.21 \\ 0.15 \\ 0.01 \end{matrix}$	$\begin{matrix} 0.00 \\ 0.21 \\ 0 \\ 0.02 \\ -0.04 \end{matrix}$	$\begin{matrix} -0.01 \\ 0.15 \\ 0.02 \\ 0 \\ 0.19 \end{matrix}$	$\begin{matrix} 0.04 \\ 0.01 \\ -0.04 \\ 0.19 \\ 0 \end{matrix}$	$\begin{pmatrix} 0 \\ 0.01 \\ 0.12 \\ 0.05 \\ -0.06 \end{pmatrix}$	$\begin{matrix} 0.01 \\ 0 \\ -0.18 \\ -0.10 \\ 0.37 \end{matrix}$	$\begin{matrix} 0.12 \\ -0.18 \\ 0 \\ -0.06 \\ 0.39 \end{matrix}$	$\begin{matrix} 0.05 \\ -0.10 \\ -0.06 \\ 0 \\ -0.01 \end{matrix}$	$\begin{matrix} -0.06 \\ 0.37 \\ 0.39 \\ -0.01 \\ 0 \end{matrix}$

Of major interest is how the biased estimates of the parameters $\boldsymbol{\theta}_{\mathrm{dep}}$ affect the ability to predict the occurrence frequencies $\Pr(Y = y \,|\boldsymbol{\theta}_{\mathrm{dep}}, O = 1)$ and $\Pr(Y = y \,|\boldsymbol{\theta}_{\mathrm{dep}}, O = 0)$. To evaluate the performance of the prediction, we evaluate the posterior predictive distributions $\Pr(Y = y \,|O = 1, \mathbf{n}_y) = \int_{-\infty}^{\infty} \Pr(Y = y \,|\boldsymbol{\theta}_{\mathrm{dep}}, O = 1) \cdot f(\boldsymbol{\theta}_{\mathrm{dep}}|\mathbf{n}_y)\, d\boldsymbol{\theta}_{\mathrm{dep}}$ and $\Pr(Y = y \,|O = 0, \mathbf{n}_y) = \int_{-\infty}^{\infty} \Pr(Y = y \,|\boldsymbol{\theta}_{\mathrm{dep}}, O = 0) \cdot f(\boldsymbol{\theta}_{\mathrm{dep}}|\mathbf{n}_y)\, d\boldsymbol{\theta}_{\mathrm{dep}}$.

Figure 6.21 illustrates the posterior predictive distributions. Additionally, the virtual truth and estimates of $\Pr(Y = y \,|\boldsymbol{\theta}_{\mathrm{indep,MAP}}, O = 1)$ and $\Pr(Y = y \,|\boldsymbol{\theta}_{\mathrm{indep,MAP}}, O = 0)$ under the assumption of independence are shown. The latter are based on the maximum a posteriori parameter (MAP) estimates $\boldsymbol{\theta}_{\mathrm{indep,MAP}}$ under independence, which have been derived with the implemented expectation maximization (EM) algorithm. Due to the negligible posterior uncertainty, the prediction with the MAP estimates is essentially identical to the corresponding predictive distribution under independence.

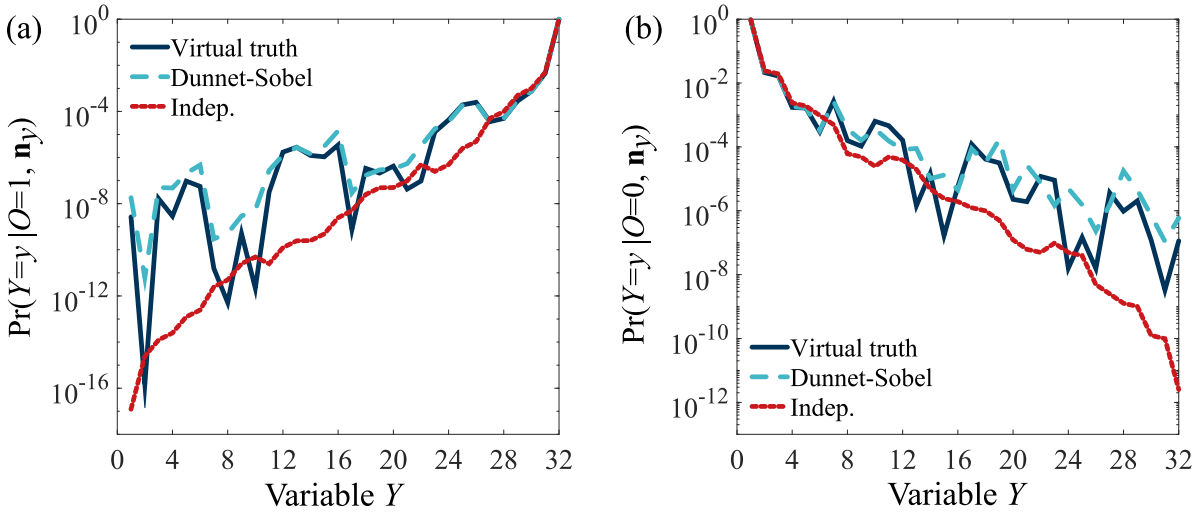

Figure 6.21 **Comparing the predictions of $Y = y|O = o$ based on the assumption of independence (dotted line) and based on the Dunnet-Sobel model (dashed line) with the virtual truth (solid line). (a) $O = 1$ an object is present and (b) $O = 0$ no object is present. Each $Y = y$ defines a different combination of sensor detections $\mathbf{D} = \mathbf{d}$ as defined in Table 6.5. Taken and adapted from [157].**

Despite the errors in estimating the correlation matrices (see Table 6.9), the predictions with the Dunnet-Sobel model are rather accurate. In both cases, Figure 6.21a) with $O = 1$ and Figure 6.21b) $O = 0$, the Dunnet-Sobel model gives better predictions than the assumption of independence. Especially the worst case errors $Y = 1$ in Figure 6.21a) (all sensors commit a FN error) and $Y = 32$ in Figure 6.21b) (all sensors commit FP error) are far more accurately predicted than under the assumption of independence.

6.2.5 Discussion

We developed a statistical framework that allows determining *sensor perception reliability* in the existence uncertainty domain without a reference truth, by exploiting sensor redundancy. In case of independent sensor errors, the approach is applicable for $n \geq 3$, and for dependent sensor errors for $n \geq 5$ redundant sensors. $n = 3$ redundant sensors is common today, and in the future, it is likely that automated driving functionalities (≥level 3) will employ more than 3 redundant sensors. Alternatively, one could also equip a selected number of vehicles with ≥ 5 redundant sensors to apply the approach.

A prerequisite for the presented framework is a binary interpretation of the sensor data. This can be achieved for instance by restricting the analysis of sensor object indications to a limited area of the field of view, by evaluating if a preceding vehicle is detected in a certain range, or by thresholding continuous parameters. Before the framework can be implemented in practice, one must devise a suitable binary representation of the sensor perception.

Another challenge lies in the dependence among sensor errors, which is potentially safety-critical and must be included in the model. In synthetic case studies, we demonstrate that one can correctly learn *sensor perception reliability* of binary sensor data (here in form of FP and FN sensor error rates) and sensor error dependencies without reference truth, if an adequate dependence model is employed. If the dependence model is not accurate, as simulated in cases (2) and (4), the estimated *sensor perception reliabilities* and dependencies are biased. In this Section we implemented a Gaussian copula to describe correlated (binary) sensor errors. To facilitate statistical inference, we used a low rank parameterization of the Gaussian copula dependence model.

It is difficult to decide without reference truth if a correct dependence model is employed. This issue has already been discussed in detail in Section 6.1.6, and equivalently applies to the framework in this Section.

Simplifications and Future Work

We only consider the perception reliability (POD and PFA) averaged over time. The perception reliabilities are changing in time due to variable environmental influencing factors (see Section 3.1.4). Including context variables in the proposed framework as covariates is conceptually straightforward but would require to collect additional data, which might be difficult. This would allow to a) include variability in the model parameters and b) allow to relax the exchangeability assumption made in Eq. (6.19). Future work could extend the framework to account for context variables by formulating the model parameters as functions of covariates, similar to Section 5.3. This could also facilitate an online estimation of sensor perception reliability for an optimization of fusion algorithms.

If the frequencies of the different detection vectors (see e.g. Table 5.3) are sensitive towards context variables, one could also utilize this information to predict which environmental conditions a vehicle is currently experiencing. Further, deviations of the frequency of the different detection vectors from their global mean could indicate that a specific sensor is not working correctly. This is an important information for sensor data fusion. In practical applications of testing sensors without a reference truth, it might also be beneficial to enhance the learning with additional information. This could for instance be the braking signal and the velocity of the ego-vehicle.

Note that every time the (binary) output of the sensors is not in line, one knows with certainty that a perception error must have occurred in one of the sensors. One could identify these events and trigger a manual data labeling process only for these events. This would allow to clarify which sensor made an error and speed up the learning. Following this approach, if the test is long enough, one does not even need any statistics to learn the correct sensor error frequencies (except for the case that all sensors make an error, as these events cannot be identified). With this partial reference truth (for some points in time one knows the error occurrence with certainty and for some not) one could combine the learning methods outlined in this Section with Section 5.4.2.

Another avenue of future research could be to extend the binary case of sensor errors (object present or not) to the multivariate problem of object classification (e.g. with classes: no object present, car, pedestrian, bicycle, motorcycle, truck, etc.). Further, one could also try to probabilistically combine the proposed framework with sensor data fusion algorithms, e.g. JIPDA [118].

Finally and most importantly, as already discussed in Section 6.1.6, additional investigations with real data into adequate models of sensor error dependence are required before the framework can be applied in practice. This could enable an unbiased estimation of sensor error rates without reference truth. Moreover, it is in any case advisable to conduct limited tests with a reference truth to preclude a large frequency of joint sensor errors.

Opportunities

With an adequate statistical model, the proposed framework is an opportunity to demonstrate *perception reliability* with a fleet of end-user vehicles, as sketched in Figure 6.22. With this approach, a fleet of end-user vehicles could collect the large amounts of data because no reference truth has to be established. Standard series sensors could be built into the vehicles and tested in Shadow Mode in real environments [68], without having the automated driving functionalities activated. Hence, the approach could complement Shadow Mode test concepts [201, 277] for automated driving functionalities. The error rates would be learned offline in the backend based on the data of all vehicles in the fleet. Only a limited amount of data would be needed to be transferred to the backend. In contrast, an offline labeling approach requires transmitting sensor raw data to the backend.

This approach can facilitate capturing a high variation of driving situations in a sensor perception reliability assessment, and the test naturally is representative (see Section 3.1.4). The approach would allow generating the large amount of data needed to demonstrate strict reliability targets for the sensor perception and ultimately for the safety of automated driving.

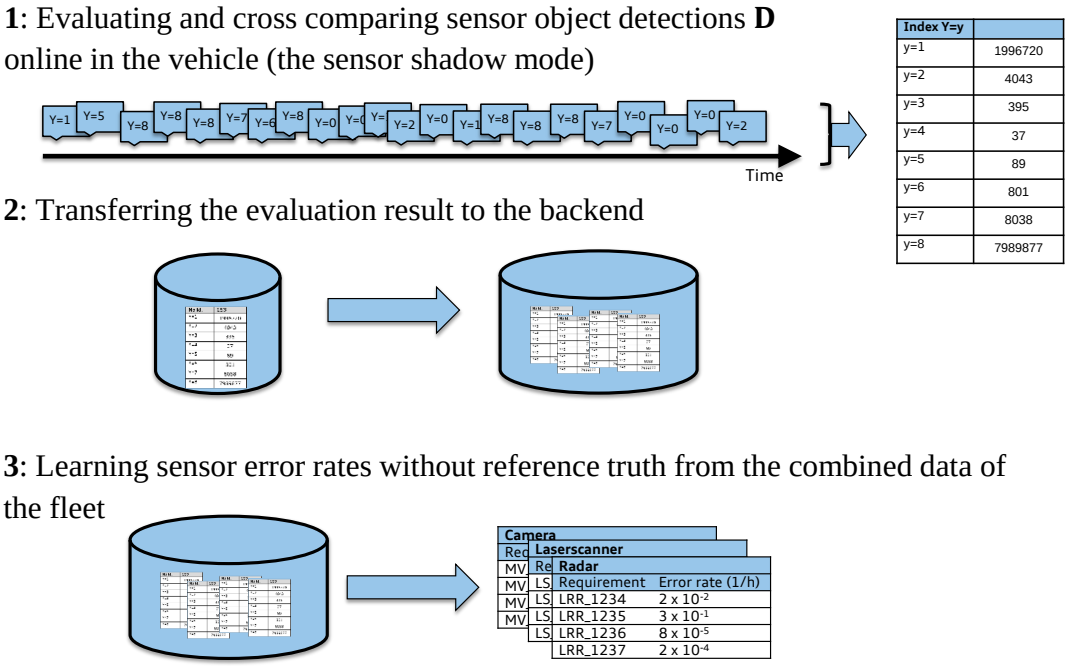

Figure 6.22 Outlook: Schematic concept of how to demonstrate the perception reliability with a fleet learning approach.[73]

[73] This Figure was originally created by Olaf Schubert, one of the supervisors of this research project.

Conclusion

We investigated on a theoretical basis if it is possible to learn sensor perception reliabilities without a reference truth. To this end, we presented a framework for estimating reliabilities of binary sensor data by exploiting sensor redundancies. The presented approach could enable learning sensor detection and false alarm rates as well as sensor error dependencies from a fleet of end-user vehicles because a reference truth is not required.

In synthetic case studies, we demonstrate that correct sensor error rates are learned despite the lack of certainty, whenever the selected sensor error dependence model is adequate. Additionally, we study the estimation errors made by the framework if the dependence model is not adequate. This Section presents an alternative testing and demonstration procedure for *sensor perception reliability*, complementing existing methods such as empirical tests with a reference truth and virtual simulations. Eventually, the developed framework could contribute to demonstrate the safety of automated driving vehicles. However, further investigations into the adequacy of statistical models to describe dependence among sensor errors are needed.

7 Conclusions and Outlook

7.1 Concluding Remarks

Even though the introduction of *automated driving systems* (ADSs) promises to increase traffic safety [3, 31–38], ADSs can and probably will cause fatal accidents. Due to technical complexities, at present open questions are how to develop safe ADSs with *higher levels of driving automation* (≥ level 3) and how to demonstrate that a developed ADS is complying with its safety requirements. To prevent unacceptable risk for society and the ADSs' producers, the safety of ADSs has to be demonstrated before their release to public traffic. Biased public risk perception [122, 125, 129, 142–144] implies that ADSs need to comply with strict safety targets.

ADSs are enabled by environment perceiving sensors such as radar, camera and lidar as well as a variety of processing and decision making algorithms. Of particular relevance for an ADS's safety is the perception based on the fused data of radar, lidar and camera sensors, because perception errors can be safety critical. Depending on the postulated capabilities of an ADS (i.e. the level of driving automation), the ADS's perception replaces human perception. For claiming *higher levels of driving automation*, it is crucial to demonstrate sufficient *perception reliability* (i.e. *probability of absence of safety-critical perception errors)* to ensure system safety.

Current standards such as ISO 26262 for functional safety and established test procedures (scenario based testing in simulations and on proving grounds, field tests) do not formally allow to demonstrate an ADS's *perception reliability* without additional measures. In practice for instance, it is hardly possible to empirically demonstrate sufficient *perception reliability* and consequently an ADS's safety because of the approval trap [35], i.e. the impracticably large number of required kilometers in field tests. Additionally, due to the complexities of environment perception, whose performance depends on numerous context variables (factors in the environment such as weather, properties of traffic participants…), the requirements for an ADS's perception can likely neither be comprehensively specified nor verified in tests. This would result in a near infinite number of specifications and, consequently, test cases.

These difficulties have motivated this thesis' objective of developing reliability analysis methods for environment perception in ADSs. By assessing *perception reliability*, it aims at contributing to a validation of the safety of the intended ADS functionality (SOTIF). Central research questions to achieve this objective are: how to derive reliability requirements for components (individual sensors) of an ADS's perception system and how to actually demonstrate these requirements, to ensure sufficient ADS *perception reliability*?

7.2 Contributions of this Thesis

In this thesis, we comprehensively structured the task of demonstrating an ADS's *perception reliability*, we identified challenges involved with this task, and we developed some possible solutions to these challenges. Our main original contributions are:

1. In Chapter 3, we formalized the task and challenges of demonstrating *perception reliability* for ADSs with *higher levels of driving automation* with Eq. (3.4). These challenges include:
   - The approval trap for environment perception
   - Perception error definition
   - Generating a reference truth
   - Time variable perception performance
   - Representativeness of tests
   - Statistical perception error dependence
   - System modifications during development

2. Based on a literature review, we comprehensibly outlined in Chapter 3 why established safety validation and testing procedures are not necessarily sufficient to achieve and demonstrate acceptable safety levels as well as *perception reliability* for an ADS with *higher levels of driving automation*. Main reasons are existing standards not addressing the safety of the intended functionality (SOTIF), the driver being out of the loop (limited driver controllability of an ADS with *higher levels of driving automation*), and testing procedures not being directly applicable:
   - Simulation frameworks for sole and comprehensive validation of *perception reliability* do not yet exist.
   - Scenario based testing has the limitation that an ADS has to handle a near infinite number of situations.
   - Field tests are impractical due to the large number of required test kilometers.

3. In Chapter 4, we described an ADS's *perception reliability* in three central uncertainty domains of environment perception (existence, classification and state uncertainty) with suitable metrics. The metrics are conditional probabilities collected in confusion matrices for existence and classification uncertainties, and joint probability density functions for state uncertainties. We related these metrics to the safety of an ADS with the example of existence uncertainties. Additionally, we introduced the notion of *higher order uncertainties* (uncertainty in the reliability metrics themselves).

4. We proposed a strategy to demonstrate *perception reliability* with feasible test effort by combining statistical dependence models for perception errors with a conceptualization of sensor data fusion in terms of individual environment perceiving sensors (k-out-of-n system) in Chapter 4. The rationale is that ADS safety can be formally demonstrated inductively (empirically), when combined with a deductive system description, as redundancies on the component (e.g. sensor) level lead to reduced reliability requirements for components. With this approach, we derived *sensor perception reliability* requirements, i.e. targets on *perception reliability* for individual sensors. Additionally, we presented a simple Bayesian method to estimate the required test effort for a given reliability target.
5. We developed a variety of methods to assess *sensor perception reliability* in Chapter 5. These include:
   - A qualitative and semi-quantitative analysis method to identify context variables. With this analysis one can preliminarily estimate the risk for perception due to a context variable.
   - A stochastic physics-based simulation framework for an assessment of rainfall influences on a lidar's perception performance. The framework exemplarily illustrates how simulation methods can help in the ADS development to verify requirements on *perception reliability*.
   - A framework to learn *sensor perception reliabilities* on proving grounds. This framework allows to learn *sensor perception* reliability more efficiently compared with pass /fail evaluated scenario based tests, because it combines the performance of environment perception in dependence of context variables with the exposure towards the context variables in a prediction.
   - Flexible dependence models for sensor errors to describe and jointly learn *sensor perception reliabilities* of redundant sensors in field tests with a reference truth.

6. Due to the considerable effort and technical difficulties of generating a reference truth, we investigated in Chapter 6 on a theoretical basis if it is possible to learn *sensor perception reliabilities* without a reference truth, by exploiting sensor redundancy. To this end, we devised a framework to learn *sensor perception reliabilities* by exploiting redundancies under simplified conditions (no distinction among different error types was made and we assumed identical sensors). We found that it is possible to accurately learn *sensor perception reliabilities* without a reference truth, if an adequate statistical model is chosen for the dependence of errors among different sensors. Based on this initial finding, we extended the framework to overcome the simplifying initial assumptions, i.e. we introduced a distinction in false positive and false negative sensor errors and a distinction in different sensors to the framework.

7.3 Discussion

In summary, to address the approval trap, we a) introduced metrics to describe the reliability of environment perception, b) decomposed an ADS's functionalities into perception (sense), function (plan) and actuation (act), and c) conceptualized perception (sense) in terms of individual environment perceiving sensors with a k-out-of-n model. Due to redundancies at the sensor level, requirements on *sensor perception reliability* derived with this approach and the assumptions made are easier to demonstrate than requirements at the ADS level. In combination with d) a variety of tools for assessing *sensor perception reliability*, it might become possible to overcome the approval trap and drive to safety. Points a)-d) are central for the *perception reliability* analysis methods presented in this Thesis.

A prerequisite for assessing *(sensor) perception reliability* is a systematic definition of safety-critical perception errors. We have only partly addressed this challenge, which is related to a), the description of environment *perception reliability* with suitable metrics. Important to our description of *perception reliability* are false positive (FP) and false negative (FN) errors (existence uncertainties), formalized in terms of a binary confusion matrix. While the methods presented in this thesis are generic and applicable to any specific definition of e.g. FP and FN errors, we did not investigate an optimal specific definition of safety-critical FP and FN errors. This specific definition of FP and FN errors could for instance be related to an association rule for the ground truth and the perceived (sensor) objects based on state quantities. Despite formulating *perception reliability* metrics for state and classification uncertainties, we did not specifically relate these uncertainties to safety critical perception errors. For example, we did not include incorrect estimations of state quantities (state uncertainties) such as a wrong object yaw angles in the definition of safety critical perception errors. However, one could implicitly include these error types in the definition of FP and FN errors with an adequate association rule for (sensor) objects and the ground truth.

It is not trivial to define safety-critical perception errors within the development process of an ADS, because at the point of time when one needs this error definition, the sensor data fusion, the situation interpretation and path planning of the ADS are typically not yet completed. During development, the system is modified constantly, e.g. with software updates altering the processing algorithms. It is the combination of sensor data fusion, situation interpretation, path planning together with the specific driving situation that determines the safety-criticality of perception errors. Therefore, one can probably only derive comprehensive perception requirements at the end of the system development, but a catalogue of perception errors that are to be prevented is required in the beginning of sensor development.

Even if an ADS is fully developed, it is not trivial to formally derive perception error definitions by analyzing the system. A system, which is adapted constantly (and often in a heuristic way) by expertise collected during the development, ultimately is hard to model. Additionally, the safety-criticality of perception errors is not only a function of the system itself, but also of the specific situation, i.e. the environment and other traffic participants. To address the discussed challenges, we proposed conservative heuristics in the definition of perception errors, however also simulation methods could be established. Our reasoning is that if the ADS complies with requirements under conservative heuristics, it is (very likely) also complying with a more specific definition of safety-critical perception errors and therefore safe. We point out that we neglected the complexity of environment perception to some degree with the binary problem interpretation underlying the confusion matrix for existence uncertainties. In reality, multiple objects are detected at once. With the binary problem interpretation utilized here, one can estimate e.g. *(sensor) perception reliabilities* related to a particular area of the field of view, such as the driving path.

The difficulty of specifying perception requirements and perception errors also challenges the established product development organization in the automotive industry. Frequently modifying the system during development requires agile methods, which are only partly practicable in collaboration with suppliers. E.g. a contracted sensor supplier typically needs a detailed specification of product requirements at the beginning of the development from the OEM, which – as pointed out above – is not readily available for perception. An option to address this dilemma is an initially well-planned (perception) system architecture designed on the safe side, because a system architecture which, due to underdesign, fails to comply with safety requirements has a large development risk. Additionally, suppliers could only be developing hardware, which has less complexity to be specified. In contrast, the software (i.e. processing algorithms for perception and the automated driving functionality) could be developed and optimized in-house of an OEM. Such an approach allows for agile development with which sufficient safety can probably be achieved more easily.

Under point b), we postulate the possibility of separating the reliabilities of perception, function and actuation by decomposing an ADS's functionalities into perception (sense), function (plan) and actuation (act). This decomposition enables to test and validate sensor perception reliability separately from an automated driving functionality. In practice, it is due to system complexities not trivial to clearly separate errors in e.g. perception (sense) and function (plan). An assumption of such a system decomposition is therefore that safety-critical errors in the perception module can be defined and separated from errors in the function module. Additionally, one must relate *sensor perception reliability* to safety targets of an automated driving functionality to validate perception separately from function. Setting up this relationship is itself challenging and requires further research. Involved difficulties are discussed above.

This thesis shows however, from a safety validation perspective, there is a huge benefit associated with decomposing an ADS's functionalities into perception (sense), function (plan) and actuation (act). A possible way forward would be to aim for modular system architectures with the possibility of a clear separation between sub-functionalities and clear interfaces between the sub-functionalities within an ADS. This would reduce interdependencies within a given ADS, and hence, reduce system complexities. In the spirit of design for test, it might be better to develop a modular system architecture (with increased cost in the beginning of the development) than to end up with a complex system whose safety is (e.g. due to the approval trap) extremely hard to assess. A modular system architecture is the prerequisite for breaking down the problem of ADS safety validation into smaller and manageable sub-problems.

The derivation of *sensor perception reliability* requirements in this thesis is conditional on point c), the conceptualization of sensor data fusion in terms of individual sensors, here with a simple k-out-of-n model. The applicability of the k-out-of-n model was already discussed in Section 4.2.1. A k-out-of-n model does not fully capture the complexity of sensor data fusion.

In practice, the parameterization and processing in sensor data fusion is updated frequently during development to include recent experience with a specific sensor behavior. The development progress hence is (partly) empirical and (partly) based on heuristics. Similar to the discussion above on decomposition, a sensor data fusion with many heuristics is difficult to model, and thus hard to be adequately conceptualized.

Because of the considerable optimization that goes into sensor data fusion during system development, the actual sensor data fusion likely performs better in most situations than a simple k-out-of-n model. We therefore believe that the k-out-of-n model is for most situations a conservative modeling choice. If the k-out-of-n model is conservative and the perception complies with its requirements under the k-out-of-n model, the perception is reliable. A problem is however, that a system might not comply with its requirements under the conservative k-out-of-n model, but the actual sensor data fusion is complying with the requirements. In this case, one would need to e.g. set up simulation methods to combine statistical sensor models with the actual software code of sensor data fusion in a SiL, to obtain more accurate reliability estimations, which requires considerable effort. Another potential solution would be to implement a sensor data fusion, which follows a formal model, i.e. the actual sensor data fusion can be conceptualized in terms of a model.

No model to represent sensor data fusion is required for d) learning *sensor perception reliability* with the methods in Chapters 5 and 6. Each of these methods addresses some but not all challenges in assessing *sensor perception reliability*. For example, simulations enable to collect a large number of virtual test kilometers. Because of the complexity of physical processes in environment perception, and consequently, the need to validate physics-based simulations, it does not seem

possible at present to solely assess *sensor perception reliability* virtually [44, 66]. If *sensor perception reliabilities* are learned with an adequate method from data, statistics based simulations are however a chance to analyze the performance of sensors and sensor data fusion jointly.

As a more realistic alternative, we proposed a framework to learn *sensor perception reliability* on proving grounds [161]. The framework alleviates the need for specifying an uncountable number of test cases because it provides statistical statements on *sensor perception reliability*. These statistical statements are derived by learning *sensor perception reliability* in dependence of the context variables and by considering the exposure to the context variables.

Field tests conducted in a representative manner naturally take context variables into account and are the most realistic choice of testing. Setting up a reference truth (i.e. labeled data) in field tests to analyze perception errors can be cumbersome (or imperfect), and because of the necessary large amount of testing, it seems impracticable to demonstrate sufficient reliability with field tests alone. We therefore developed a framework to assess *sensor perception reliability* without a reference truth, by exploiting sensor redundancies [156, 157]. The idea behind this framework is to equip end-user vehicles with the required hardware for automated driving and perception, but without activating the automated driving functionality. The proposed framework allows to test sensors in Shadow Mode [68]. The data can be preprocessed in the vehicles, transferred to the backend and *sensor perception reliabilities* are learned from the aggregated data of a fleet. With such an approach one would cover a vast amount of real driving situations, thereby accounting for context variables. As we discussed in Chapter 6, a prerequisite for learning *sensor perception reliability* with a fleet of end-user vehicles in Shadow Mode is an adequate dependence model for perception errors in different sensors. In this context, the adequacy of dependence models needs to be further studied before implementing the framework in practice.

Each of the different methods for assessing *sensor perception reliability* has strengths and weaknesses. Learning *sensor perception reliability* based on solely one of these methods does not appear to be sufficient. We therefore recommend to combine them in practice. For example, early in the system development, simulations could be employed to optimize the perception architecture and the sensor design. Simulations also allow to preclude systematic effects due to limitations of the sensor set and to preliminarily estimate *perception reliability* for selected context variables. Tests on proving grounds later enable to more realistically preclude unacceptable *sensor perception reliability* for known context variables. A fleet learning based Sensor Shadow Mode could contribute the required large scale testing to obtain credibility in strict reliability requirements. Finally, a limited field test with a reference truth could preclude an unacceptably large frequency of common cause errors among different sensors (i.e. shock events, all sensors failing deterministically, see Chapter 6), which are the most critical errors.

The considerable challenges pointed out in this discussion and the challenges summarized in Section 3.1.4 indicate that not all problems associated with validating *perception reliability* or ADS safety are solved. However, this thesis provides a starting point for further developments.

7.4 Future Research Opportunities

Research opportunities are summarized in this Section, which follow from the discussion in the previous Section and from open points in this thesis.

Importantly, future research should address in larger detail how to specifically define safety-critical perception errors. Due to the involved challenges, it could be necessary to develop heuristics in this definition, taking into account specific fusion and function (situation interpretation, path planning) algorithms. The error definition has to be considered holistically with special attention towards the interfaces of individual sensors with sensor data fusion, and of sensor data fusion with the automated driving functionality. The complete system needs to be understood for this task. To account for different driving situations, it might be necessary to have a flexible error definition, which depends on the specific situation.

Of practical relevance in this context are temporal aspects (e.g. how to model that perception errors can have a different duration and hence have different safety-criticality?) and spatial aspects (e.g. when does a true object in the environment count as detected by perception?). The latter is related to the definition of false positive and false negative sensor errors, which is not trivial in a multi-object environment. To extend the binary problem interpretation in this thesis, one needs e.g. rules on how to associate the output of perception with a ground truth (if available) for the purpose of a reliability assessment. Ultimately this is an association problem similar to the association in sensor data fusion, with the difference of not being limited by computational constraints and having the opportunity to propagate information back into the past. For example, one could propagate the shape of a vehicle back to previous time steps, when the shape is only fully revealed after observing a vehicle for some time. A simple solution to the association problem could be an association gate around a ground truth object in the hyperspace of relevant state quantities. It could also be studied in this context how to optimally discretize the field of view into cells, as exemplarily illustrated in Figure 4.6.

Not in the scope of this thesis was the reliability of interpreting a perceived situation. However, it is not only the perception but also its interpretation and the subsequent action that determines ADS safety. One could try to develop methods similar to the ones presented this thesis for situation interpretation.

It should be investigated in depth how to optimally conceptualize sensor data fusion in terms of individual environment perceiving sensors for the purpose of a *perception reliability* assessment. This investigation should assess how reliabilities and error dependencies of individual sensors propagate through a sensor data fusion algorithm. An adequate conceptualization of sensor data fusion allows to define *sensor perception reliability* requirements with an inverse relationship in function of *perception reliability*. As illustrated in Figure 4.9 with the k-out-of-n model, such a conceptualization has the potential of practically enabling an empirical safety demonstration on the individual sensor level. In this thesis, we did not explicitly model perception errors arising in sensor data fusion, e.g. a wrong association of sensor object detections. Future research should therefore extend the methods presented in this thesis to account for association uncertainties. Another avenue of future research is to connect the reliability of a raw data fusion to the reliabilities of individual sensors.

We see a large potential to test the perception in a vast number of real driving situations by a fleet of end-user vehicles with the framework presented in Chapter 6, which relies on exploiting sensor redundancy. As was discussed in depth, a prerequisite for its application is an adequate dependence model for perception errors among different sensors. A challenge to the framework is that one cannot detect without reference truth whether a statistical model is adequate or not. If it is not adequate, the estimated *sensor perception reliability* can be biased. Therefore, it needs to be studied which dependence model is adequate for the framework, before implementing it in practice in a fleet. Additionally, in most parts of the thesis, we neglected temporal statistical dependence in perception errors by assuming exchangeability. When developing suitable dependence models, one should also account for temporal dependencies. Ultimately, the dependence models are not only important for validation, but also for development (e.g. optimizing algorithms).

As was discussed, during development, the ADS is modified constantly. In this thesis we did not include a potential reliability growth in the presented models, or more generally, how to deal with system changes. A research opportunity is to explicitly account for such changes, for instance with common reliability growth models [151]. An alternative is the development of simulation methods to account for different software versions in the reliability estimation.

Finally, we recommended in the previous Section to combine different test methods (e.g. presented in Chapters 5 and 6) for a demonstration of *perception reliability*. An open question is to quantify how much each method contributes to the safety assurance, to obtain a single quantitative statement on *perception reliability*. Future research should study how to optimally combine different test methods. To large parts, this thesis was lying out theoretical foundations for analyzing *perception reliability*. Applying these methods in practice should enable to extend and optimize them with data.

References

[1] D. H. Keller, “The Living Machine,” in *Vol. 6*, No. 12, *Wonder Stories*, H. Gernsback, Ed., 1935.

[2] F. Kröger, “Automated Driving in Its Social, Historical and Cultural Contexts,” in *Autonomous Driving*, M. Maurer, J. C. Gerdes, B. Lenz, and H. Winner, Eds., Berlin, Heidelberg: Springer Berlin Heidelberg, 2016, pp. 41–68.

[3] R. E. Fenton, “Automatic vehicle guidance and control—A state of the art survey,” *IEEE Trans. Veh. Technol.*, vol. 19, no. 1, pp. 153–161, 1970.

[4] R. E. Fenton and K. W. Olson, “The electronic highway,” *IEEE Spectr.*, vol. 6, no. 7, pp. 60–66, 1969.

[5] S. Tsugawa, “Vision-based vehicles in Japan: Machine vision systems and driving control systems,” *IEEE Trans. Ind. Electron.*, vol. 41, no. 4, pp. 398–405, 1994.

[6] S. Tsugawa, T. Yatabe, T. Hirose, and S. Matsumoto, “An automobile with artificial intelligence,” in *Proceedings of the 6th international joint conference on Artificial intelligence-Volume 2*, 1979, pp. 893–895.

[7] H. P. Moravec, “The Stanford Cart and the CMU Rover,” *Proc. IEEE*, vol. 71, no. 7, pp. 872–884, 1983.

[8] E. D. Dickmanns, B. Mysliwetz, and T. Christians, “An integrated spatio-temporal approach to automatic visual guidance of autonomous vehicles,” *IEEE Trans. Syst., Man, Cybern.*, vol. 20, no. 6, pp. 1273–1284, 1990.

[9] R. Behringer and R.B.M. Maurer, “Results on visual road recognition for road vehicle guidance,” in *Proceedings of Conference on Intelligent Vehicles*, Tokyo, Japan, 1996, pp. 415–420.

[10] D. Pomerleau and T. Jochem, “Rapidly adapting machine vision for automated vehicle steering,” *IEEE Expert*, vol. 11, no. 2, pp. 19–27, 1996.

[11] B. Ulmer, “VITA II-active collision avoidance in real traffic,” in *Proceedings of the Intelligent Vehicles '94 Symposium*, Paris, France, 1994, pp. 1–6.

[12] S. Thrun *et al.*, “Stanley: The robot that won the DARPA Grand Challenge,” *J. Field Robotics*, vol. 23, no. 9, pp. 661–692, 2006.

[13] C. Urmson *et al.*, “Tartan racing: A multi-modal approach to the darpa urban challenge,” 2007.

[14] SAE International, “J3016: Taxonomy and Definitions for Terms Related to Driving Automation Systems for On-Road Motor Vehicles,” 2016.

[15] S. A. Beiker, “Deployment Scenarios for Vehicles with Higher-Order Automation,” in *Autonomous Driving*, M. Maurer, J. C. Gerdes, B. Lenz, and H. Winner, Eds., Berlin, Heidelberg: Springer Berlin Heidelberg, 2016, pp. 193–211.

[16] S. Kammel *et al.*, "Team AnnieWAY's autonomous system for the 2007 DARPA Urban Challenge," *J. Field Robotics*, vol. 25, no. 9, pp. 615–639, 2008.

[17] E. Guizzo, "How google's self-driving car works," *IEEE Spectrum Online, October*, vol. 18, 2011.

[18] U. Franke *et al.*, "Making Bertha See," in *2013 IEEE International Conference on Computer Vision Workshops (ICCVW)*, Sydney, Australia, 2013, pp. 214–221.

[19] M. Aeberhard *et al.*, "Experience, Results and Lessons Learned from Automated Driving on Germany's Highways," *IEEE Intell. Transport. Syst. Mag.*, vol. 7, no. 1, pp. 42–57, 2015.

[20] Audi, *Mission accomplished: Audi A7 piloted driving car completes 550-mile automated test drive.* [Online] Available: https://www.audiusa.com/newsroom/news/press-releases/2015/01/550-mile-piloted-drive-from-silicon-valley-to-las-vegas. Accessed on: Feb. 25 2016.

[21] Auto Express, *Audi A8 could get posher and longer to rival Maybach.* [Online] Available: http://www.autoexpress.co.uk/audi/a8/95433/audi-a8-could-get-posher-and-longer-to-rival-maybach. Accessed on: Jul. 05 2016.

[22] Auto Zeitung, *Audi A8 (2017): Neuer A8 mit Staupilot: Neuer A8 fährt bis 60 km/h selbst.* [Online] Available: http://www.autozeitung.de/auto-news/audi-a8-2017-staupilot. Accessed on: Jun. 22 2016.

[23] A. Broggi, S. Debattisti, P. Grisleri, and M. Panciroli, "The deeva autonomous vehicle platform," in *2015 IEEE Intelligent Vehicles Symposium (IV)*, Seoul, South Korea, pp. 692–699.

[24] Google, *Google Self-Driving Car Project.* [Online] Available: http://static.googleusercontent.com/media/www.google.com/en/us/selfdrivingcar/. Accessed on: Feb. 25 2016.

[25] J. Walker, *The Self-Driving Car Timeline - Predictions from the TOP 11 Global Automakers.* [Online] Available: https://www.techemergence.com/self-driving-car-timeline-themselves-top-11-automakers/. Accessed on: Nov. 02 2018.

[26] O. Pink, J. Becker, and S. Kammel, "Automated driving on public roads: Experiences in real traffic," *it - Information Technology*, vol. 57, no. 4, 2015.

[27] J. Becker, *BMW's driverless cars: A quantum leap to Level 5.* [Online] Available: https://www.2025ad.com/latest/2017-11/bmw-driverless-cars-level-5/. Accessed on: Nov. 02 2018.

[28] Daimler, *Bosch and Daimler: Metropolis in California to become a pilot city for automated driving.* [Online] Available: https://media.daimler.com/marsMediaSite/en/instance/ko/Bosch-and-Daimler-Metropolis-in-California-to-become-a-pilot-city-for-automated-driving.xhtml?oid=40688558. Accessed on: Nov. 02 2018.

[29] H. Winner, S. Hakuli, F. Lotz, and C. Singer, Eds., *Handbook of Driver Assistance Systems*. Cham: Springer International Publishing, 2016.

[30] F. Kröger, "Das automatisierte Fahren im gesellschaftsgeschichtlichen und Kulturwissenschaftlichen Kontext," in *Autonomes Fahren: Technische, rechtliche und gesellschaftliche Aspekte*, M. Maurer, J. C. Gerdes, B. Lenz, and H. Winner, Eds., Berlin, Heidelberg: Springer Berlin Heidelberg, 2015, pp. 41–67.

[31] R. Matthaei *et al.*, "Autonomous Driving," in *Handbook of Driver Assistance Systems*, H. Winner, S. Hakuli, F. Lotz, and C. Singer, Eds., Cham: Springer International Publishing, 2016, pp. 1519–1556.

[32] D. Heinrichs, "Autonomous Driving and Urban Land Use," in *Autonomous Driving*, M. Maurer, J. C. Gerdes, B. Lenz, and H. Winner, Eds., Berlin, Heidelberg: Springer Berlin Heidelberg, 2016, pp. 213–231.

[33] J. M. Anderson *et al.*, *Autonomous vehicle technology: A guide for policymakers*, 2016.

[34] M. Pavone, "Autonomous Mobility-on-Demand Systems for Future Urban Mobility," in *Autonomes Fahren: Technische, rechtliche und gesellschaftliche Aspekte*, M. Maurer, J. C. Gerdes, B. Lenz, and H. Winner, Eds., Berlin, Heidelberg: Springer Berlin Heidelberg, 2015, pp. 399–416.

[35] H. Winner, "ADAS, Quo Vadis?," in *Handbook of Driver Assistance Systems*, H. Winner, S. Hakuli, F. Lotz, and C. Singer, Eds., Cham: Springer International Publishing, 2016, pp. 1557–1584.

[36] P. Wagner, "Traffic Control and Traffic Management in a Transportation System with Autonomous Vehicles," in *Autonomous Driving*, M. Maurer, J. C. Gerdes, B. Lenz, and H. Winner, Eds., Berlin, Heidelberg: Springer Berlin Heidelberg, 2016, pp. 301–316.

[37] B. Friedrich, "The Effect of Autonomous Vehicles on Traffic," in *Autonomous Driving*, M. Maurer, J. C. Gerdes, B. Lenz, and H. Winner, Eds., Berlin, Heidelberg: Springer Berlin Heidelberg, 2016, pp. 317–334.

[38] T. M. Gasser *et al.*, "Legal consequences of an increase in vehicle automation: Consolidated final report of the project group," *BASt-Report F83 (Part 1)*, no. 83, 2012.

[39] M. Berk, O. Schubert, H.-M. Kroll, B. Buschardt, and D. Straub, "Assessing the Safety of Environment Perception in Automated Driving Vehicles," *Submitted to SAE International Journal of Transportation Safety*, 2019.

[40] A. Weitzel *et al.*, "Absicherungsstrategien für Fahrerassistenzsysteme mit Umfeldwahrnehmung," *Berichte der Bundesanstalt für Straßenwesen. Unterreihe Fahrzeugtechnik*, no. 98, 2014.

[41] N. Kalra and S. M. Paddock, "Driving to safety: How many miles of driving would it take to demonstrate autonomous vehicle reliability?," *Transportation Research Part A: Policy and Practice*, vol. 94, pp. 182–193, 2016.

[42] S. Shalev-Shwartz, S. Shammah, and A. Shashua, “On a Formal Model of Safe and Scalable Self-driving Cars,” *arXiv preprint arXiv:1708.06374*, 2017.

[43] T. Winkle, “Development and Approval of Automated Vehicles: Considerations of Technical, Legal, and Economic Risks,” in *Autonomous Driving*, M. Maurer, J. C. Gerdes, B. Lenz, and H. Winner, Eds., Berlin, Heidelberg: Springer Berlin Heidelberg, 2016, pp. 589–618.

[44] W. Wachenfeld and H. Winner, “The Release of Autonomous Vehicles,” in *Autonomous Driving*, M. Maurer, J. C. Gerdes, B. Lenz, and H. Winner, Eds., Berlin, Heidelberg: Springer Berlin Heidelberg, 2016, pp. 425–449.

[45] T. M. Gasser, A. Seeck, and B. W. Smith, “Framework Conditions for the Development of Driver Assistance Systems,” in *Handbook of Driver Assistance Systems*, H. Winner, S. Hakuli, F. Lotz, and C. Singer, Eds., Cham: Springer International Publishing, 2016, pp. 35–68.

[46] A. Reschka, “Safety Concept for Autonomous Vehicles,” in *Autonomous Driving*, M. Maurer, J. C. Gerdes, B. Lenz, and H. Winner, Eds., Berlin, Heidelberg: Springer Berlin Heidelberg, 2016, pp. 473–496.

[47] *Vienna Convention on Road Traffic*, 1968.

[48] Deutscher Bundestag, “Entwurf eines Gesetzes zur Änderung der Artikel 8 und 39 des Übereinkommens vom 8. November 1968 über den Straßenverkehr: Beschlussempfehlung und Bericht des Ausschusses für Verkehr und digitale Infrastruktur (15. Ausschuss,” Sep. 2016.

[49] T. M. Gasser *et al.*, “Rechtsfolgen zunehmender Fahrzeugautomatisierung,” *Berichte der Bundesanstalt für Straßenwesen. Unterreihe Fahrzeugtechnik*, no. 83, 2012.

[50] P. Lin, “Why Ethics Matters for Autonomous Cars,” in *Autonomous Driving*, M. Maurer, J. C. Gerdes, B. Lenz, and H. Winner, Eds., Berlin, Heidelberg: Springer Berlin Heidelberg, 2016, pp. 69–85.

[51] J. C. Gerdes and S. M. Thornton, “Implementable Ethics for Autonomous Vehicles,” in *Autonomous Driving*, M. Maurer, J. C. Gerdes, B. Lenz, and H. Winner, Eds., Berlin, Heidelberg: Springer Berlin Heidelberg, 2016, pp. 87–102.

[52] Ethics Commission, “Automated and Connected Driving: Appointed by the Federal Minister of Transport and Digital Infrastructure,” Jun. 2017. [Online] Available: WWW.BMVI.DE.

[53] E. Donges, “Driver Behavior Models,” in *Handbook of Driver Assistance Systems*, H. Winner, S. Hakuli, F. Lotz, and C. Singer, Eds., Cham: Springer International Publishing, 2016, pp. 19–33.

[54] U. Wilhelm, S. Ebel, and A. Weitzel, “Functional Safety of Driver Assistance Systems and ISO 26262,” in *Handbook of Driver Assistance Systems*, H. Winner, S. Hakuli, F. Lotz, and C. Singer, Eds., Cham: Springer International Publishing, 2016.

[55] J. Breuer, C. von Hugo, S. Mücke, and S. Tattersall, “User-Oriented Evaluation of Driver Assistance Systems,” in *Handbook of Driver Assistance Systems*, H. Winner, S. Hakuli, F. Lotz, and C. Singer, Eds., Cham: Springer International Publishing, 2016, pp. 231–248.

[56] PReVENT, “Code of Practice for the Design and Evaluation of ADAS,” Aug. 2009.

[57] H. Durrant-Whyte and T. C. Henderson, “Multisensor Data Fusion,” in *Springer Handbook of Robotics*, B. Siciliano and O. Khatib, Eds., Berlin, Heidelberg: Springer Berlin Heidelberg, 2008, pp. 585–610.

[58] M. Darms, “Data Fusion of Environment-Perception Sensors for ADAS,” in *Handbook of Driver Assistance Systems*, H. Winner, S. Hakuli, F. Lotz, and C. Singer, Eds., Cham: Springer International Publishing, 2016, pp. 549–566.

[59] M. Mählisch, *Filtersynthese zur simultanen Minimierung von Existenz-, Assoziations- und Zustandsunsicherheiten in der Fahrzeugumfelderfassung mit heterogenen Sensordaten*. Ulm: Univ. Ulm, 2009.

[60] K. Dietmayer, “Predicting of Machine Perception for Automated Driving,” in *Autonomous Driving*, M. Maurer, J. C. Gerdes, B. Lenz, and H. Winner, Eds., Berlin, Heidelberg: Springer Berlin Heidelberg, 2016, pp. 407–424.

[61] P. Glauner, A. Blumenstock, and M. Haueis, “Effiziente Felderprobung von Fahrerassistenzsystemen,” in *8. Workshop Fahrerassistenzsysteme: FAS 2012*, Walting, 2012, pp. 5–14.

[62] *Road vehicles — Functional safety — Part 1: Vocabulary*, ISO 26262-1:2011(E), 2011.

[63] A. Weitzel, “Objektive Bewertung der Kontrollierbarkeit nicht situationsgerechter Reaktionen umfeldsensorbasierter Fahrerassistenzsysteme,” Dissertation, Fachbereich Maschinenbau, Technischen Universität Darmstadt, Darmstadt, 2013.

[64] M. Fach, F. Baumann, J. Breuer, and A. May, “Bewertung der Beherrschbarkeit von Aktiven Sicherheits-und Fahrerassistenzsystemen an den Funktionsgrenzen,” in *Fahrerassistenz und Integrierte Sicherheit: 26. VDI/VW-Gemeinschaftstagung*, Wolfsburg, 2010, pp. 425–435.

[65] M. Berk, H.-M. Kroll, O. Schubert, B. Buschardt, and D. Straub, “Bayesian Test Design for Reliability Assessments of Safety-Relevant Environment Sensors Considering Dependent Failures,” in *SAE Technical Paper 2017-01-0050*, 2017.

[66] W. Wachenfeld and H. Winner, “The New Role of Road Testing for the Safety Validation of Automated Vehicles,” in *Automated Driving: Safer and More Efficient Future Driving*, D. Watzenig and M. Horn, Eds., Cham: Springer International Publishing, 2017, pp. 419–435.

[67] K. C. J. Dietmayer, S. Reuter, and D. Nuss, “Representation of Fused Environment Data,” in *Handbook of Driver Assistance Systems*, H. Winner, S. Hakuli, F. Lotz, and C. Singer, Eds., Cham: Springer International Publishing, 2016, pp. 567–603.

[68] F. Netter, “Künstliche Intelligenz im Auto — Applikationen, Technologien und Herausforderungen,” *ATZelektronik*, no. 1, pp. 20–24, https://www.springerprofessional.de/kuenstliche-intelligenz-im-auto-applikationen-technologien-und-h/15131302, 2017.

[69] R. A. Brooks, “Intelligence without reason,” *Artificial intelligence: critical concepts*, vol. 3, pp. 107–163, 1991.

[70] M. Shanahan, “Reinventing shakey,” in *Logic-based artificial intelligence*: Springer, 2000, pp. 233–253.

[71] F. Saust, T. C. Müller, J. M. Wille, and M. Maurer, “Entwicklungsbegleitendes Simulations- und Testkonzept für autonome Fahrzeuge in städtischen Umgebungen,” *AAET 2009. Automatisierungs-, Assistenzsysteme und eingebettete Systeme für Transportmittel*, vol. 129, 2009.

[72] N. Steinhardt and S. Leinen, “Data Fusion for Precise Localization,” in *Handbook of Driver Assistance Systems*, H. Winner, S. Hakuli, F. Lotz, and C. Singer, Eds., Cham: Springer International Publishing, 2016, pp. 605–646.

[73] Bosch, *Ultrasonic sensor: Characteristics of the ultrasonic sensor.* [Online] Available: https://www.bosch-mobility-solutions.com/en/products-and-services/passenger-cars-and-light-commercial-vehicles/driver-assistance-systems/construction-zone-assist/ultrasonic-sensor/. Accessed on: Apr. 24 2018.

[74] M. Noll and P. Rapps, “Ultrasonic Sensors for a K44DAS [sic!],” in *Handbook of Driver Assistance Systems*, H. Winner, S. Hakuli, F. Lotz, and C. Singer, Eds., Cham: Springer International Publishing, 2016, pp. 303–323.

[75] H. Kuttruff, *Physik und Technik des Ultraschalls*. Stuttgart: Hirzel, 1988.

[76] H. Winner, “Automotive RADAR,” in *Handbook of Driver Assistance Systems*, H. Winner, S. Hakuli, F. Lotz, and C. Singer, Eds., Cham: Springer International Publishing, 2016, pp. 325–403.

[77] J. A. Scheer, “The Radar Range Equation,” in *Principles of modern radar*, W. A. Holm, M. A. Richards, and J. A. Scheer, Eds., 2010th ed., Raleigh, NC: SciTech Publ, 2015, pp. 59–86.

[78] W. A. Holm, M. A. Richards, and J. A. Scheer, Eds., *Principles of modern radar*, 2010th ed. Raleigh, NC: SciTech Publ, 2015.

[79] M. I. Skolnik, Ed., *Radar handbook*, 3rd ed. New York NY u.a.: McGraw-Hill, 2008.

[80] A. Ludloff, *Praxiswissen Radar und Radarsignalverarbeitung*, 4th ed. Wiesbaden: Vieweg + Teubner, 2008.

[81] C. Wolff, *Frequency-Modulated Continuous-Wave Radar (FMCW Radar).* [Online] Available: http://www.radartutorial.eu/02.basics/Frequenzmodulierte%20Dauerstrichradarger%C3%A4te.de.html. Accessed on: Apr. 28 2018.

[82] C. Wolff, *Antenna Pattern.* [Online] Available: http://www.radartutorial.eu/06.antennas/Antenna%20Pattern.en.html. Accessed on: Apr. 29 2018.

[83] A. Arage, W. M. Steffens, G. Kuehnle, and R. Jakoby, "Effects of water and ice layer on automotive radar," in *Proceedings of the German Microwave Conference*, 2006.

[84] M. Punke, S. Menzel, B. Werthessen, N. Stache, and M. Höpfl, "Automotive Camera (Hardware)," in *Handbook of Driver Assistance Systems*, H. Winner, S. Hakuli, F. Lotz, and C. Singer, Eds., Cham: Springer International Publishing, 2016, pp. 431–460.

[85] C. Stiller, A. Bachmann, and A. Geiger, "Fundamentals of Machine Vision," in *Handbook of Driver Assistance Systems*, H. Winner, S. Hakuli, F. Lotz, and C. Singer, Eds., Cham: Springer International Publishing, 2016, pp. 461–493.

[86] S. Gehrig and U. Franke, "Stereovision for ADAS," in *Handbook of Driver Assistance Systems*, H. Winner, S. Hakuli, F. Lotz, and C. Singer, Eds., Cham: Springer International Publishing, 2016, pp. 495–524.

[87] J.-E. Källhammer, "Night vision: Requirements and possible roadmap for FIR and NIR systems," in Strasbourg, France, 2006, 61980F.

[88] P. Viola and M. Jones, "Rapid object detection using a boosted cascade of simple features," in *Proceedings of the 2001 IEEE Computer Society Conference on Computer Vision and Pattern Recognition. CVPR 2001*, Kauai, HI, USA, 2001, I-511-I-518.

[89] M. Nentwig and M. Stamminger, "A method for the reproduction of vehicle test drives for the simulation based evaluation of image processing algorithms," in *13th International IEEE Conference on Intelligent Transportation Systems*, Funchal, Madeira Island, Portugal, 2010, pp. 1307–1312.

[90] P. Geismann and G. Schneider, "A two-staged approach to vision-based pedestrian recognition using Haar and HOG features," in *2008 IEEE Intelligent Vehicles Symposium*, Eindhoven, Netherlands, 2008, pp. 554–559.

[91] Y. Freund and R. E. Schapire, "A decision-theoretic generalization of on-line learning and an application to boosting," *Journal of computer and system sciences*, vol. 55, no. 1, pp. 119–139, 1997.

[92] G. E. Hinton, S. Osindero, and Y.-W. Teh, "A fast learning algorithm for deep belief nets," *Neural computation*, vol. 18, no. 7, pp. 1527–1554, 2006.

[93] A. Géron, *Hands-on machine learning with Scikit-Learn and TensorFlow: Concepts, tools, and techniques to build intelligent systems*, 2017.

[94] A. Krizhevsky, I. Sutskever, and G. E. Hinton, "ImageNet Classification with Deep Convolutional Neural Networks," in *Advances in Neural Information Processing Systems 25*, F. Pereira, C. J. C. Burges, L. Bottou, and K. Q. Weinberger, Eds.: Curran Associates, Inc, 2012, pp. 1097–1105.

[95] I. Goodfellow, Y. Bengio, and A. Courville, *Deep learning*. Cambridge, Massachusetts, London, England: MIT Press, 2016.

[96] W. Liu *et al.,* "SSD: Single shot multibox detector," in *European conference on computer vision*, 2016, pp. 21–37.

[97] S. Ren, K. He, R. Girshick, and J. Sun, "Faster R-CNN: Towards Real-Time Object Detection with Region Proposal Networks," in *Advances in Neural Information Processing Systems 28*, C. Cortes, N. D. Lawrence, D. D. Lee, M. Sugiyama, and R. Garnett, Eds.: Curran Associates, Inc, 2015, pp. 91–99.

[98] J. Dai, Y. Li, K. He, and J. Sun, "R-FCN: Object Detection via Region-based Fully Convolutional Networks," in *Advances in Neural Information Processing Systems 29*, D. D. Lee, M. Sugiyama, U. V. Luxburg, I. Guyon, and R. Garnett, Eds.: Curran Associates, Inc, 2016, pp. 379–387.

[99] A. Geiger, P. Lenz, and R. Urtasun, "Are we ready for Autonomous Driving? The KITTI Vision Benchmark Suite," in *Conference on Computer Vision and Pattern Recognition (CVPR)*, 2012.

[100] E. Reinhard, E. A. Khan, and Akyuz, Ahmet Oguz, Johnson, Garret, *Color imaging: Fundamentals and applications*. Wellesley Mass.: Peters, 2008.

[101] J. R. V. Rivero *et al.,* "Characterization and simulation of the effect of road dirt on the performance of a laser scanner," in *2017 IEEE 20th International Conference on Intelligent Transportation Systems (ITSC)*, Yokohama, 2017, pp. 1–6.

[102] H. Gotzig and G. Geduld, "Automotive LIDAR," in *Handbook of Driver Assistance Systems*, H. Winner, S. Hakuli, F. Lotz, and C. Singer, Eds., Cham: Springer International Publishing, 2016, pp. 405–430.

[103] S. Kruapech and J. Widjaja, "Laser range finder using Gaussian beam range equation," *Optics & Laser Technology*, vol. 42, no. 5, pp. 749–754, 2010.

[104] R. H. Rasshofer, M. Spies, and H. Spies, "Influences of weather phenomena on automotive laser radar systems," *Adv. Radio Sci.*, vol. 9, pp. 49–60, 2011.

[105] M. Berk *et al.,* "A Stochastic Physical Simulation Framework to Quantify the Effect of Rainfall on Automotive Lidar," in *SAE Technical Paper 2019-01-0134*, 2019.

[106] A. Petrovskaya and S. Thrun, "Model based vehicle detection and tracking for autonomous urban driving," *Autonomous Robots*, vol. 26, no. 2-3, pp. 123–139, 2009.

[107] M. Bruneau *et al.,* "A Framework to Quantitatively Assess and Enhance the Seismic Resilience of Communities," *Earthquake Spectra*, vol. 19, no. 4, pp. 733–752, 2003.

[108] T. Schaller and B. Dehlink, "Sensor Standardization Initiative for Automated Driving," in *8. Tagung Fahrerassistenz*, Munich, 2017.

[109] M. Darms, "Eine Basis-Systemarchitektur zur Sensordatenfusion von Umfeldsensoren für Fahrerassistenzsysteme," Dissertation, Maschinenbau, Technische Universität Darmstadt, Darmstadt, 2007.

[110] D. L. Hall and J. Llinas, “An introduction to multisensor data fusion,” *Proc. IEEE*, vol. 85, no. 1, pp. 6–23, 1997.

[111] S. Kraemer, M. E. Bouzouraa, and C. Stiller, “Simultaneous tracking and shape estimation using a multi-layer laserscanner,” in *2017 IEEE 20th International Conference on Intelligent Transportation Systems (ITSC)*, Yokohama, 2017, pp. 1–7.

[112] J. M. Richardson and K. A. Marsh, “Fusion of Multisensor Data,” *The International Journal of Robotics Research*, vol. 7, no. 6, pp. 78–96, https://doi.org/10.1177/027836498800700607, 1988.

[113] E. Schröder, M. Mählisch, J. Vitay, and F. Hamker, “Fusion of Camera and Lidar Data for Object Detection using Neural Networks,” in *12. Workshop Fahrerassistenzsysteme und automatisiertes Fahren: FAS 2018*, Walting, 2018.

[114] A. Gelman *et al.*, *Bayesian Data Analysis, Third Edition*, 3rd ed. Hoboken: CRC Press, 2013.

[115] K. P. Murphy, *Machine learning: A probabilistic perspective*. Cambridge, Massachusetts: MIT Press, 2012.

[116] F. V. Jensen, M. Jordan, J. Kleinberg, T. D. Nielsen, and B. Schölkopf, Eds., *Bayesian Networks and Decision Graphs: February 8, 2007*. New York, NY: Springer New York, 2007.

[117] Y. Bar-Shalom, F. Daum, and J. Huang, “The probabilistic data association filter,” *IEEE Control Systems*, vol. 29, no. 6, 2009.

[118] D. Musicki and R. Evans, “Joint integrated probabilistic data association: JIPDA,” *IEEE transactions on Aerospace and Electronic Systems*, vol. 40, no. 3, pp. 1093–1099, 2004.

[119] C. M. Bishop, *Pattern recognition and machine learning*, 8th ed. New York, NY: Springer, 2009.

[120] *Act on making products available on the market*, 2012.

[121] *Allgemeine Leitsätze für das sicherheitsgerechte Gestalten von Produkten*, DIN 31000:2011-05.

[122] I. Häring, “Risk Acceptance Criteria,” in *Risk Analysis and Management: Engineering Resilience*, I. Häring, Ed., Singapore: Springer Singapore, 2015, pp. 313–342.

[123] I. Häring, “Introduction to Risk Analysis and Risk Management Processes,” in *Risk Analysis and Management: Engineering Resilience*, I. Häring, Ed., Singapore: Springer Singapore, 2015, pp. 9–26.

[124] D. Straub, “Engineering Risk Assessment,” in *Risk - A Multidisciplinary Introduction*, C. Klüppelberg, D. Straub, and I. M. Welpe, Eds., Cham: Springer International Publishing, 2014, pp. 333–362.

[125] I. L. Johansen, “Foundations and fallacies of risk acceptance criteria,” Norwegian University of Science and Technology; Dept. of Production and Quality Engineering, Trondheim, ROSS (NTNU) 201001, 2010.

[126] Health & Safety Executive, *Reducing risks, protecting people: HSE's decision-making process*. Sudbury: HSE Books, 2001.

[127] *Railway applications - The specification and demonstration of reliability, availability, maintainability and safety (RAMS)*, EN 50126:1999, 2000.

[128] H. Schäbe, "Different approaches for determination of tolerable hazard rates," in *ESREL Conference Proceedings*, Torino, 2001, pp. 435–442.

[129] C. Starr, "Social Benefit versus Technological Risk," *Science*, vol. 165, no. 3899, pp. 1232–1238, http://www.jstor.org/stable/1727970, 1969.

[130] W. K. Viscusi and J. E. Aldy, "The value of a statistical life: A critical review of market estimates throughout the world," *Journal of risk and uncertainty*, vol. 27, no. 1, pp. 5–76, 2003.

[131] J. K. Hammitt, "Valuing Mortality Risk: Theory and Practice †," *Environ. Sci. Technol.*, vol. 34, no. 8, pp. 1396–1400, 2000.

[132] United States Environmental Protection Agency, "Mortality Risk Valuation," 2019. [Online] Available: https://www.epa.gov/environmental-economics/mortality-risk-valuation#whatvalue. Accessed on: Mar. 18 2019.

[133] United States Environmental Protection Agency, "Valuing mortality risk reductions for policy: a meta-analytic approach," 2016. [Online] Available: https://yosemite.epa.gov/sab/SABPRODUCT.NSF/81e39f4c09954fcb85256ead006be86e/0CA9E925C9A702F285257F380050C842/$File/VSL+white+paper_final_020516.pdf.

[134] W. Wachenfeld and H. Winner, "The Release of Autonomous Vehicles," in *Autonomous Driving*, M. Maurer, J. C. Gerdes, B. Lenz, and H. Winner, Eds., Berlin, Heidelberg: Springer Berlin Heidelberg, 2016, pp. 425–449.

[135] P. Feig, J. Schatz, V. Labenski, and T. Leonhardt, "Assessment of Technical Requirements for Level 3 and Beyond Automated Driving Systems Based on Naturalistic Driving and Accident Data Analysis: Paper Number 19-0280," in *26th International Technical Conference on The Enhanced Safety of Vehicles (ESV)*, Eindhoven, Netherlands, 2019.

[136] U.S. Department of Transportation, Bureau of Transportation Statistics, "Motor Vehicle Safety Data," [Online] Available: https://www.bts.gov/content/motor-vehicle-safety-data. Accessed on: Jul. 18 2018.

[137] Federal Aviation Administration, "AC 25.1309-1A: System Design and Analysis," 1988.

[138] *Road vehicles — Functional safety — Part 5: Product development at the hardware level*, ISO 26262-5:2011(E), 2011.

[139] *Product Liability Act*, 1989.

[140] *Zur Haftung des Fahrzeugherstellers für einen Produktfehler - Urteil vom 16.6.2009 - VI ZR 107/08*, 2009.

[141] P. Seininger and A. Weitzel, “Test Methods for Consumer Protection and Legislation for ADAS,” in *Handbook of Driver Assistance Systems*, H. Winner, S. Hakuli, F. Lotz, and C. Singer, Eds., Cham: Springer International Publishing, 2016, pp. 213–230.

[142] P. Slovic, “Perception of risk,” *Science*, vol. 236, no. 4799, pp. 280–285, 1987.

[143] A. Tversky and D. Kahneman, “Judgment under Uncertainty: Heuristics and Biases,” *Science*, vol. 185, no. 4157, pp. 1124–1131, http://www.jstor.org/stable/1738360, 1974.

[144] B. Fischhoff, P. Slovic, S. Lichtenstein, S. Read, and B. Combs, “How safe is safe enough?: A psychometric study of attitudes towards technological risks and benefits,” *Policy Sci*, vol. 9, no. 2, pp. 127–152, 1978.

[145] S. Mitsch, K. Ghorbal, and A. Platzer, “On Provably Safe Obstacle Avoidance for Autonomous Robotic Ground Vehicles,” in *Robotics: Science and Systems IX*, 2013.

[146] D. Althoff, J. J. Kuffner, D. Wollherr, and M. Buss, “Safety assessment of robot trajectories for navigation in uncertain and dynamic environments,” *Auton Robot*, vol. 32, no. 3, pp. 285–302, 2012.

[147] S. Bouraine, T. Fraichard, and H. Salhi, “Provably safe navigation for mobile robots with limited field-of-views in dynamic environments,” *Auton Robot*, vol. 32, no. 3, pp. 267–283, 2012.

[148] K. Macek, D. A. V. Govea, T. Fraichard, and R. Siegwart, “Towards safe vehicle navigation in dynamic urban scenarios,” *Automatika*, 2009.

[149] H. Täubig *et al.*, “Guaranteeing functional safety: Design for provability and computer-aided verification,” *Auton Robot*, vol. 32, no. 3, pp. 303–331, 2012.

[150] M. Rausand and A. Høyland, *System reliability theory: Models, statistical methods, and applications*, 2nd ed. Hoboken, NJ: Wiley-Interscience, 2004.

[151] M. R. Lyu, *Handbook of software reliability engineering*. New York: McGraw Hill, 1996.

[152] C. A. Ericson, *Hazard analysis techniques for system safety*. Hoboken, New Jersey: Wiley, 2005.

[153] O. D. Ditlevsen and H. O. Madsen, *Structural reliability methods*. Chichester, New York: J. Wiley & Sons, 1999, ©1996.

[154] D. Straub, “Lecture Notes in Structural Reliability,” Munich, 2014.

[155] G. Weller and B. Schlag, “Behavioral Aspects of Driver Assistance Systems,” in *Handbook of Driver Assistance Systems*, H. Winner, S. Hakuli, F. Lotz, and C. Singer, Eds., Cham: Springer International Publishing, 2016, pp. 91–107.

[156] M. Berk, O. Schubert, H.-M. Kroll, B. Buschardt, and D. Straub, “Reliability assessment of safety-critical sensor information: Does one need a reference truth?,” *IEEE Transactions on Reliability*, 2019.

[157] M. Berk, O. Schubert, H.-M. Kroll, B. Buschardt, and D. Straub, “Exploiting Redundancy for Reliability Analysis of Sensor Perception in Automated Driving Vehicles,” *Accepted by IEEE Transactions on Intelligent Transportation Systems*, 2019.

[158] S. Hoermann, P. Henzler, M. Bach, and K. Dietmayer, “Object Detection on Dynamic Occupancy Grid Maps Using Deep Learning and Automatic Label Generation,” in *2018 IEEE Intelligent Vehicles Symposium (IV)*, Changshu, 2018, pp. 826–833.

[159] A. Ishimaru, “Wave propagation and scattering in random media and rough surfaces,” *Proc. IEEE*, vol. 79, no. 10, pp. 1359–1366, 1991.

[160] B. Blevis, “Losses due to rain on radomes and antenna reflecting surfaces,” *IEEE Trans. Antennas Propagat.*, vol. 13, no. 1, pp. 175–176, 1965.

[161] M. Berk, H.-M. Kroll, O. Schubert, B. Buschardt, and D. Straub, “Zuverlässigkeitsanalyse umfelderfassender Sensorik: Eine stochastische Methodik zur Berücksichtigung von Umgebungseinflüssen am Beispiel von LiDAR Sensoren,” in *Fahrerassistenz und automatisiertes Fahren: 32. VDI-VW-Gemeinschaftstagung*, Wolfsburg, 2016, pp. 455–475.

[162] S. Yoo and M. Harman, “Regression testing minimization, selection and prioritization: A survey,” *Software Testing, Verification and Reliability*, vol. 22, no. 2, pp. 67–120, 2012.

[163] R. S. Arnold and S. Bohner, *Software Change Impact Analysis*. Los Alamitos, CA, USA: IEEE Computer Society Press, 1996.

[164] B. Li, X. Sun, H. Leung, and S. Zhang, “A survey of code-based change impact analysis techniques,” *Software Testing, Verification and Reliability*, vol. 23, no. 8, pp. 613–646, 2013.

[165] A. D. Lucia, F. Fasano, and R. Oliveto, “Traceability management for impact analysis,” in *2008 Frontiers of Software Maintenance*, 2008, pp. 21–30.

[166] *Functional Safety of Electrical/Electonic/ Programmable Electronic Safety-Related Systems*, IEC 61508:2010, 2010.

[167] *Road vehicles - Functional safety - Part 3: Concept phase*, ISO 26262-3:2011(E), 2011.

[168] S. Kemmann and M. Trapp, “SAHARA -A Systematic Approach for Hazard Analysis and Risk Assessment,” in *SAE 2011 World Congress & Exhibition: SAE Technical Paper 2011-01-1003*, 2011.

[169] T. Ständer, “Eine modellbasierte Methode zur Objektivierung der Risikoanalyse nach ISO 26262,” Dissertation, Fakultät für Maschinenbau, Technische Universität Carolo-Wilhelmina zu Braunschweig, Braunschweig, 2010.

[170] S. Ebel, U. Wilhelm, A. Grimm, and U. Sailer, “Ganzheitliche Absicherung von Fahrerassistenzsystemen in Anlehnung an ISO 26262,” in *Fahrerassistenz und Integrierte Sicherheit: 26. VDI/VW-Gemeinschaftstagung*, Wolfsburg, 2010, pp. 393–405.

[171] E. M. Marszal, “Tolerable risk guidelines,” *ISA Transactions*, vol. 40, no. 4, pp. 391–399, http://www.sciencedirect.com/science/article/pii/S0019057801000118, 2001.

[172] *Road vehicles — Functional safety — Part 4: Product development at the system level*, ISO 26262-4:2011(E), 2011.

[173] *Road vehicles — Functional safety — Part 2: Management of functional safety*, ISO 26262-2:2011(E), 2011.

[174] *Road vehicles — Functional safety — Part 6: Product development at the software level*, ISO 26262-6:2011(E), 2011.

[175] *Road vehicles — Functional safety — Part 9: Automotive Safety Integrity Level (ASIL)-oriented and safety-oriented analyses*, ISO 26262-9:2011(E), 2011.

[176] H. Versmodl and T. Gleissner, "Einfluss des Technologiewandels auf die zukünftige Gestaltung von Fahrzeugelektronik und Systemarchitekturen," in Aachen, 2004.

[177] A. Neukum, E. Mehrjerdian, R. Greul, and A. Gaedke, "Einflussfaktor Fahrzeug - Zur Übertragbarkeit von Aussagen über die Wirkung von Zusatzlenkmomenten," in *Fahrerassistenz und Integrierte Sicherheit: 26. VDI/VW-Gemeinschaftstagung*, Wolfsburg, 2010.

[178] S. Geyer *et al.*, "Concept and development of a unified ontology for generating test and use-case catalogues for assisted and automated vehicle guidance," *IET Intelligent Transport Systems*, vol. 8, no. 3, pp. 183–189, 2014.

[179] C. Domsch and H. Negele, "Einsatz von Referenzfahrsituation bei der Entwicklung von Fahrerassistenzsystemen," in *3. Tagung Aktive Sicherheit durch Fahrerassistenz*, 2008.

[180] G. Bagschik, T. Menzel, A. Reschka, and M. Maurer, "Szenarien für Entwicklung, Absicherung und Test von automatisierten Fahrzeugen," in *11. Workshop Fahrerassistenzsysteme und automatisiertes Fahren*, Walting, 2017.

[181] A. Pütz, A. Zlocki, and L. Eckstein, "Absicherung hochautomatisierter Fahrfunktionen mithilfe einer Datenbank relevanter Szenarien," in *11. Workshop Fahrerassistenzsysteme und automatisiertes Fahren*, Walting, 2017.

[182] S. Ulbrich, T. Menzel, A. Reschka, F. Schuldt, and M. Maurer, "Defining and Substantiating the Terms Scene, Situation, and Scenario for Automated Driving," in *2015 IEEE 18th International Conference on Intelligent Transportation Systems*, Gran Canaria, Spain, 2015, pp. 982–988.

[183] B. Schick, J. Henning, U. Wurster, and B. Klein-Ridder, "Simulationsmethoden zur Evaluierung und Verifizierung von Funktion, Güte und Sicherheit von Fahrerassistenzsystemen im durchgängigen MIL-, SIL-und HIL-Prozess," in *3. Tagung Aktive Sicherheit durch Fahrerassistenz*, Garching, 2008.

[184] F. Schuldt, F. Saust, B. Lichte, M. Maurer, and S. Scholz, "Effiziente systematische Testgenerierung für Fahrerassistenzsysteme in virtuellen Umgebungen," *AAET2013-Automatisierungssysteme, Assistenzsysteme und eingebettete Systeme für Transportmittel*, 2013.

[185] M. Horstmann, “Verflechtung von Test und Entwurf für eine verlässliche Entwicklung eingebetteter Systeme im Automobilbereich,” Disseratation, Fakultät für Maschinenbau, Technische Universität Carolo-Wilhelmina zu Braunschweig, Braunschweig, 2005.

[186] L. Eckstein and A. Zlocki, “Safety potential of ADAS-Combined methods for an effective evaluation,” in *23rd International Technical Conference on the Enhanced Safety of Vehicles (ESV)*, Seoul, South Korea, 2013.

[187] F. Schuldt, T. Menzel, and M. Maurer, “Eine Methode für die Zuordnung von Testfällen für automatisierte Fahrfunktionen auf X-in-the-Loop Verfahren im modularen virtuellen Testbaukasten,” in *10. Workshop Fahrerassistenzsysteme: FAS 2015*, Walting, 2015, pp. 171–182.

[188] F. Schuldt, B. Lichte, M. Maurer, and S. Scholz, “Systematische Auswertung von Testfällen für Fahrfunktionen im modularen virtuellen Testbaukasten,” in *9. Workshop Fahrerassistenzsysteme: FAS2014*, Walting, 2014, pp. 169–178.

[189] R. Bellman, R. E. Bellman, and K. M. R. Collection, *Adaptive Control Processes: A Guided Tour*: Princeton University Press, 1961.

[190] H.-P. Schöner and W. Hurich, “Testing with Coordinated Automated Vehicles,” in *Handbook of Driver Assistance Systems*, H. Winner, S. Hakuli, F. Lotz, and C. Singer, Eds., Cham: Springer International Publishing, 2016, pp. 261–276.

[191] S. Hakuli and M. Krug, “Virtual Integration in the Development Process of ADAS,” in *Handbook of Driver Assistance Systems*, H. Winner, S. Hakuli, F. Lotz, and C. Singer, Eds., Cham: Springer International Publishing, 2016, pp. 159–176.

[192] M. Martinus, M. Deicke, and M. Folie, “Virtual Test Driving Hardware-Independent Integration of Series Software,” *ATZelektronik worldwide*, pp. 16–21, 2013.

[193] M. Nentwig and M. Stamminger, “Hardware-in-the-loop testing of computer vision based driver assistance systems,” in *2011 IEEE Intelligent Vehicles Symposium (IV)*, Baden-Baden, Germany, 2011, pp. 339–344.

[194] G. Berg, V. Nitsch, and B. Färber, “Vehicle in the Loop,” in *Handbook of Driver Assistance Systems*, H. Winner, S. Hakuli, F. Lotz, and C. Singer, Eds., Cham: Springer International Publishing, 2016, pp. 199–210.

[195] B. Strasser, “Vernetzung von Test- und Simulationsmethoden für die Entwicklung von Fahrerassistenzsystemen,” Dissertation, Technische Universität München, München, 2012.

[196] T. Bock, M. Maurer, and G. Farber, “Validation of the Vehicle in the Loop (VIL); A milestone for the simulation of driver assistance systems,” in *2007 IEEE Intelligent Vehicles Symposium*, Istanbul, Turkey, 2007, pp. 612–617.

[197] E. Roth *et al.*, “Analyse und Validierung vorausschauender Sensormodelle in einer integrierten Fahrzeug- und Umfeldsimulation,” in *Fahrerassistenz und Integrierte Sicherheit: 26. VDI/VW-Gemeinschaftstagung*, Wolfsburg, 2010, pp. 153–173.

[198] T. Bock, “Vehicle in the Loop: Test- und Simulationsumgebung für Fahrerassistenzsysteme,” Dissertation, Chair of Real-Time Computer Systems, Technische Universität München, München, 2008.

[199] F. Schmidt, “Funktionale Absicherung kamerabasierter Aktiver Fahrerassistenzsysteme durch Hardware-in the-Loop-Tests,” Dissertation, Fachbereich Informatik, Universität Kaiserslautern, Kaiserslautern, 2012.

[200] K. von Neumann-Cosel, “Virtual Test Drive: Simulation umfeldbasierter Fahrzeugfunktionen,” Dissertation, Lehrstuhl für Echzeitsyteme und Robotik, Technische Universität München, München, 2012.

[201] W. Wachenfeld and H. Winner, “Virtual Assessment of Automation in Field Operation A New Runtime Validation Method,” in *10. Workshop Fahrerassistenzsysteme: FAS 2015*, Walting, 2015, pp. 161–170.

[202] N. Fecher, J. Hoffman, and H. Winner, “Evaluation Concept EVITA,” in *Handbook of Driver Assistance Systems*, H. Winner, S. Hakuli, F. Lotz, and C. Singer, Eds., Cham: Springer International Publishing, 2016, pp. 249–260.

[203] M. Benmimoun, M. Ljung Aust, F. Faber, G. Saint Pierre, and A. Zlocki, “Safety analysis method for assessing the impacts of advanced driver assistance systems within the European large scale field test euroFOT,” in *8th ITS European Congress*, 2011.

[204] V. L. Neale, T. A. Dingus, S. G. Klauer, J. Sudweeks, and M. Goodman, “An overview of the 100-car naturalistic study and findings,” *National Highway Traffic Safety Administration, Paper*, vol. 5, p. 400, 2005.

[205] R. Eenink, Y. Barnard, M. Baumann, X. Augros, and F. Utesch, “UDRIVE: The European naturalistic driving study,” in *Proceedings of Transport Research Arena*, 2014.

[206] W. Peterson, T. Birdsall, and W. Fox, “The theory of signal detectability,” *Transactions of the IRE professional group on information theory*, vol. 4, no. 4, pp. 171–212, 1954.

[207] J. A. Swets, *Signal detection theory and ROC analysis in psychology and diagnostics: Collected papers*: Psychology Press, 2014.

[208] T. Fawcett, “An introduction to ROC analysis,” *Pattern Recognition Letters*, vol. 27, no. 8, pp. 861–874, 2006.

[209] M. Sättele, M. Bründl, and D. Straub, “Reliability and effectiveness of early warning systems for natural hazards: Concept and application to debris flow warning,” *Reliability Engineering & System Safety*, vol. 142, pp. 192–202, 2015.

[210] K.-H. Lee and R. Ehsani, “Comparison of two 2D laser scanners for sensing object distances, shapes, and surface patterns,” *Computers and Electronics in Agriculture*, vol. 60, no. 2, pp. 250–262, 2008.

[211] L. Kneip, F. Tache, G. Caprari, and R. Siegwart, “Characterization of the compact Hokuyo URG-04LX 2D laser range scanner,” in *2009 IEEE International Conference on Robotics and Automation (ICRA)*, Kobe, 2009, pp. 1447–1454.

[212] C. Ye and J. Borenstein, “Characterization of a 2D laser scanner for mobile robot obstacle negotiation,” in *2002 IEEE International Conference on Robotics and Automation*, Washington, DC, USA, 2002, pp. 2512–2518.

[213] S. Coles, *An Introduction to Statistical Modeling of Extreme Values*. London: Springer London : Imprint: Springer, 2001.

[214] M. Berk, O. Schubert, H.-M. Kroll, B. Buschardt, and D. Straub, “Absicherung der Umfeldwahrnehmung von hoch- und vollautomatisierten Fahrzeugen,” in *Fahrerassistenzsysteme und automatisiertes Fahren: 34. VDI/VW-Gemeinschaftstagung*, Wolfsburg, 2018, pp. 165–184.

[215] F. Bock, S. Siegl, and R. German, “Mathematical Test Effort Estimation for Dependability Assessment of Sensor-based Driver Assistance Systems,” in *2016 42st Euromicro Conference on Software Engineering and Advanced Applications (SEAA)*, Limassol, Cyprus.

[216] M. Hisakado, K. Kitsukawa, and S. Mori, “Correlated binomial models and correlation structures,” *Journal of Physics A: Mathematical and General*, vol. 39, no. 50, p. 15365, 2006.

[217] S. R. Paul, “Applications of the beta distribution,” in *Handbook of beta distribution and its applications*, A. K. Gupta and S. Nadarajah, Eds.: CRC Press, 2004, pp. 423–436.

[218] T. J. Maharry, “Proportion Differences Using The Beta-Binomial Distribution,” Dissertation, Graduate College, Oklahoma State University, Stillwater, Oklahoma, 2006.

[219] B. Rosner, “Beta-Binomial Distribution,” in *Encyclopedia of Biostatistics*, P. Armitage and T. Colton, Eds., Chichester, UK: John Wiley & Sons, Ltd, 2005.

[220] W. Gutjahr, “Reliability optimization of redundant software with correlated failures,” in *Ninth International Symposium on Software Reliability Engineering*, Paderborn, Germany, Nov. 1998, pp. 293–302.

[221] V. F. Nicola and A. Goyal, “Modeling of correlated failures and community error recovery in multiversion software,” *IEEE Trans. Software Eng. (IEEE Transactions on Software Engineering)*, vol. 16, no. 3, pp. 350–359, 1990.

[222] P. Hokstad, “A shock model for common-cause failures,” *Reliability Engineering & System Safety*, vol. 23, no. 2, pp. 127–145, 1988.

[223] S. Greenland *et al.*, “Statistical tests, P values, confidence intervals, and power: a guide to misinterpretations,” (eng), *European journal of epidemiology*, vol. 31, no. 4, pp. 337–350, 2016.

[224] J. Cohen, “The earth is round (p < .05),” *American Psychologist*, vol. 49, no. 12, pp. 997–1003, 1994.

[225] R. Nuzzo, “Scientific method: statistical errors,” *Nature*, vol. 506, pp. 150–152, 2014.

[226] M. Baker, “Statisticians issue warning over misuse of P values,” *Nature*, vol. 531, p. 151, 2016.

[227] R. L. Wasserstein and N. A. Lazar, “The ASA's Statement on p -Values: Context, Process, and Purpose,” *The American Statistician*, vol. 70, no. 2, pp. 129–133, 2016.

[228] J. VanderPlas, “Frequentism and bayesianism: a python-driven primer,” *arXiv preprint arXiv:1411.5018*, 2014.

[229] E. T. Jaynes and O. Kempthorne, “Confidence intervals vs Bayesian intervals,” in *Foundations of probability theory, statistical inference, and statistical theories of science*: Springer, 1976, pp. 175–257.

[230] A. Gelman, “Induction and deduction in Bayesian data analysis,” *Rationality, Markets and Morals*, vol. 2, pp. 67–78, 2011.

[231] R. D. Morey, R. Hoekstra, J. N. Rouder, M. D. Lee, and E.-J. Wagenmakers, “The fallacy of placing confidence in confidence intervals,” *Psychonomic bulletin & review*, vol. 23, no. 1, pp. 103–123, 2016.

[232] M. S. Hamada, A. G. Wilson, C. S. Reese, and H. F. Martz, *Bayesian Reliability*, 1st ed. s.l.: Springer-Verlag, 2008.

[233] M. Fitzgerald, H. F. Martz, and R. L. Parker, “Bayesian Single-Level Binomial And Exponential Reliability Demonstration Test Plans,” *Int. J. Rel. Qual. Saf. Eng.*, vol. 06, no. 02, pp. 123–137, 1999.

[234] H. Singh, V. Cortellessa, B. Cukic, E. Gunel, and V. Bharadwaj, “A Bayesian approach to reliability prediction and assessment of component based systems,” in *12th International Symposium on Software Reliability Engineering. ISSRE 2001*, Hong Kong, China, Nov. 2001, pp. 12–21.

[235] D. M. Brender, “The Bayesian Assessment of System Availability: Advanced Applications and Techniques,” *IEEE Trans. Rel.*, vol. R-17, no. 3, pp. 138–147, 1968.

[236] H. F. Martz and R. A. Waller, “Bayesian reliability analysis of complex series/parallel systems of binomial subsystems and components,” *Technometrics*, vol. 32, no. 4, pp. 407–416, 1990.

[237] M. Guida and G. Pulcini, “Automotive reliability inference based on past data and technical knowledge,” *Reliability Engineering & System Safety*, vol. 76, no. 2, pp. 129–137, 2002.

[238] F. P. A. Coolen and P. Coolen-Schrijner, “On Zero-Failure Testing for Bayesian High-Reliability Demonstration,” *Proceedings of the Institution of Mechanical Engineers, Part O: Journal of Risk and Reliability*, vol. 220, no. 1, pp. 35–44, 2006.

[239] K. W. Miller *et al.*, “Estimating the probability of failure when testing reveals no failures,” *IEEE Trans. Software Eng. (IEEE Transactions on Software Engineering)*, vol. 18, no. 1, pp. 33–43, 1992.

[240] H. F. Martz and R. A. Waller, *Bayesian reliability analysis*. New York NY u.a.: Wiley, 1982.

[241] D. V. Mastran and N. D. Singpurwalla, “A Bayesian Estimation of the Reliability of Coherent Structures,” *Operations Research*, vol. 26, no. 4, pp. 663–672, 1978.

[242] D. Straub, “Lecture Notes in Engineering Risk Analysis,” München, Apr. 2017.

[243] R. E. Kass and L. Wasserman, “The selection of prior distributions by formal rules,” *Journal of the American Statistical Association*, vol. 91, no. 435, pp. 1343–1370, 1996.

[244] H. Jeffreys, *Theory of Probability*, 3rd ed. London: Oxford University press, 1961.

[245] H. Raiffa and R. Schlaifer, *Applied statistical decision theory*, 6th ed. Boston: Div. of Research Graduate School of Business Administration Harvard Univ, 1974.

[246] A. N. Sanborn and T. T. Hills, “The frequentist implications of optional stopping on Bayesian hypothesis tests,” (eng), *Psychonomic bulletin & review*, vol. 21, no. 2, pp. 283–300, 2014.

[247] *International Electrotechnical Vocabulary (IEV) - Chapter 191 - Dependability and Quality of Service*, 1990.

[248] N. Hirsenkorn *et al.*, “Learning Sensor Models for Virtual Test and Development,” in *11. Workshop Fahrerassistenzsysteme und automatisiertes Fahren*, Walting, 2017, pp. 115–124.

[249] T. Hanke *et al.*, “Generic architecture for simulation of ADAS sensors,” in *2015 16th International Radar Symposium (IRS)*, Dresden, Germany, 2015, pp. 125–130.

[250] S. Wang, S. Heinrich, M. Wang, and R. Rojas, “Shader-based sensor simulation for autonomous car testing,” in *2012 15th International IEEE Conference on Intelligent Transportation Systems*, 2012, pp. 224–229.

[251] M. Weiskopf, C. Wohlfahrt, and A. Schmidt, “Integrationslösung zur Absicherung eines realen Radarsensors im Systemverbund mit der Hardware-in-the-Loop Testtechnologie,” *Automotive-Safety & Security 2014 (2015)*, 2015.

[252] M. Pharr, W. Jakob, and G. Humphreys, *Physically based rendering: From theory to implementation*: Morgan Kaufmann, 2016.

[253] M. E. O’Brien and D. G. Fouche, “Simulation of 3D laser radar systems,” *Lincoln Laboratory Journal*, vol. 15, no. 1, pp. 37–60, 2005.

[254] G. Mie, “Beiträge zur Optik trüber Medien, speziell kolloidaler Metallösungen,” *Annalen der physik*, vol. 330, no. 3, pp. 377–445, 1908.

[255] C. F. Bohren and D. R. Huffman, *Absorption and scattering of light by small particles*: Wiley, 1983.

[256] M. Kerker and E. M. Loebl, *The Scattering of Light and Other Electromagnetic Radiation: Physical Chemistry: A Series of Monographs*. Burlington: Elsevier Science, 1969.

[257] F. Träger, *Springer handbook of lasers and optics: With 136 tables*, 2nd ed. New York, NY: Springer, 2012.

[258] M. I. Mishchenko, L. D. Travis, and A. A. Lacis, *Scattering, absorption, and emission of light by small particles*, 2002.

[259] C. Mätzler, "MATLAB functions for Mie scattering and absorption, version 2," University of Bern: Institute of Applied Physics, Bern, Aug. 2002.

[260] J. Joss and A. Waldvogel, "Raindrop Size Distribution and Sampling Size Errors," *J. Atmos. Sci.*, vol. 26, no. 3, pp. 566–569, 1969.

[261] J. S. Marshall and W. M. K. Palmer, "The Distribution of Raindrops With Size," *Journal of Meteorology*, vol. 5, no. 4, pp. 165–166, 1948.

[262] C. W. Ulbrich, "Natural Variations in the Analytical Form of the Raindrop Size Distribution," *J. Climate Appl. Meteor.*, vol. 22, no. 10, pp. 1764–1775, 1983.

[263] B. E. Sheppard and P. I. Joe, "Comparison of Raindrop Size Distribution Measurements by a Joss-Waldvogel Disdrometer, a PMS 2DG Spectrometer, and a POSS Doppler Radar," *J. Atmos. Oceanic Technol.*, vol. 11, no. 4, pp. 874–887, 1994.

[264] J. S. Marshall and W. M. K. Palmer, "THE DISTRIBUTION OF RAINDROPS WITH SIZE," *Journal of Meteorology*, vol. 5, no. 4, pp. 165–166, 1948.

[265] N. T. Kottegoda and R. Rosso, *Applied Statistics for Civil and Environmental Engineers*, 2nd ed. Hoboken: John Wiley & Sons, Ltd, 2009.

[266] S. Brooks, "Markov chain Monte Carlo method and its application," *J Royal Statistical Soc D*, vol. 47, no. 1, pp. 69–100, 1998.

[267] S. Brooks, Ed., *Handbook of Markov chain Monte Carlo*. Boca Raton Fla. u.a.: CRC Press, 2011.

[268] H. Barth, "LiDAR Technology for Active Safety and Automated Driving," in *3. VDI-Konferenz - Automatisiertes Fahren*, Düsseldorf, 2016.

[269] D. Lunn, D. Spiegelhalter, A. Thomas, and N. Best, "The BUGS project: Evolution, critique and future directions," *Statistics in medicine*, vol. 28, no. 25, pp. 3049–3067, 2009.

[270] A. K. Nikoloulopoulos, "Copula-Based Models for Multivariate Discrete Response Data," in *Lecture Notes in Statistics, Copulae in Mathematical and Quantitative Finance*, P. Jaworski, F. Durante, and W. K. Härdle, Eds., Berlin, Heidelberg: Springer Berlin Heidelberg, 2013, pp. 231–249.

[271] H. Joe, *Multivariate models and dependence concepts*, 1st ed. Boca Raton, Fla.: Chapman & Hall/CRC, 2001.

[272] P. Jaworski, F. Durante, and W. K. Härdle, Eds., *Copulae in Mathematical and Quantitative Finance*. Berlin, Heidelberg: Springer Berlin Heidelberg, 2013.

[273] A. Talhouk, A. Doucet, and K. Murphy, "Efficient Bayesian Inference for Multivariate Probit Models With Sparse Inverse Correlation Matrices," *Journal of Computational and Graphical Statistics*, vol. 21, no. 3, pp. 739–757, 2012.

[274] R. E. Melchers, *Structural reliability: Analysis and prediction*, 2nd ed. Chichester u.a.: John Wiley, 1999.

[275] J. Song and W.-H. Kang, “System reliability and sensitivity under statistical dependence by matrix-based system reliability method,” *Structural Safety*, vol. 31, no. 2, pp. 148–156, 2009.

[276] C. W. Dunnett and M. Sobel, “Approximations to the Probability Integral and Certain Percentage Points of a Multivariate Analogue of Student's t-Distribution,” *Biometrika*, vol. 42, no. 1/2, p. 258, 1955.

[277] A. Koenig, M. Gutbrod, S. Hohmann, and J. Ludwig, “Bridging the Gap between Open Loop Tests and Statistical Validation for Highly Automated Driving,” *SAE Int. J. Trans. Safety*, vol. 5, pp. 81–87, https://doi.org/10.4271/2017-01-1403, 2017.

[278] B. Hoye, D. Lambert, and G. Sutton, “Autonomous driving comparison and evaluation,” US9650051 B2, May 16, 2017.

[279] H. Winner, “Einrichtung zum Bereitstellen von Signalen in einem Kraftfahrzeug,” DE10102771A1, Germany.

[280] Tesla, *Upgrading Autopilot: Seeing the World in Radar*. [Online] Available: https://www.tesla.com/de_DE/blog/upgrading-autopilot-seeing-world-radar?redirect=no. Accessed on: Sep. 25 2017.

[281] J.-H. Yan and M. Mazumdar, “A Comparison of Several Component-Testing Plans for a Series System,” *IEEE Trans. Rel.*, vol. 35, no. 4, pp. 437–443, 1986.

[282] J.-H. Yan and M. Mazumdar, “A Comparison of Several Component-Testing Plans For A Parallel System,” *IEEE Trans. Rel.*, vol. R-36, no. 4, pp. 419–424, 1987.

[283] J.-H. Yan and M. Mazumdar, “A Component-Testing Procedure For A Parallel System With Type II Censoring,” *IEEE Trans. Rel.*, vol. R-36, no. 4, pp. 425–428, 1987.

[284] A. J. Fernández, “Optimal Reliability Demonstration Test Plans for k-out-of-n Systems of Gamma Distributed Components,” *IEEE Trans. Rel.*, vol. 60, no. 4, pp. 833–844, 2011.

[285] R. G. Easterling, M. Mazumdar, F. W. Spencer, and K. V. Diegert, “System-Based Component-Test Plans and Operating Characteristics: Binomial Data,” *Technometrics*, vol. 33, no. 3, pp. 287–298, 1991.

[286] M. Mazumdar, “An Optimum Procedure for Component Testing in the Demonstration of Series System Reliability,” *IEEE Trans. Rel.*, vol. R-26, no. 5, pp. 342–345, 1977.

[287] J. Rajgopal and M. Mazumdar, “A type-II censored, log test time based, component-testing procedure for a parallel system,” *IEEE Trans. Rel.*, vol. 37, no. 4, pp. 406–412, 1988.

[288] S. Gal, “Optimal Test Design for Reliability Demonstration,” *Operations Research*, vol. 22, no. 6, pp. 1236–1242, 1974.

[289] P. Hokstad and M. Rausand, “Common Cause Failure Modeling: Status and Trends,” in *Handbook of performability engineering*, K. B. Misra, Ed., London: Springer, 2008, pp. 621–640.

[290] E. L. Melnick and B. S. Everitt, *Encyclopedia of quantitative risk analysis and assessment*. Chichester: Wiley, 2008.

[291] K. N. Fleming, "Reliability model for common mode failures in redundant safety systems," General Atomic Co, 1974.

[292] W. E. Vesely, "Estimating common cause failure probabilities in reliability and risk analysis: Marshall-Olkin specializations," *Nuclear systems reliability engineering and risk assessment*, pp. 314–341, 1977.

[293] C. L. Atwood, "The Binomial Failure Rate Common Cause Model," *Technometrics*, vol. 28, no. 2, p. 139, 1986.

[294] M. Stephens, "Dealing with label switching in mixture models," *J Royal Statistical Soc B*, vol. 62, no. 4, pp. 795–809, 2000.

[295] A. P. Dempster, N. M. Laird, and D. B. Rubin, "Maximum Likelihood from Incomplete Data via the EM Algorithm," *Journal of the Royal Statistical Society. Series B (Methodological)*, vol. 39, no. 1, pp. 1–38, http://www.jstor.org/stable/2984875, 1977.

[296] R. M. Neal, "MCMC Using Hamiltonian Dynamics," in *Chapman & Hall/CRC handbooks of modern statistical methods, Handbook of Markov chain Monte Carlo*, S. Brooks, Ed., Boca Raton Fla. u.a.: CRC Press, 2011, pp. 113–162.