Statistical modelling of algorithms for signal processing in systems based on environment perception

zur Erlangung des akademischen Grades eines

Doktors der Ingenieurwissenschaften

von der KIT-Fakultät für Informatik

des Karlsruher Instituts für Technologie (KIT)

genehmigte

DISSERTATION

von

Jan Erik Stellet

aus Braunschweig

Tag der mündlichen Prüfung: 09.06.2016
Erster Gutachter: Prof. Dr.-Ing. J. Marius Zöllner
Zweiter Gutachter: Prof. Dr.-Ing. habil. Jürgen Beyerer

Document template provided by:

Institute of Industrial Information Technology (IIIT)
Karlsruhe Institute of Technology (KIT)
Hertzstraße 16
76187 Karlsruhe

Abstract

Advanced driver assistance systems (ADAS), for example autonomous emergency brake (AEB) systems, aim to sense a vehicle's environment, understand the current traffic situation and react appropriately in order to support the human driver. One cornerstone for reliable systems is an appropriate handling of uncertainties. Uncertainties arise for instance due to noisy sensor measurements or the unknown future evolution of a traffic situation.

This work's objective is to contribute to the understanding of these uncertainties. To this end, parametric probability distributions are used to analytically model and propagate uncertainties at individual parts of ADAS signal processing chains. Previous works approach this mostly with numerical simulations. An inherent drawback is that only a concrete implementation of algorithms can be analysed. Analytical modelling on the other hand, as pursued in this thesis, allows drawing more generic conclusions. One can for example attempt to derive upper performance bounds which apply to any implementation of (sub-) optimal solutions.

This thesis devises probabilistic models of uncertainty in selected algorithms with a distinct role in current and future ADAS. The models are applied to the derivation of sensor parameter constraints for feature-based localisation for urban automated driving and the analysis of performance limitations in AEB systems.

First, environment perception tasks are studied. This comprises the detection of obstacles with a stereo vision sensor and feature-based localisation. Instead of a theoretical top-down analysis, an environment sensor's inaccuracy can also be empirically evaluated by comparison to a reference sensor. Advanced off-line approaches can be leveraged to obtain accurate reference data from raw sensor measurements. The potential of such methods, in contrast to an on-line processing under real-time constraints, is studied on the example of a laser scanner sensor.

Second, the models of errors in individual measurements are propagated to the uncertainty in estimates of unobservable dynamic states, for instance motion state variables.

Third, the prediction and risk assessment of traffic situations is considered. Novel expressions for performance bounds on the recognition of semantic driver intentions for long-term motion predictions are derived. Moreover, parametric models of uncertainty in kinematic motion models for short-term predictions are estimated from empirical driving data. Based on these predictions, the risk of an imminent collision can be quantified in terms of criticality measures. A comprehensive approach to probabilistic modelling of these risk metrics is developed. Performance bounds on the timely activation of an AEB intervention are then derived by uncertainty propagation from perception and state estimation to the criticality assessment.

Zusammenfassung

Umfelderfassende Fahrerassistenzsysteme (FAS), zum Beispiel automatische Notbremssysteme, erfassen und interpretieren das Fahrumfeld, um den Fahrer bei der Fahraufgabe zu unterstützen. Eine wichtige Voraussetzung für ein robustes Systemverhalten ist ein angemessener Umgang mit Unsicherheiten. Unsicherheiten treten beispielsweise aufgrund von Sensormessrauschen sowie des unbekannten zukünftigen Verhaltens von Verkehrsteilnehmern auf.

Das Ziel der vorliegenden Arbeit ist es, zum Verständnis von derartigen Unsicherheiten beizutragen. Hierzu werden parametrische Wahrscheinlichkeitsdichtefunktionen angesetzt, um Unsicherheiten zu modellieren und entlang der Signalverarbeitungskette eines FAS fortzupflanzen. Bisherige Arbeiten verwenden hierfür meist numerische Simulationen. Nachteilig ist dabei, dass lediglich konkrete Implementierungen von Algorithmen untersucht werden können. Analytische Modelle hingegen können zu generischen Erkenntnissen führen. Beispielsweise kann eine theoretische obere Schranke für die Güte einer Lösung direkt aus der Problemformulierung abgeleitet werden.

In dieser Arbeit werden probabilistische Modelle für Unsicherheiten in für heutige und zukünftige FAS relevanten Algorithmen und Aufgabenstellungen entwickelt. Die Anwendung dieser Modelle wird anhand einer Ableitung von Sensorgenauigkeitsanforderungen für eine Landmarken-basierte Lokalisierung sowie einer Analyse von Systemgrenzen in Notbremssystemen demonstriert.

Zuerst werden Algorithmen zur Umfeldwahrnehmung untersucht. Dies beinhaltet die Erkennung von Hindernissen mittels einer Stereo-Video-Kamera sowie die Lokalisierung durch Korrespondenzsuche im statischen Umfeld. Anstelle einer theoretischen Modellbildung kann die Genauigkeit einer Umfeldwahrnehmung auch empirisch durch Vergleich mit Referenzmessungen ermittelt werden. Am Beispiel eines Laserscanners wird in dieser Arbeit untersucht, wie sich die Genauigkeit von Referenzdaten durch akausale Signalverarbeitungsmethoden ohne Echtzeitanforderungen verbessern lässt.

In einem zweiten Schritt werden Unsicherheiten in den Messungen aus individuellen Zeitschritten auf die Genauigkeit eines daraus geschätzten Bewegungszustandes übertragen. Drittens werden Unsicherheiten in der Vorhersage und Risikobewertung von Fahrsituationen untersucht. Hierbei werden neue obere Schranken für die Güte einer Manövererkennung zur Langzeitprädiktion hergeleitet. Parametrische Modelle für Unsicherheiten in kinematischen Fahrzeugbewegungsmodellen für Kurzzeitprädiktionen werden aus empirischen Daten ermittelt. Basierend auf diesen Prädiktionen kann das Risiko einer bevorstehenden Kollision in Form von Kritikalitätsmaßen quantifiziert werden. Eine umfassende Unsicherheitsmodellierung in derartigen Risikometriken wird in dieser Arbeit entwickelt.

Vorwort

Die hier vorliegende Arbeit entstand während meiner Forschungstätigkeit in der Abteilung Fahrzeugsicherheit und Assistenzsysteme (CR/AEV) im Zentralbereich Forschung und Vorausentwicklung der Robert Bosch GmbH. Die folgenden Zeilen möchte ich jenen widmen, die diesen Weg begleitet haben.

Die wissenschaftliche Betreuung und das Hauptreferat wurden von Professor J. Marius Zöllner, Direktor für Technisch-Kognitive Assistenzsysteme (TKS) am Forschungszentrum für Informatik (FZI) in Karlsruhe, übernommen. Für die engagierte Begleitung meiner Arbeit und den Vertrauensvorschuss in dieses Projekt danke ich Dir, Marius, sehr. Die Regelmäßigkeit und Effizienz unserer Diskussionen haben zweifelsohne entscheidend zum Gelingen beigetragen. Professor Jürgen Beyerer, Leiter des Fraunhofer-Instituts für Optronik, Systemtechnik und Bildauswertung (IOSB), danke ich für das Interesse an meiner Arbeit und die freundliche Bereitschaft, das Koreferat zu übernehmen.

Diese Arbeit wäre ohne das Forschungsumfeld der Robert Bosch GmbH nie entstanden. Dr. Wolfgang Branz als Projektleiter und Dr. Jan Schumacher als wissenschaftlicher Betreuer haben hervorragende Rahmenbedingungen für meine Tätigkeit geschaffen und diese mit großem Interesse verfolgt, wofür ich Euch meinen herzlichen Dank aussprechen möchte. In den vergangenen Jahren hast Du, Jan, kontinuierlich den Entstehungsprozess der Arbeit begleitet. Dein naturwissenschaftlicher Blick hat mir bei der Schärfung der Forschungsfrage wertvolle Anregungen geliefert. Mit unermüdlichen Einsatz hast Du gefühlt überabzählbar viele Seiten an Veröffentlichungen aufs Iota Korrektur gelesen und diskutiert, was die Gesamtqualität ohne Zweifel wesentlich gesteigert hat.

Darüber hinaus möchte ich den vielen Menschen danken, die mich in den letzten Jahren fachlich, mental oder mit Humor unterstützt haben. Im Rahmen Ihres Studiums haben Fabian Straub, Leopold Walkling, Patrick Vogt und Marco Hoch an diesem Projekt tatkräftig mitgewirkt. Euch vielen Dank und weiterhin viel Erfolg! Für die gute Zusammenarbeit danke ich Christian Heigele, Francesco Massa Gray, Florian Kuhnt, Markus Mazzola, Jan Rohde, Benjamin Völz und Marc René Zofka. Professor Fernando Puente León danke ich für die Möglichkeit, die ausgefeilten Dokumentenvorlagen des Instituts für Industrielle Informationstechnik (IIIT) verwenden zu dürfen, sowie Thomas Nürnberg dafür, dass er auch bei den kompliziertesten Formatierungsfragen stets eine passende Antwort parat hat. Weiterer Dank richtet sich an die Kollegen in der Abteilung CR/AEV für die motivierende Arbeitsatmosphäre und inspirierenden Diskussionen. Den Mitarbeitern am FZI gilt ebenfalls mein Dank für die freundliche Aufnahme in den Kreis der Abteilung TKS sowie Rat und Tat bei organisatorischen Fragen.

Oft wird der Weg zu einer Dissertation mit dem Erklimmen eines steilen Berges verglichen. Für die Freiheit, diesen Weg einschlagen zu dürfen, für die Ermutigung, nicht stehen zu bleiben und für das Vertrauen, in die richtige Richtung zu gehen, bin ich meinen Eltern, meinen Freunden sowie den mich inspirierenden Menschen zutiefst dankbar.

Stuttgart, im Juli 2016 *Jan Erik Stellet*

Contents

Nomenclature

Symbols

Latin Letters

Symbol	Description
$\mathbf{0}_{n\times m}$	Matrix of zeros with dimensions $n \times m$
$\mathbf{1}_{n\times m}$	Matrix of ones with dimensions $n \times m$
$\operatorname{diag}(a_1, \ldots, a_n)$	Diagonal matrix with diagonal elements $a_1, \ldots, a_n$
$\mathbf{I}_{n\times n}$	Identity matrix with dimensions $n \times n$
$a, b, \ldots$	Scalars (lower-case)
$\mathbf{a}, \mathbf{b}, \ldots$	Column vectors (lower-case, bold upright font)
$\mathbf{A}, \mathbf{B}, \ldots$	Matrices (upper-case, bold upright font)
$\mathcal{A}, \mathcal{B}, \ldots$	Sets (upper-case, calligraphic font)
$\mathtt{a}, \mathtt{b}, \ldots$	Random variables (typewriter font)
$a : b$	Sequence of the integers $a, a+1, \ldots, b \;\; (a \in \mathbb{Z},\; b \in \mathbb{Z},\; a \leq b)$
$\mathbf{a}_{1:k}$	Sequence of the vectors $\mathbf{a}_1, \mathbf{a}_2, \ldots, \mathbf{a}_k$
$\mathbf{A}_{1:k}$	Stacked vectors $\left[\mathbf{a}_1^{\mathrm{T}} \mathbf{a}_2^{\mathrm{T}} \ldots \mathbf{a}_k^{\mathrm{T}}\right]^{\mathrm{T}}$

Greek Letters

Symbol	Description
$\nabla_{\mathbf{x}} f(\mathbf{x})$	Gradient of a scalar-valued function (row vector)
$\Delta_{\mathbf{x}}^{\mathbf{x}} f(\mathbf{x})$	Hessian of a scalar-valued function
$\nabla_{\mathbf{x}} \mathbf{f}(\mathbf{x})$	Jacobian of a vector-valued function

Functions and operators

Operator	Description
$\operatorname{cov}(\cdot)$	Covariance
$\det(\cdot)$	Determinant of a matrix
$\mathbb{E}[\cdot]$	Expectation
$\otimes$	Kronecker product
$\arg\max_x f(x)$	The x that maximises $f(x)$

Operator	Description
$p(\cdot)$	Probability density function
$P(\cdot)$	Cumulative density function
$\underset{x}{\text{sol}}\{f(x) = 0\}$	The x that fulfils $f(x) = 0$
$\text{std}(\cdot)$	Standard deviation
$\text{tr}(\cdot)$	Trace of a matrix
$\text{var}(\cdot)$	Variance

Common abbreviations

Abbreviation	Description
ADAS	Advanced driver assistance system
AEB	Autonomous emergency brake
BN	Bayesian network
BTN	Brake-threat-number
CA	Constant acceleration
CDF	Cumulative distribution function
CRB	Cramér-Rao bound
CTRA	Constant turn rate and acceleration
CV	Constant velocity
DBN	Dynamic Bayesian network
EKF	Extended Kalman filter
EKS	Extended Kalman smoother
EM	Expectation maximisation
FIM	Fisher information matrix
GLR	Generalised likelihood ratio
GLS	Generalised least squares
GMM	Gaussian mixture model
GPS	Global positioning system
HMM	Hidden Markov model
IA	Intrinsic accuracy
IMU	Inertial measurement unit
KF	Kalman filter
LIDAR	Light detection and ranging
LR	Likelihood ratio
LS	Least squares
ML	Maximum likelihood
PDF	Probability density function
ROC	Receiver operating characteristic
TTB	Time-to-brake
TTC	Time-to-collision

1 Introduction

Increasingly, the technical progress in the automotive industry is not only measured in terms of how safe, reliable or efficient but how *intelligent* a vehicle is. This shift of attention can be traced back to numerous driver assistance functions, which have made their way from research labs to series production over the recent years.

According to DONGES [47] and as seen in Fig. 1.1, the driving task can be structured into three levels on different time scales. The longest forecasting horizon is available for navigation, that is planning of a route to a destination through the road network. (Electronic) maps and global positioning aid the driver with this task. In contrast, guiding a vehicle in a dynamic traffic situation involves processing higher amounts of information and shorter reaction times. Advanced driver assistance systems (ADAS) offer support at this level to ensure safe and comfortable travel. Electronic support at the third layer, the stabilisation of the vehicle, is already a mature and widespread technology [191].

Currently, ADAS only support the driver by information and warnings or by autonomous interventions in a limited scope of situations. For instance, an autonomous emergency brake (AEB) is only triggered in the case of an impending collision with other vehicles or pedestrians. Significant research effort is made to realise automated vehicles which relieve humans from the driving task entirely. Research on *intelligent vehicles* strives for more comprehensive environment perception, an accurate interpretation of traffic scenes with a prediction of the future paths of other participants and the planning of safe trajectories.

One key question for realising the aforementioned vision is how to handle uncertainties. Due to a limited field of view of exteroceptive sensors, dynamic occlusions or inevitable measurement noise, uncertainties in the environment perception are unavoidable. Concerning the interpretation of a scene, no simple law describes the intentions of human drivers in ambiguous situations. Therefore, algorithms that predict the future trajectories of other vehicles are likewise affected by inherent uncertainties.

Despite these uncertainties, ADAS are required to act correctly in all cases. For example, triggering an emergency brake manoeuvre introduces potential hazards for the traffic behind. Thus, such an intervention is only justified in the rare case of an imminent collision. Developers are therefore faced with the challenge of either testing that a given assistance system is free of hazardous errors or, during the engineering stage, find a design which guarantees this property.

The objective of this thesis is to contribute to the understanding of uncertainties in ADAS. To this end, probabilistic models of uncertainty in exemplary signal processing chains are developed. These models address the questions of how different kinds of

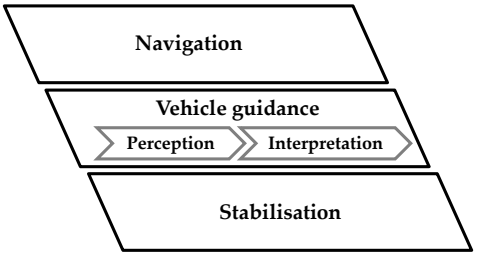


Figure 1.1 Three level model of the driving task [47].

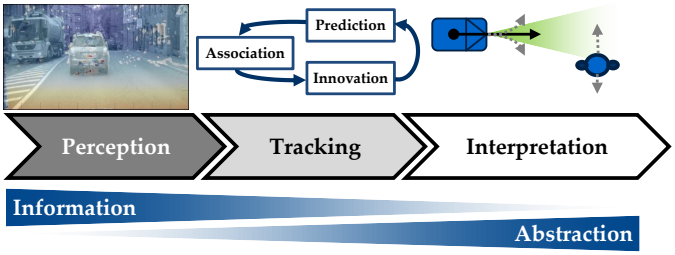


Figure 1.2 ADAS signal processing chain.

uncertainties propagate through the processing chain and how they impact the overall system performance.

The signal processing chains of different ADAS functions are usually not monolithic but – as it is visualised in Fig. 1.2 – comprise generic, reusable algorithms for common estimation and detection tasks. Thus, models derived for these parts can lead to valuable conclusions beyond a particular combination of ADAS function and environment sensor.

It is the hypothesis of this thesis that a deeper understanding of the problem space will lead to a more efficient engineering process. For instance, model-based knowledge can be employed to select appropriate components or to identify infeasible system designs prior to exhaustive testing.

The introductory chapter of this thesis is organised as follows: Firstly, the problem formulation and derived research questions will be detailed in Sec. 1.1. The concept which addresses these aspects is then outlined in Sec. 1.2. Thirdly, an overview of the main contributions will be given in Sec. 1.3 before this chapter concludes with an outline of the remainder of the document in Sec. 1.4.

1.1 Problem statement

Given an algorithm which relies on noisy input data and potentially inaccurate assumptions, the effects of these uncertainties on the algorithm's output data are to be analysed. This question might be addressed by an empirical observation in the field. Due to the abundant traffic situations and other relevant influences, an extensive field test is currently the only known approach to ensure that very rare effects can be discovered [15].

Besides the repeated effort and often a lack of reproducibility that come with empirical observations, a drawback is that no generic conclusions can be drawn beyond the current selection of components, algorithms and parametrisations. For a more structured approach it is therefore desirable to find a *model*, an idealised representation of the relevant influences and their impact.

One objective that is addressed by a model is to explain observations of a phenomenon by a mathematical description of cause and effect. For example, the accuracy of a distance measurement obtained by a stereo vision sensor can be described by a characteristic quadratic dependence of the measurement's standard deviation on the distance.

A second objective is to predict a response under not yet observed conditions, for example for different values of a system parameter. Such predictions are of course subject to uncertainty. Nevertheless, a model might yield sensible initial values or best case requirements before costly experimental evaluations are conducted.

An important characterisation of a model is to which degree it falls into the category of a *purely empirical* or *substantive* model. The latter are characterised by specific aspects of the phenomenon's underlying subject-matter, for example derived from theoretical considerations. On the contrary, purely empirical models are generic and can be applied to describe common patterns in different domains [41]. This thesis focusses primarily on substantive models. It is attempted to find statistical models of specific algorithms in ADAS signal processing chains and to develop generic frameworks which can be applied to other algorithms with a similar purpose.

The majority of the algorithms that are analysed in this thesis can be found in a typical AEB system, for example based on a stereo vision camera. Therefore, statistical models of the individual algorithms can be concatenated in order to draw conclusions on the overall behaviour of a realistic exemplary system.

However, AEB systems have a purely reactive behaviour within a limited scope of situations and thus present only a first step towards functions with a higher degree of automation. One central distinction of automated driving functions is that they continuously guide the vehicle through a dynamic environment. This poses higher requirements on accurate self-localisation and long-term predictions of other traffic participants. Algorithms for these two tasks are analysed in this work as well in order to contribute to the development of automated vehicles.

Due to the complexity of an intelligent vehicle's workspace, simplified representation of real-world phenomena are usually required. Concerning algorithms for ADAS applications, a number of typical challenges arise when attempting to find a statistical model.

1. In estimation and detection tasks, an algorithm's output is often defined implicitly in terms of a cost function. Often, no closed-form solution of the optimisation is available and the eventual result can only be obtained by numerical methods.

2. Tracking and interpreting the behaviour of other traffic participants is usually based not on single observations but on a time series thereof. When considering uncertainties in these input quantities, the combinatorics and possible correlations have to be addressed.

3. Uncertainties often result from inaccurate model assumptions in an algorithm. For instance, to cope with the natural lack of measurement information on the future evolution of a traffic scene, motion models are employed to predict the trajectories of other vehicles. Thus, probabilistic models of the deviation between the true trajectory and the prediction have to be retrieved from empirical data.

The aforementioned aspects lead to the following research questions:

1. Is it possible to derive theoretical performance bounds in order to reveal fundamental limitations of a system?
2. Is it possible to derive inverse models in order to obtain sensor parameter constraints from functional requirements?
3. Which simplifying assumptions on real-world conditions can be introduced in order to obtain insightful models?
4. How to analytically propagate uncertainty in estimates which are implicitly defined and thus rely on numerical methods?
5. How to efficiently model tracking and estimation algorithms which aggregate information over time?
6. How to describe inaccurate model assumptions, for example in vehicle motion models, with an unsupervised, data-based approach?

The concept that is pursued in order to address these questions is outlined in the following section.

1.2 Concept overview

The overall approach and the thesis are organised according to the modularised structure of an ADAS processing chain as seen in Fig. 1.2. It is structured in terms of the major, qualitatively different *tasks*. In a specific implementation, *algorithms* are employed to address these tasks. *Perception* refers to (often spatial) measurements of environmental features and is treated in Ch. 3. *Tracking* and *prediction* additionally introduce the time domain and are considered in Ch. 4. Previous measurements are aggregated and possible future evolutions are estimated. Thirdly, *interpretation* is the abstraction from temporal-spatial estimates to semantic aspects such as intentions or risks. Exemplary algorithms for this purpose are studied in Ch. 5.

Ch. 6 addresses the generation of reference measurements from raw sensor data using off-line signal processing algorithms. Accurate reference measurements are a key requirement for experimental evaluations of a system and model identification based on empirical data.

In a concluding part of the thesis in Ch. 7, it is exemplarily demonstrated how the previously described models can be applied in order to derive performance bounds of an AEB system and sensor accuracy requirements for feature-based self-localisation.

Essentially, parametric models of the uncertain inputs of an algorithm have to be derived and propagated to parametric models of the output quantities. The latter thus describe the inputs of a subsequent algorithm. Therefore, a systematic and comprehensive treatment of the entire signal processing chain can be achieved. Often, multiple algorithms exist that address the same task and differ for example by the use of different

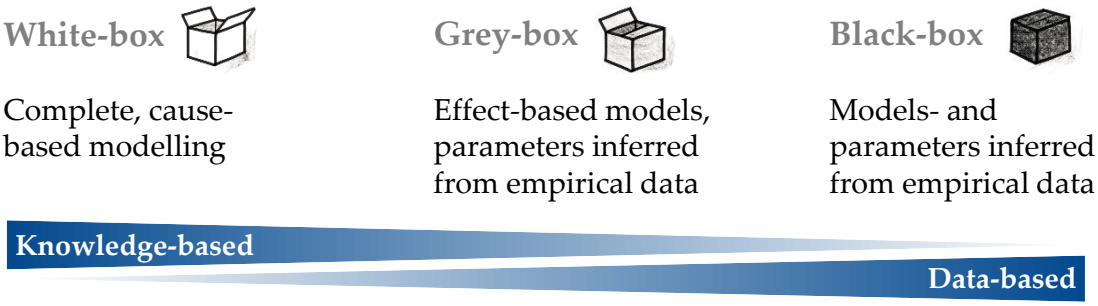


Figure 1.3 Comparison of data- and knowledge-based approaches to modelling.

approximations. To achieve generic models it is therefore advisable to focus on the task itself instead of particular algorithms. That is why it is valuable to derive theoretical performance bounds which apply to any (sub-) optimal algorithm.

In general, two different directions can be followed in order to obtain models of a process. As visualised in Fig. 1.3, *white-box* models are built entirely on knowledge (for example, laws of physics or mathematical relationships) whereas *black-box* models are obtained from empirical observations. An intermediate way is offered by *grey-box* approaches where simplified models of the central characteristics are combined with empirical data.

This work pursues to develop white- or grey-box models in order to achieve generic and insightful results. The knowledge-driven approach can reveal which aspects of an algorithm's output are due to the choice of a specific implementation or parametrisation and which are inherent limitations of the problem itself. Such upper performance bounds allow finding necessary requirements on the input data or the design of objective performance metrics. In contrast, black-box approaches only yield a model of one particular system. Furthermore, they do not leverage the available knowledge about a system in order to alleviate the required amount of empirical data.

However, pure white-box models of common ADAS sensors, for example vision-based, are extremely difficult to obtain and feature a high number of necessary parametrisations [35, 89]. It is thus crucial to assess the accuracy and limitations of simplified models.

Assuming that a probabilistic model of the input quantities of a component has been obtained, the goal is then to propagate the input distribution to the algorithm output. To this end, there are two principal approaches:

1. *Numerical* methods[1] are based on iterated computations for varying inputs. Therefore, input data is randomly sampled from the respective distribution and fed into the algorithm. This yields a non-parametric distribution of the algorithm's output.

2. *Analytical* propagation means to derive a parametric model of the output distribution as a function of a parametric model of the input.

[1] Also termed Monte-Carlo experiments or simulations.

Table 1.1 Comparison of analytical and numerical methods for the propagation of probability distributions. The implementation of numerical simulations is more straightforward but only allows the analysis of a specific algorithm. Analytical approaches yield more generic models but are often limited to certain families of input distributions.

	Analytical	**Numerical**
Approach	Propagate a parametric distribution	Sample from a distribution
Result	▪ Parametric models ▪ Generic performance bounds	▪ Singular evaluation for specific algorithm and parametrisation
Challenges	▪ Effort for derivation ▪ Restrictions imposed by parametric models	▪ Repeated numerical effort

The two approaches feature different strengths and weaknesses which are compared in Tab. 1.1. On the one hand, numerical computations are appealing since they have fewer prerequisites. The drawbacks on the other hand are that only a specific algorithm and singular parametrisation can be evaluated at a time. Therefore, no further insight on the (asymptotic) relationship of parameter values or general performance bounds is obtained. Furthermore, depending on the dimensionality of the input space, the computational burden may become inacceptable.

The strong advantage of an analytical approach is that much more generic conclusions can be drawn. It is possible to take on a problem centric view and thus to derive upper performance bounds which apply to any (sub-) optimal algorithm implementation. Moreover, informative insights on relevant influencing factors, such as system parameters, can be obtained. Finally, analytical results facilitate the implementation of probabilistic models on embedded devices and thus the development of algorithms with a self-assessment of uncertainty. However, analytical approaches are often restricted to certain classes of input densities that allow the propagation through an algorithm. Furthermore, the exact propagation in closed form is often intractable and only approximate models can be derived.

Previous works on the analysis of entire ADAS signal processing chains, for example [73, 111, 143, 170, 183], employ mostly numerical methods. These examples differ by the considered sensors and functions. Using numerical simulations is a viable approach to yield a precise evaluation of an already specified system. However, in order to build a solid understanding for the development of future systems, simplified but analytical models are valuable tools.

In-depth analyses of specific algorithms in ADAS are reported in the literature both with numerical and analytical approaches. Detailed reviews and comparisons to the results from this thesis are given in the respective sections. The conceptual difference is

that this work considers the individual tasks in the context of the entire processing chain. Therefore, models that describe the inputs of an algorithm are consistently motivated. Previous works which only consider one specific algorithm have to substitute such models by assumptions on the input signals.

For example, algorithms for the risk assessment of traffic situations rely on uncertain state estimates from a tracking filter. Previous works which study the impact of these uncertainties often introduce the assumption of a known constant estimation error covariance. The value of this covariance might be chosen as an average or theoretical worst case value. However, more realistic and time-dependent models can be obtained by explicitly modelling the sensor's uncertainty and its propagation to the state estimates, as is pursued in this work.

In summary, this thesis focusses on analytical modelling and uses numerical methods to validate the results. Results and conclusions that were obtained by numerical studies in previous works will be confirmed or extended with mathematical arguments. The main contributions are summarised in the following section.

1.3 Contributions

This thesis contributes to the state of the art in research on intelligent vehicles with a comprehensive analytical modelling of algorithms in an ADAS signal processing chain. A similar end-to-end analysis has to the best of the author's knowledge not been developed before.

Contributions on specific problems can be summarised as follows:

- **Stereo vision based object detection:** A prototypical object detection algorithm is assumed. The influence of measurement noise is studied analytically, the models are evaluated with empirical data and the influencing factors are discussed. The analysis (Sec. 3.1, [223]) contributes to the understanding of uncertainty propagation in stereo vision based algorithms for object detection.

- **Feature-based localisation:** To achieve highly accurate localisation without satellite-based methods, static environmental features can be matched against a known map. The underlying task is to estimate transformation parameters from noisy sets of corresponding feature positions (Procrustes problem). This work (Sec. 3.2.4, [211, 212]) derives novel closed-form expressions of the estimate's variance for the matrix- and scalar-weighted Procrustes problem. Moreover, a novel bias-corrected solution for the scalar-weighted case is proposed (Sec. 3.2.4.3, [217]).

- **Identification of process noise models:** Algorithms for tracking and situation interpretation employ vehicle motion models for short-term trajectory predictions. The prediction's inaccuracy can be modelled in a probabilistic sense as process noise. This work estimates the parameters of Gaussian process noise models from measured vehicle trajectories using the expectation maximisation (EM) algorithm

(Sec. 4.2.5, [221]). A novel closed-form adaptation of the algorithm to a covariance matrix with Kronecker product structure presents a theoretical contribution (Sec. 4.2.4.4).

- **Performance bounds on driver intention recognition:** Long-term predictions of vehicle trajectories can be obtained with a dedicated recognition of semantic driver intentions. This work formulates this task as a hypothesis test and thus a novel framework for the derivation of analytical performance bounds is proposed (Sec. 5.1.3). The development of an equivalent recursive form of the performance bound presents a contribution to change detection problems in general linear dynamic systems (Sec. 5.1.4.4, [219]). In contrast to previous results, the recursive calculation avoids the inversion of large matrices with growing dimensions.

- **Probabilistic models of criticality measures:** Situation interpretation for emergency brake systems can be tailored to the specific use cases of these systems. Thereby, *criticality measures* quantify the risk of an impending collision. A novel method for the propagation of stochastic errors in these algorithms is developed (Sec. 5.2.4, [218]). In contrast to previous works, the approach is fully analytical and takes inherent prediction errors into account. Moreover, a previous study on the activation timing of an AEB system is generalised to the case of correlated errors.

Additional contributions address the testing and validation of higher automated driving functions. Despite the strong progress in the development of automated vehicles, a lack of efficient and standardised methods for approval is regarded as a potential bottleneck for market introduction [15]. One root cause for the necessity of exhaustive field tests is a lack of complete and accurate models of an intelligent vehicle's workspace.

In the course of writing this thesis, a framework for the testing problem has been developed. This enables to assess different approaches and identify open questions in a systematic manner. Relevant publications have been categorised in the proposed framework and discussed in a collaborative survey [220]. This thesis does not entail the survey in its entirety but includes relevant parts to compare own approaches with the state of the art in the respective sections. Furthermore, methods for post processing of laser scanner measurements have been developed (Ch. 6, [215]) which contributes to the complex of generating ground truth data and test scenarios.

1.4 Document outline

The thesis is structured in eight chapters as visualised in Fig. 1.4. Following this introductory chapter, an introduction to probabilistic models is provided in Ch. 2.

Thereafter, Ch. 3, Ch. 4 and Ch. 5 entail the main contents on statistical modelling. Each of these chapters contains two independent sections on specific algorithms.

A third and more application-oriented part is constituted by Ch. 6-Ch. 7. Firstly, the generation of accurate reference measurements of vehicle trajectories is considered in

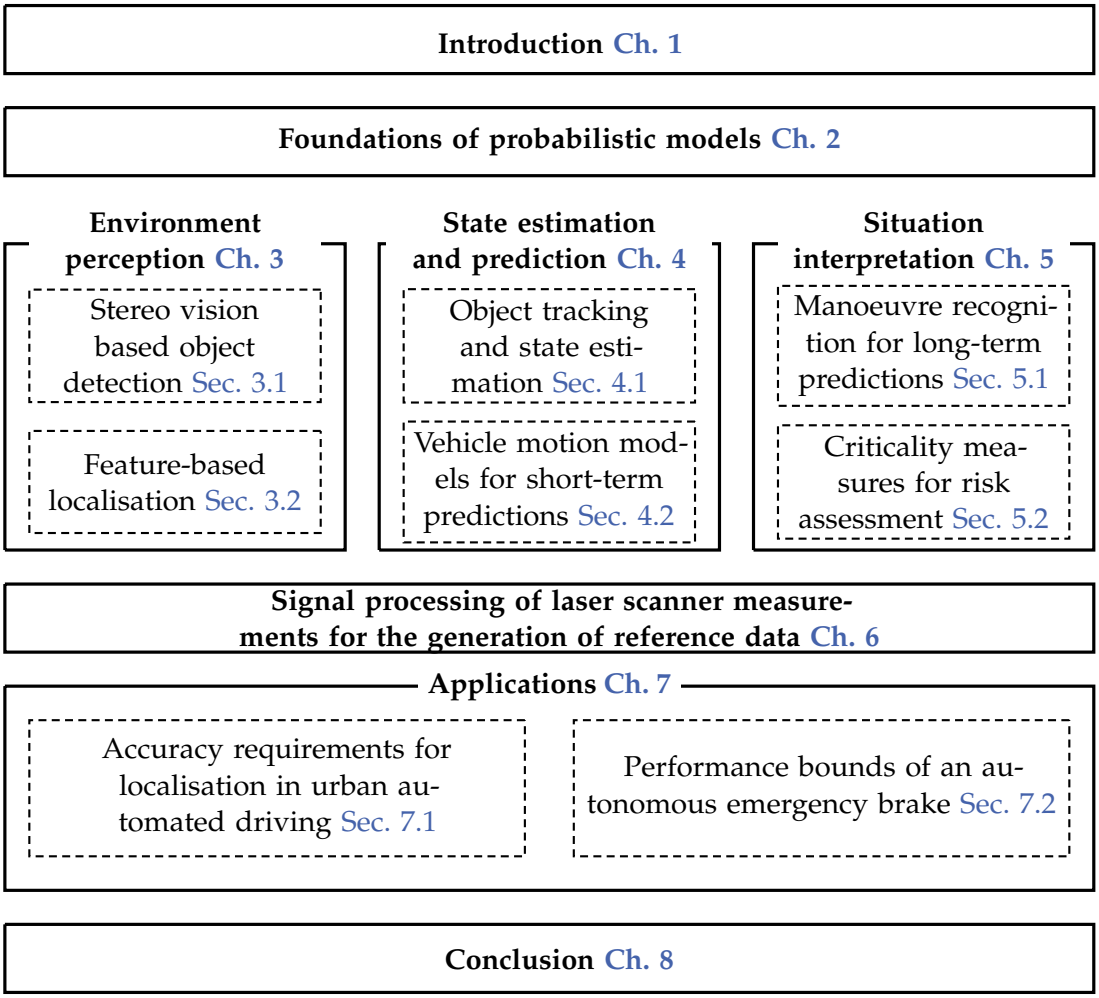


Figure 1.4 Thesis outline.

Ch. 6. This topic is relevant for empirical analyses and testing in general. Secondly, Ch. 7 contains two exemplary applications of the probabilistic models from the main body of the thesis.

Finally, Ch. 8 contains the conclusion and a discussion of possible directions for further research.

2 Foundations of probabilistic models

This chapter introduces basic concepts and notations that are employed throughout the thesis. It consists of three parts. First, foundations of random variables and probability distributions are introduced in Sec. 2.1. Second, the transformation of random variables by non-linear functions will be studied in Sec. 2.2. Third, different interpretations of probabilities, known as Bayesian and frequentist, will be discussed in Sec. 2.3.

2.1 Random variables and probability distributions

Phenomena with an unknown exact but *random* behaviour are modelled with *random variables* in this thesis. Instead of a deterministic value, probability distributions are used to characterise these variables. A brief introduction to random variables and probability distributions will be given in Sec. 2.1.1. *Gaussian* random variables constitute one particular and important example. They will be introduced in Sec. 2.1.2. The derivation of optimal performance bounds in estimation tasks relies on the analysis of *information in distributions* which will be studied in Sec. 2.1.3.

2.1.1 Basic definitions and notations

A random variable x does not possess one definite value but takes on individual realisations x with a certain likelihood. These realisations may be part of a discrete set $\mathcal{X} = \{ x_1, \ldots, x_N \}$, for example the possible outcomes when rolling a dice. A probability $\mathrm{P}\big(\mathrm{x} = x\big)$ that is assigned to each possible realisation characterises a discrete random variable.

In a driver assistance context, random variables are frequently needed to describe a representation of the environment that consists of continuous states $\mathbf{x} \in \mathbb{R}^n$, for example distances and velocities. A continuous random variable x can be defined by a *probability density function (PDF)* $p_{\mathrm{x}}\big(\mathbf{x}\big)$. The probability that a realisation $\mathbf{x}$ of x falls into a set $\mathcal{A}$ is given by the integral over the probability density function:

$$\mathrm{P}\big(\mathbf{x} \in \mathcal{A}\big) = \int_{\mathcal{A}} p_{\mathrm{x}}\big(\mathbf{x}\big) \ \mathrm{d}\mathbf{x} \ . \tag{2.1}$$

Instead of the probability density function, a random variable is also fully defined by the *cumulative distribution function (CDF)* $P_{\mathrm{x}}\big(\mathbf{x}\big) \coloneqq \mathrm{P}\big(\mathrm{x}_1 \leq x_1, \ldots, \mathrm{x}_n \leq x_n\big)$.

PDF and CDF both provide a full description of a random variable. An alternative but usually not complete characterisation is given by the central moments. Especially the

first two central moments are often used to compare two distributions or to characterise samples for which the underlying distribution is unknown.

- **Expected value:** According to the law of large numbers, the average of a large number of independently drawn realisations of a random variable converges against the expected value $\mathbb{E}_{\mathrm{x}}[\mathrm{x}]$:

$$\boldsymbol{\mu}_{\mathrm{x}} \coloneqq \mathbb{E}_{\mathrm{x}}[\mathrm{x}] = \int \mathbf{x} p_{\mathrm{x}}(\mathbf{x}) \, \mathrm{d}\mathbf{x} \, . \tag{2.2}$$

- **Covariance:** The second central moment describes the expected squared deviation from the mean:

$$\boldsymbol{\Sigma}_{\mathrm{x}} \coloneqq \mathrm{cov}(\mathrm{x}) = \mathbb{E}_{\mathrm{x}}\left[(\mathrm{x} - \mathbb{E}_{\mathrm{x}}[\mathrm{x}])(\mathrm{x} - \mathbb{E}_{\mathrm{x}}[\mathrm{x}])^{\mathrm{T}}\right] \, . \tag{2.3}$$

  In the case of a vector-valued random variable x, one refers to $\mathrm{cov}(\mathrm{x})$ as the *covariance matrix* whereas for scalar x, this quantity is known as the *variance* $\mathrm{var}(\mathrm{x})$. Furthermore, $\sigma_{\mathrm{x}} \coloneqq \sqrt{\mathrm{var}(\mathrm{x})}$ is known as the *standard deviation*.

Often, a model contains not one but multiple variables which refer to different influences. Two random variables x and y can be described by their *joint probability density function* $p_{\mathrm{x},\mathrm{y}}(\mathbf{x}, \mathbf{y})$. The *marginal density* of just one variable follows after integrating out the other:

$$p_{\mathrm{x}}(\mathbf{x}) = \int p_{\mathrm{x},\mathrm{y}}(\mathbf{x}, \mathbf{y}) \, \mathrm{d}\mathbf{y} \, . \tag{2.4}$$

An important property is *independence* of multiple random variables which means that the joint density function can be factorised:

$$p_{\mathrm{x},\mathrm{y}}(\mathbf{x}, \mathbf{y}) = p_{\mathrm{x}}(\mathbf{x}) \cdot p_{\mathrm{y}}(\mathbf{y}) \, . \tag{2.5}$$

Random variables being independent can simplify calculations to a great extent. However, the assumption of strictly independent influences is often not accurate in practical applications. The *conditional probability density* $p_{\mathrm{x}|\mathrm{y}}(\mathbf{x}|\mathbf{y})$ describes this dependence, namely the distribution of x given that y takes on a specific realisation $\mathbf{y}$:

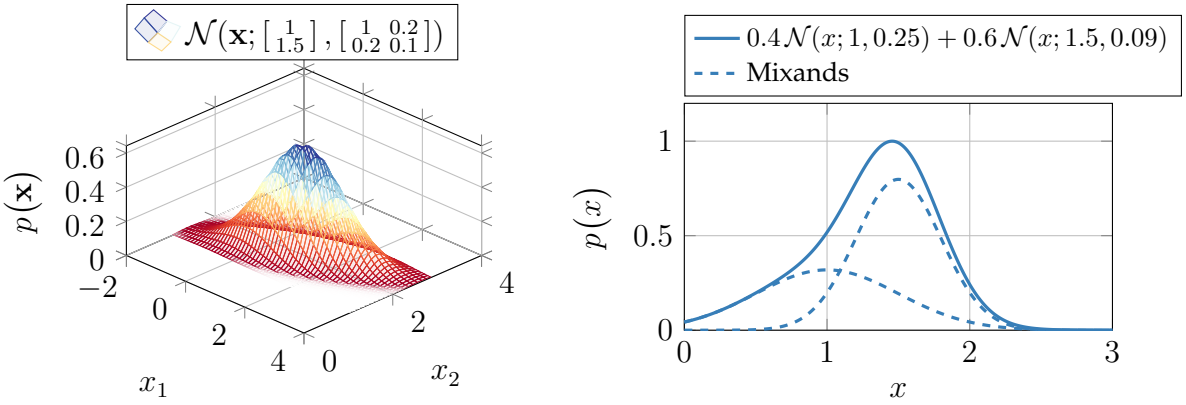
$$p_{\mathrm{x}|\mathrm{y}}(\mathbf{x}|\mathbf{y}) = \frac{p_{\mathrm{x},\mathrm{y}}(\mathbf{x}, \mathbf{y})}{p_{\mathrm{y}}(\mathbf{y})} \, . \tag{2.6}$$

In conjunction with the definition of marginal densities from (2.4) one obtains the *law of total probability*:

$$p_{\mathrm{x}}(\mathbf{x}) = \int p_{\mathrm{x}|\mathrm{y}}(\mathbf{x}|\mathbf{y}) p_{\mathrm{y}}(\mathbf{y}) \, \mathrm{d}\mathbf{y} \, . \tag{2.7}$$

Noting that $p_{\mathrm{x},\mathrm{y}}(\mathbf{x}, \mathbf{y}) = p_{\mathrm{y},\mathrm{x}}(\mathbf{y}, \mathbf{x})$, one has a relation known as *Bayes' rule*:

$$p_{\mathrm{x}|\mathrm{y}}(\mathbf{x}|\mathbf{y}) = \frac{p_{\mathrm{y}|\mathrm{x}}(\mathbf{y}|\mathbf{x}) p_{\mathrm{x}}(\mathbf{x})}{p_{\mathrm{y}}(\mathbf{y})} = \frac{p_{\mathrm{y}|\mathrm{x}}(\mathbf{y}|\mathbf{x}) p_{\mathrm{x}}(\mathbf{x})}{\int p_{\mathrm{y}|\mathrm{x}}(\mathbf{y}|\mathbf{x}) p_{\mathrm{x}}(\mathbf{x}) \, \mathrm{d}\mathbf{x}} \, . \tag{2.8}$$



(a) Bivariate Gaussian distribution. **(b)** Gaussian mixture distribution.

Figure 2.1 Visualisation of exemplary probability density functions: (a) a bivariate Gaussian distribution according to (2.10) and (b) a univariate Gaussian mixture model with two mixands according to (2.14).

In the remainder of this thesis, the difference between a random variable x and its realisation $\mathbf{x}$ will be clear from the context. Therefore, the notational differentiation is omitted and solely $\mathbf{x}$ is used for clarity. Moreover, the shorthand notations $p(\mathbf{x}) \coloneqq p_{\mathrm{x}}(\mathbf{x})$ and $P(\mathbf{x}) \coloneqq P_{\mathrm{x}}(\mathbf{x})$ are employed. Likewise, the operator $\mathbb{E}[\mathrm{x}] \coloneqq \mathbb{E}_{\mathrm{x}}[\mathrm{x}]$ is written without the subscript that indicates over which distribution the expectation is taken, unless needed for clarification.

2.1.2 Gaussian distributions

Among numerous parametric distributions, the Gaussian distribution plays an important role. Its definition and favourable properties will be introduced in the following. The practical importance of Gaussian distributions is motivated by the *central limit theorem*. Under weak assumptions, the sum of infinitely many random variables converges against a Gaussian distribution. In practice, noise processes in a technical component are caused by a combination of multiple random phenomena. Therefore, the overall noise may be regarded as the sum of individual random variables and hence is distributed approximately Gaussian.

A Gaussian random vector $\mathbf{x} \in \mathbb{R}^n \sim \mathcal{N}(\boldsymbol{\mu}_{\mathbf{x}}, \boldsymbol{\Sigma}_{\mathbf{x}})$ is characterised by the following probability density function:

$$p(\mathbf{x}) = \mathcal{N}(\mathbf{x}; \boldsymbol{\mu}_{\mathbf{x}}, \boldsymbol{\Sigma}_{\mathbf{x}}) = \frac{1}{\sqrt{\det(2\pi\boldsymbol{\Sigma}_{\mathbf{x}})}} \exp\left(-\frac{1}{2}(\mathbf{x} - \boldsymbol{\mu}_{\mathbf{x}})^{\mathrm{T}} \boldsymbol{\Sigma}_{\mathbf{x}}^{-1} (\mathbf{x} - \boldsymbol{\mu}_{\mathbf{x}})\right). \tag{2.9}$$

An example of this density for a two-dimensional random vector is visualised in Fig. 2.1(a). The Gaussian density function[1] $\mathcal{N}(\mathbf{x}; \boldsymbol{\mu}_{\mathbf{x}}, \boldsymbol{\Sigma}_{\mathbf{x}})$ is fully defined by the first two

[1] In this work, $\mathrm{x} \sim \mathcal{N}(\boldsymbol{\mu}_{\mathbf{x}}, \boldsymbol{\Sigma}_{\mathbf{x}})$ is used to indicate that x is a Gaussian random variable. Whereas $\mathcal{N}(\mathbf{x}; \boldsymbol{\mu}_{\mathbf{x}}, \boldsymbol{\Sigma}_{\mathbf{x}})$ refers to the Gaussian probability density function (2.9).

moments $\boldsymbol{\mu}_\mathbf{x} = \mathbb{E}[\mathbf{x}]$ and $\boldsymbol{\Sigma}_\mathbf{x} = \mathrm{cov}(\mathbf{x})$. For $\det(\boldsymbol{\Sigma}_\mathbf{x}) \to 0$ one has the limiting case of a *Dirac delta distribution* which is denoted by $p(\mathbf{x}) = \delta(\mathbf{x} - \boldsymbol{\mu}_\mathbf{x})$.

If a random variable, whose exact distribution is unknown, is described in terms of mean and covariance this implies a Gaussian approximation. This approximation will be frequently employed to model the propagation of a random input signal to an algorithm's output. As will be shown in Sec. 2.2, it is often impossible to analytically obtain the exact density of a non-linear function of a random variable. On the other hand, calculating the mean and variance and thus a Gaussian approximation of this density is usually much easier.

In the following, three properties which facilitate analytical derivations with dependent Gaussian random variables will be stated. Assume that $\mathbf{x} \in \mathbb{R}^n$ and $\mathbf{y} \in \mathbb{R}^m$ are jointly Gaussian with

$$p(\mathbf{x}, \mathbf{y}) = \mathcal{N}\left(\begin{bmatrix}\mathbf{x}\\ \mathbf{y}\end{bmatrix}; \begin{bmatrix}\boldsymbol{\mu}_\mathbf{x}\\ \boldsymbol{\mu}_\mathbf{y}\end{bmatrix}, \begin{bmatrix}\boldsymbol{\Sigma}_{\mathbf{xx}} & \boldsymbol{\Sigma}_{\mathbf{xy}}\\ \boldsymbol{\Sigma}_{\mathbf{yx}} & \boldsymbol{\Sigma}_{\mathbf{yy}}\end{bmatrix}\right) . \tag{2.10}$$

First, inserting the joint density into (2.4) yields that the marginal densities are Gaussian as well [172]:

$$p(\mathbf{x}) = \mathcal{N}(\mathbf{x}; \boldsymbol{\mu}_\mathbf{x}, \boldsymbol{\Sigma}_{\mathbf{xx}}) , \qquad p(\mathbf{y}) = \mathcal{N}(\mathbf{y}; \boldsymbol{\mu}_\mathbf{y}, \boldsymbol{\Sigma}_{\mathbf{yy}}) . \tag{2.11}$$

If $\boldsymbol{\Sigma}_{\mathbf{xy}} = \boldsymbol{\Sigma}_{\mathbf{yx}}^\mathrm{T} = \mathbf{0}$ it follows that $p(\mathbf{x}, \mathbf{y}) = p(\mathbf{x}) \cdot p(\mathbf{y})$ and thus $\mathbf{x}$ and $\mathbf{y}$ are independent. Second, the conditional density $p(\mathbf{x}|\mathbf{y})$ can be derived from (2.6). Again, this density remains a Gaussian [172]:

$$p(\mathbf{x}|\mathbf{y}) = \mathcal{N}(\mathbf{x}; \boldsymbol{\mu}_\mathbf{x} + \boldsymbol{\Sigma}_{\mathbf{xy}}\boldsymbol{\Sigma}_{\mathbf{yy}}^{-1}(\mathbf{y} - \boldsymbol{\mu}_\mathbf{y}), \boldsymbol{\Sigma}_{\mathbf{xx}} - \boldsymbol{\Sigma}_{\mathbf{xy}}\boldsymbol{\Sigma}_{\mathbf{yy}}^{-1}\boldsymbol{\Sigma}_{\mathbf{yx}}) . \tag{2.12}$$

The expectation of the random variable $\mathbf{x}$ is a linear function of the realisation $\mathbf{y}$.

A third important relationship concerns the marginal density that results from a Gaussian conditional density of the form $\mathcal{N}(\mathbf{x}; \mathbf{A}\mathbf{y} + \mathbf{b}, \boldsymbol{\Sigma}_{\mathbf{x}|\mathbf{y}})$ and a Gaussian prior density $\mathcal{N}(\mathbf{y}; \boldsymbol{\mu}_\mathbf{y}, \boldsymbol{\Sigma}_\mathbf{y})$. Here, we assume that $\mathbf{x}, \mathbf{b} \in \mathbb{R}^n$, $\mathbf{y} \in \mathbb{R}^m$ and $\mathbf{A} \in \mathbb{R}^{n \times m}$. Applying the law of total probability (2.7) yields for the posterior density $p(\mathbf{x})$ [172]:

$$\begin{aligned} p(\mathbf{x}) &= \int \mathcal{N}(\mathbf{x}; \mathbf{A}\mathbf{y} + \mathbf{b}, \boldsymbol{\Sigma}_{\mathbf{x}|\mathbf{y}})\, \mathcal{N}(\mathbf{y}; \boldsymbol{\mu}_\mathbf{y}, \boldsymbol{\Sigma}_\mathbf{y})\ \mathrm{d}\mathbf{y} \\ &= \mathcal{N}\left(\mathbf{x}; \mathbf{A}\boldsymbol{\mu}_\mathbf{y} + \mathbf{b}, \boldsymbol{\Sigma}_{\mathbf{x}|\mathbf{y}} + \mathbf{A}\boldsymbol{\Sigma}_\mathbf{y}\mathbf{A}^\mathrm{T}\right) . \end{aligned} \tag{2.13}$$

Sec. 4.1 will show in more detail how the property of a Gaussian distribution being its own conjugate prior [160] is helpful for estimating a system's state that evolves according to a conditional Gaussian model.

Despite the widespread applicability of a Gaussian noise assumption, some processes cannot be accurately described by a unimodal Gaussian. Measurements with a Gaussian distribution and additional sporadic outlier values are one example that will be studied in Sec. 4.1.5.1. To cope with these and other non-Gaussian phenomena, a more general

Gaussian mixture model (GMM) can be employed. The probability density function consists of a weighted sum of K Gaussian mixands:

$$p(\mathbf{x}) = \sum_{k=1}^{K} \omega_k \cdot \mathcal{N}(\mathbf{x}; \boldsymbol{\mu}_k, \boldsymbol{\Sigma}_k) \ , \qquad \sum_{k=1}^{K} \omega_k = 1 \ , \qquad \omega_k \geq 0 \quad \forall k \ . \tag{2.14}$$

Exemplarily, this density is shown in Fig. 2.1(b) for a mixture model with $K = 2$ components. In principle, any distribution can be approximated by a sufficient number of mixands [43]. One advantage of this generic model is that many results for unimodal Gaussians, as will be derived in this thesis, can be repeated en mass for multimodal GMMs. In practice, this is limited by a growing number of mixands. Additional means for recombination make this more efficient [71].

2.1.3 Information in distributions

Often, quantities of interest are not directly and accurately observable. This leads to an estimation problem where an unknown parameter $\boldsymbol{\theta}$ is to be estimated from measured data $\mathbf{x}$. A known density function $p(\mathbf{x}|\boldsymbol{\theta})$ is assumed that describes the relation between the data and the parameter.

The *Fisher information matrix (FIM)* $\boldsymbol{\mathcal{I}}_{\mathbf{x}}(\boldsymbol{\theta})$ with respect to $\boldsymbol{\theta}$ quantifies the information in $p(\mathbf{x}|\boldsymbol{\theta})$ about $\boldsymbol{\theta}$. It can be calculated according to the following two expressions that involve either the Hessian $\Delta^{\boldsymbol{\theta}}_{\boldsymbol{\theta}}(\cdot)$ or gradient[2] $\nabla_{\boldsymbol{\theta}}(\cdot)$ [74]:

$$\boldsymbol{\mathcal{I}}_{\mathbf{x}}(\boldsymbol{\theta}) = -\mathbb{E}_{\mathbf{x}|\boldsymbol{\theta}}\left[\Delta^{\boldsymbol{\theta}}_{\boldsymbol{\theta}} \log p(\mathbf{x}|\boldsymbol{\theta})\right] = \mathbb{E}_{\mathbf{x}|\boldsymbol{\theta}}\left[\left(\nabla_{\boldsymbol{\theta}} \log p(\mathbf{x}|\boldsymbol{\theta})\right)^{\mathrm{T}} \left(\nabla_{\boldsymbol{\theta}} \log p(\mathbf{x}|\boldsymbol{\theta})\right)\right] \ . \tag{2.15}$$

The covariance of any unbiased estimate $\hat{\boldsymbol{\theta}}$ of $\boldsymbol{\theta}$, that is $\mathbb{E}\left[\hat{\boldsymbol{\theta}}\right] = \boldsymbol{\theta}$, is bounded below by the inverse Fisher information matrix under certain regularity conditions [11]. This important relation is known as the Cramér-Rao bound (CRB):[3]

$$\mathrm{cov}\left(\hat{\boldsymbol{\theta}}\right) \succeq \boldsymbol{\mathcal{I}}_{\mathbf{x}}^{-1}(\boldsymbol{\theta}) \ . \tag{2.16}$$

An estimator which reaches the CRB is termed *efficient*. If an efficient unbiased estimator exists for a given problem, this is the *maximum likelihood (ML)* estimator [122]:

$$\hat{\boldsymbol{\theta}} = \arg\max_{\boldsymbol{\theta}} p(\mathbf{x}|\boldsymbol{\theta}) \ . \tag{2.17}$$

The FIM with respect to the expected value $\boldsymbol{\mu}_{\mathbf{x}}$ of a random variable $\mathbf{x}$ is a special case of (2.15). This quantity is termed *intrinsic accuracy (IA)* and is especially useful to characterise zero mean noise processes [178]:

$$\boldsymbol{\mathcal{I}}_{\mathbf{x}} \coloneqq \mathbb{E}_{\mathbf{x}}\left[\left(\nabla_{\boldsymbol{\mu}_{\mathbf{x}}} \log p(\mathbf{x})\right)^{\mathrm{T}} \left(\nabla_{\boldsymbol{\mu}_{\mathbf{x}}} \log p(\mathbf{x})\right)\right] \ . \tag{2.18}$$

[2] In this work, the numerator-layout notation is employed for derivatives of vector-valued functions or functions with multi-dimensional arguments. This notation is exemplified in (A.1) in the appendix.

[3] The notation $\mathbf{A} \succeq \mathbf{B}$ for matrices $\mathbf{A}, \mathbf{B}$ denotes that the difference $\mathbf{A} - \mathbf{B}$ is positive semidefinite.

The following general inequality holds [74]:

$$\operatorname{cov}(\mathbf{x}) \succeq \boldsymbol{\mathcal{I}}_{\mathbf{x}}^{-1} . \tag{2.19}$$

Furthermore, it is easy to show that for a Gaussian density (2.9), $\boldsymbol{\mathcal{I}}_{\mathbf{x}} = \boldsymbol{\Sigma}_{\mathbf{x}}^{-1}$ and thus (2.19) holds with equality. Therefore, it follows that if a noise process is approximated by a Gaussian with the same mean and covariance as the original density, information is always lost. To quantify this loss of information, the notion of *relative accuracy (RA)* is introduced in [74]. If (2.19) holds with equality after multiplication with a scalar $\Psi_{\mathbf{x}} \geq 1$, this coefficient denotes the relative accuracy of the distribution $p(\mathbf{x})$:

$$\operatorname{cov}(\mathbf{x}) \overset{!}{=} \Psi_{\mathbf{x}} \cdot \boldsymbol{\mathcal{I}}_{\mathbf{x}}^{-1} . \tag{2.20}$$

One advantage of a Gaussian approximation is that the maximum likelihood estimate becomes an analytical, linear function of the data for models with additive Gaussian noise. For non-Gaussian densities, an ML estimate is often not available in closed form. Instead, numerical methods are required to calculate the maximisation in (2.17). However, the computational power of embedded devices for driver assistance applications is usually fairly limited. Therefore, it is of great value to objectively assess the difference between an ML estimate with a minimum variance according to the CRB for the exact density and a sub-optimal Gaussian approximation. This will be discussed on the example of motion state estimation in Sec. 4.1.

2.2 Functions of random variables

In order to propagate probabilistic models of uncertainty through a signal processing chain, the probability densities of (non-)linear transformations of random variables have to be derived. Often, this task is not solvable in closed form and a number of methods for approximate solutions have been developed. An introduction to a linearisation-based approach that is mostly used within this thesis is given in the following. A more complete overview and more examples are provided in [69, 74, 162].

Let x be a scalar[4] random variable that is defined by a probability density $p(x)$. Applying a non-linear, differentiable function $f : \mathbb{R} \mapsto \mathbb{R}$ yields a second random variable $y = f(x)$ and the goal is to find the density function $p(y)$. This can be achieved with the definition of the density function being the derivative of the cumulative distribution function:

$$p(y) = \frac{\mathrm{d}}{\mathrm{d}y} P(y) = \frac{\mathrm{d}}{\mathrm{d}y} \int_{f(x) \leq y} p(x) \ \mathrm{d}x . \tag{2.21}$$

Unfortunately, not every non-linear transformation of a random variable possess an analytical solution of (2.21). A numerical *Monte-Carlo* approach is to draw $N_{\mathrm{s}} \gg 1$

[4] Only the scalar case is considered for clarity but all derivations can be extended to random vectors.

random samples $x^{(i)}$ from the input distribution $p(x)$ and to transform these to samples $y^{(i)} = f(x^{(i)})$. As long as sufficiently many samples are drawn, the relative frequencies of the transformed samples yield a non-parametric representation of $p(y)$. Monte-Carlo estimates of the expected value and variance can be obtained as [157]:

$$\mu_y = \frac{1}{N_\mathrm{s}} \sum_{i=1}^{N_\mathrm{s}} y^{(i)} \,, \qquad \sigma_y^2 = \frac{1}{N_\mathrm{s}-1} \sum_{i=1}^{N_\mathrm{s}} \left(y^{(i)} - \mu_y\right)^2 \,. \tag{2.22}$$

The result is non-deterministic due to the random nature of the input. Since this thesis focusses on analytical methods, numerical sampling is mainly employed for model evaluation. A more extensive account on Monte-Carlo methods can be found [49, 184].

In order to obtain a closed-form approximation of $p(y)$, one might assume a Gaussian density $\mathcal{N}(y; \mu_y, \sigma_y^2)$. To this end, the expected value and variance of y have to be calculated. A simple approach is to linearise $f(x)$ around the expected value μ_x of x:

$$f(x) \approx f(\mu_x) + A(x - \mu_x) \,, \qquad A = \frac{\mathrm{d}}{\mathrm{d}x} f(x)\bigg|_{\mu_x} \,. \tag{2.23}$$

Therefore:

$$\mu_y = \mathbb{E}[f(x)] \approx \mathbb{E}[f(\mu_x) + A(x - \mu_x)] = f(\mu_x) \tag{2.24a}$$

$$\sigma_y^2 = \mathbb{E}\left[(f(x) - \mathbb{E}[f(x)])^2\right] \approx \mathbb{E}\left[A^2 (x - \mu_x)^2\right] = A^2 \operatorname{var}(x) \,. \tag{2.24b}$$

We remark that this transformation makes two assumptions:

1. The densities $p(x)$ and $p(y)$ are accurately approximated by a Gaussian density function.
2. The linearisation of $f(x)$ is sufficiently accurate.

If $p(x)$ is in fact Gaussian and $f(x)$ a linear function, the transformation (2.24) is exact. Therefore, Gaussians are mapped to Gaussians under affine transformations which is a further reason for their frequent use. The following example will illustrate the differences between the approximation (2.24) and the exact solution (2.21).

Example 2.1 (Quadratic function of Gaussian random variable)
We consider the case where $p(x) = \mathcal{N}(x; \mu_x, \sigma_x^2)$ and $y = x^2$. Inserting into (2.21) yields the exact transformation:

$$\begin{aligned} p(y) &= \frac{\mathrm{d}}{\mathrm{d}y} \int_{-\sqrt{y}}^{\sqrt{y}} \frac{1}{\sqrt{2\pi\sigma_x^2}} \exp\left(-\frac{1}{2}\left(\frac{x-\mu_x}{\sigma_x}\right)^2\right) \mathrm{d}x \\ &= \frac{1}{\sqrt{2\pi\sigma_x^2}} \frac{1}{\sqrt{y}} \exp\left(-\frac{1}{2}\frac{y+\mu_x^2}{\sigma_x^2}\right) \cosh\left(\frac{\sqrt{y}\mu_x}{\sigma_x^2}\right) \,, \quad y > 0 \,. \end{aligned} \tag{2.25}$$

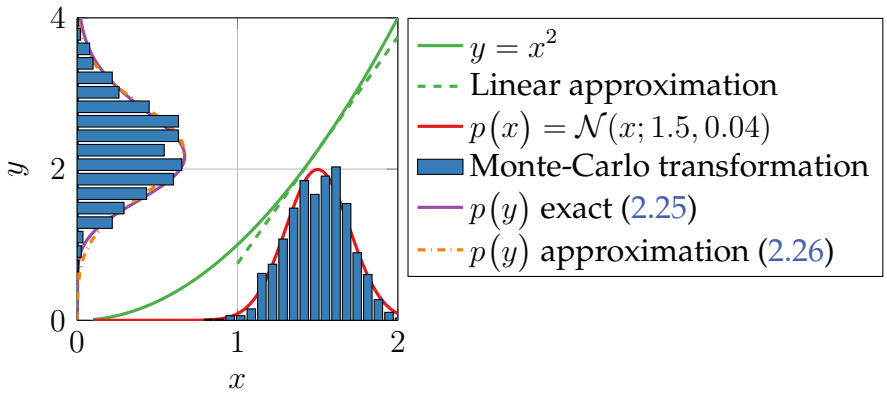


Figure 2.2 Gaussian random variable x transformed by a quadratic function $y = x^2$. The exact probability density function $p(y)$ from (2.25) is compared to the Gaussian approximation (2.26). Since the linear approximation is reasonably accurate for the considered range, a good fit between the Gaussian and the exact density is observed. A non-parametric approximation of the transformed density is obtained by individually transforming a set of Monte-Carlo samples. Note that although $N_\mathrm{s} = 1 \times 10^3$ samples were drawn, notable deviations between the samples' distribution and the true density occur. This emphasises the high computational effort that is needed to obtain an accurate Monte-Carlo approximation.

The result is known as a *non-central* χ^2-distribution with $n = 1$ degree of freedom. It can be generalised to quadratic forms $y = \mathbf{x}^\mathrm{T}\boldsymbol{\Sigma}_\mathbf{x}^{-1}\mathbf{x}$ where $\mathbf{x} \in \mathbb{R}^n \sim \mathcal{N}(\boldsymbol{\mu}_\mathbf{x}, \boldsymbol{\Sigma}_\mathbf{x})$. With the non-centrality parameter $\lambda := \boldsymbol{\mu}_\mathbf{x}^\mathrm{T}\boldsymbol{\Sigma}_\mathbf{x}^{-1}\boldsymbol{\mu}_\mathbf{x}$, the density is denoted by $\chi_n^2(y; \lambda)$. Furthermore, $\chi_n^2(y) := \chi_n^2(y; 0)$ denotes the *central* χ^2-distribution.

For the linearisation-based approach (2.24), one obtains:

$$p(y) \approx \mathcal{N}\left(y; \mu_x^2, 4\mu_x^2\sigma_x^2\right) \ . \tag{2.26}$$

We remark that this density does not per se restrict values $y < 0$ although x^2 is never negative. This indicates a weakness of only approximating the function locally around μ_x but not globally. A visual comparison of both methods is shown in Fig. 2.2.

In conclusion, an exact propagation of a probability density function through a non-linear function is often not solvable in closed form. For the special case of linear transformations and Gaussian densities, an exact analytical solution exists. Furthermore, this solution can be generalised to a linearisation-based approximation. Due to its notational consistency and analytical simplicity, this technique will be employed in the remainder of this thesis. In principle, all derivations can be repeated with more sophisticated approximations, for example based on higher-order approximation of the function [162] or by approximating the exact posterior density by a Gaussian [97].

2.3 Bayesian and frequentist interpretations of probability

Although the mathematical definitions and relationships concerning probabilities are unambiguous, different views exist on the interpretation in the context of real-world applications. The two common views of *Bayesian* and *frequentist* understanding as well as their occurrence in this thesis will be briefly discussed in the following.

In a frequentist interpretation, probabilities refer to the limiting behaviour of relative frequencies, that is, if an experiment were repeated infinitely many times. The Bayesian characterisation on the other hand regards probabilities as a quantification of belief or knowledge. This understanding is particularly well suited to formulate decision making problems.

The differences between these views become clear for the ubiquitous task of inferring the unknown value of a parameter $\boldsymbol{\theta}$ from a measured data sample $\mathbf{x}$. The measurement is assumed to be affected by random noise and is therefore described by a known probabilistic model $p(\mathbf{x}|\boldsymbol{\theta})$. Examples of this kind of model will be given in Sec. 3.2 for estimating a vehicle's pose from measured landmark locations or in Sec. 5.1 for recognising the intended manoeuvre of another driver from trajectory data.

Frequentist inference methods calculate a point estimate $\hat{\boldsymbol{\theta}}$ from a random realisation $\mathbf{x}$ and the model $p(\mathbf{x}|\boldsymbol{\theta})$. To quantify the certainty of such an estimate, *confidence regions* can be constructed from the estimator's density $p(\hat{\boldsymbol{\theta}}|\boldsymbol{\theta})$. These regions in the parameter space are defined relative to the random estimate $\hat{\boldsymbol{\theta}}$ and designed in order to include the true deterministic value $\boldsymbol{\theta}$ with a desired relative frequency.

Bayesian inference techniques on the other hand regard $\boldsymbol{\theta}$ itself as a random variable [160]. This facilitates introducing additional knowledge in the form of a prior distribution $p(\boldsymbol{\theta})$. Concerning the example of driver intention recognition, a useful prior could be obtained from traffic rules, which render certain manoeuvre hypotheses much less likely than others in a particular situation. By applying Bayes' rule (2.8) to the model $p(\mathbf{x}|\boldsymbol{\theta})$, the knowledge on $\boldsymbol{\theta}$ is updated with the information given by a measurement $\mathbf{x}$:

$$p(\boldsymbol{\theta}|\mathbf{x}) = \frac{p(\mathbf{x}|\boldsymbol{\theta})p(\boldsymbol{\theta})}{\int p(\mathbf{x}|\boldsymbol{\theta})p(\boldsymbol{\theta}) \ \mathrm{d}\boldsymbol{\theta}} \,. \tag{2.27}$$

A point estimate $\hat{\boldsymbol{\theta}}$ can be obtained from the updated density $p(\boldsymbol{\theta}|\mathbf{x})$. With the understanding that $\boldsymbol{\theta}$ is random, fixed *credible* regions can be defined using $p(\boldsymbol{\theta}|\mathbf{x})$. Confidence regions coincide with their Bayesian counterparts only in special cases since they are built from different densities and rely solely on information in the data but not prior distributions [96].

In this thesis, Bayesian approaches play an important role in the context of state estimation (Sec. 4.1), state prediction (Sec. 4.2) and decision making (Sec. 5.2). Despite the debatable subjectivity that is introduced by prior distributions, Bayesian methods are highly popular in the domain of autonomous systems since they allow for many problems to be solved in a straight-forward manner. For example, the estimate of a dynamic state may be recursively updated over time using (2.27).

3 Modelling of algorithms for environment perception

3.1 Stereo vision based object detection

Detecting static and dynamic obstacles in the surroundings is a cornerstone for the guidance of intelligent vehicles. Environment perception for automotive applications can be realised with a number of different measurement principles which include radar, ultrasound or video sensors. Recently, stereoscopic (stereo) cameras have been frequently used to obtain both 3-D position and additional visual information.

However, sensor measurements are in general affected by noise. In order to process uncertain information in a Bayes filter or to find minimum accuracy requirements, statistical models of the sensor measurements have to be derived. Following an introduction to this task in Sec. 3.1.1, the stereo vision measurement principle is outlined in Sec. 3.1.2. Multiple processing steps are usually performed to aggregate the vast information in stereo images to a compact environment representation. An exemplary algorithm to this end is introduced in Sec. 3.1.3. As the main contribution of this section, a statistical model of this algorithm is developed in Sec. 3.1.4 and evaluated in Sec. 3.1.5. The concluding Sec. 3.1.6 summarises the modelling approach and discusses the possibility of its extension to other algorithms.

3.1.1 Introduction

The stereoscopic vision measurement principle provides dense 3-D depth maps. Additionally, visual cues can be obtained from the image sensor output. Stereo vision is thus frequently used in intelligent vehicles because information rich environment perception can be achieved with a single sensor [8, 12, 17, 59, 152].

Depth information is obtained from the displacement, termed *disparity*, between images which are taken by two cameras from slightly different viewing angles. Hence, finding correspondence pairs between the two images is the first necessary step to obtain a 3-D scene representation [84]. Subsequently, relevant information is extracted from the high-dimensional disparity maps.

This section focusses on the detection of and distance measurement to a preceding vehicle, which is relevant for example for automotive emergency brake systems. Many approaches have been proposed to this end [17]. Usually, parametrised models of the expected appearance of an object are introduced and compared to the measurements.

Knowing the statistical properties of the detection result is important for system development and further processing, for example sensor data fusion [94]. However, the complexity of real-world stereo images as well as algorithms that are built on heuristics make a rigorous analytical treatment often infeasible. Previously, modelling of stereo vision based obstacle detection has been considered only in non-automotive contexts, for example robotics [135, 177]. Since the goals and underlying assumptions differ from the domain of intelligent vehicles, the statistical models are not immediately transferable.

In order to develop a probabilistic model, a simplified algorithm for vehicle detection in stereo images is considered in this work. The intention is to provide building blocks

for statistical modelling of more complex state of the art algorithms. The following material is a revised and extended version of the conference publication [223].

3.1.2 Stereo vision measurement principle

Stereo vision measurements consist of image coordinate values (u, v) and a third disparity information d which relates each point as seen in one image to the coordinates of the corresponding pixel in the second image.

Due to the perspective projection, a point (u, v, d) in disparity coordinates is related to Cartesian coordinates by a non-linear map. For convenience, it is usually differentiated between two different Cartesian coordinate systems: First, a Cartesian world coordinate system is used as a joint representation frame of points $\mathbf{p}^w = \begin{bmatrix} x^w & y^w & z^w \end{bmatrix}^{\mathrm{T}}$ in the vehicle's environment. In an automotive context, the origin usually lies in the middle of the vehicle's rear axle projected at ground plane level. The x^w-axis is pointing in the driving direction. Second, a Cartesian sensor coordinate system with the origin at the centre of the left sensor's image plane and its optical axis as x^c-axis is introduced. The transformation between both coordinate systems is defined by the extrinsic camera parameters. These are the camera's orientation expressed in a rotation matrix $\mathbf{R}$ and its position with respect to the vehicle's rear axle $\mathbf{t}$:

$$\mathbf{p}^c = \mathbf{R}\mathbf{p}^w - \mathbf{t} \ . \tag{3.1}$$

Due to the linearity of this transformation, the statistics of a measurement are easily transformed between both coordinate systems. For simplified expressions, it is assumed in the following that camera mounting angles are usually small and thus $\mathbf{R} \approx \mathbf{I}_{3\times3}$. Without loss of generality, a setup with $t_x = t_y = 0$ is considered, that is the origins are only shifted in the vertical direction.

Concerning the image generation, an ideal pinhole camera model is assumed. The two cameras are assumed to form a standard stereo configuration where the image planes are separated by the base-width b_w. Then, a point $\mathbf{p}^c$ in Cartesian sensor coordinates can be mapped to pixel locations (u_l, v_l) in the left and (u_r, v_r) in the right image.

In a single image, the depth information x_c is lost, but it can be recovered by triangulation from a second image. In the assumed rectified stereo system, corresponding points lie in same image rows and therefore $v_l = v_r$. The column-wise displacement defines the disparity $d = u_r - u_l$. Hence, all three coordinate values (x^c, y^c, z^c) of $\mathbf{p}^c$ can be calculated from (u_l, v_l, u_r) [39]:

$$x^c = \frac{c_\mathrm{k} b_\mathrm{w}}{d} \ , \qquad y^c = \frac{b_\mathrm{w}}{d}\left(u_l - u_0\right) \ , \qquad z^c = \frac{b_\mathrm{w}}{d}\left(v_l - v_0\right) \ . \tag{3.2}$$

Here, the camera constant c_k denotes the camera's focal length in pixel (pel) and (u_0, v_0) is the pixel location of the origin of the sensor coordinate system.

The transformation from Cartesian coordinates to (u_l, v_l, u_r) values reads:

$$u_l = u_0 + c_\mathrm{k}\frac{y^c}{x^c} \ , \qquad v_l = v_0 - c_\mathrm{k}\frac{z^c}{x^c} \ , \qquad d = \frac{c_\mathrm{k} b_\mathrm{w}}{x^c} \ . \tag{3.3}$$

An important aspect of the stereo measurement principle is to find corresponding pixels in both images. In the assumed standard stereo configuration, this simplifies to a search within image rows. To obtain accurate results, the displacement d between corresponding pixels is usually estimated with a finer quantisation than the native image resolution. To this end, different algorithms have been developed which estimate the displacement from multiple adjacent pixels [84].

3.1.3 Derivation of an exemplary object detection algorithm

A simplified object detection algorithm for collision warning in car-following scenarios is derived in this section. Its statistical properties will then be analysed in Sec. 3.1.4.

The proposed scheme evaluates disparity measurements in central columns of a stereo image by means of an objective function. Its central idea is to assume that relevant objects can be approximated by a vertical plane. Thus, a correspondence measure between the measurements and measurements expected from an idealised vertical plane at a specific distance is formulated. This scalar objective function is then numerically maximised over the distance parameter and the maximum peak value is compared to a threshold.

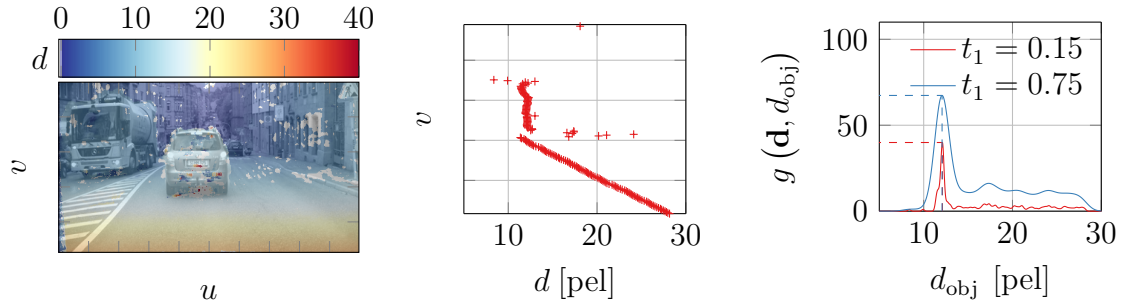
The approach is inspired by published full-scale methods. Column-wise detection of vertical obstacles – though not only in the central region but on the entire image – can be used to obtain a compact environment representation termed *stixel world* [8, 12, 152]. Hence, these algorithms have to employ similar criteria to decide whether a column in the disparity map includes a relevant object. Maximising an objective function of disparity measurements is common to many object detection approaches. These can be simple histograms [59], histograms over image rows *(v-disparity)* [92, 112] or probabilistic approaches with likelihood functions [37, 61, 152].

Due to the narrow image region that is considered here, the usual task of segmenting multiple relevant objects from the image background simplifies to a binary detection task. Moreover, instead of estimating an object's position in all three dimensions, only the one-dimensional distance information is extracted.

A rectangular disparity map is formally described by a matrix $\mathbf{D}$ of disparity values $D_{v,u}$ for each image row v and column index u. An example is visualised in Fig. 3.1(a). In the following, only a narrow rectangular region of interest is considered which is defined by the indices $\mathcal{R} := \{\, \underline{u} : \overline{u} \,\} \times \{\, \underline{v} : \overline{v} \,\}$. This region of interest is assumed to stretch over $n_{\Delta u}$ central columns and $n_{\Delta v} \gg n_{\Delta u}$ rows of the image. Disparity values from this region are firstly represented in the form of row vectors $\mathbf{d}_v := \left[D_{v,u}\right]_{u=\underline{u}:\overline{u}}$.

For improved robustness and under the assumption of spatial homogeneity of an object contour, disparity values within the region of interest are condensed to their row-wise mean value $\bar{d}_v$:

$$\bar{d}_v := \frac{1}{n_{\Delta u}} \sum_{u=\underline{u}}^{\overline{u}} D_{v,u} = \frac{1}{n_{\Delta u}} \mathbf{d}_v \cdot \mathbf{1}_{n_{\Delta u} \times 1} \,. \tag{3.4}$$



(a) Disparity map. **(b)** Row-wise mean (3.4). **(c)** Objective function (3.5).

Figure 3.1 Visualisation of stereo vision disparity measurements and the object detection algorithm in a scene with a vehicle at $x^c_{\mathrm{obj}} = 12\,\mathrm{m} \rightarrow \hat{d}_{\mathrm{obj}} \approx 12\,\mathrm{pel}$. The vehicle's rear lies within the central columns which are considered in the algorithm and it can be recognised how it is mapped to similar disparity values over multiple image rows v. The objective function $g(\mathbf{d}, d_{\mathrm{obj}})$ is calculated for two different values of the system parameter t_1. Increasing t_1 yields higher maximum values and less sensitivity to an object's contour but reduced distinction between object and background.

The values $\bar{d}_v$ from all considered image rows are written in the form of a column vector $\mathbf{d} := \left[\bar{d}_v\right]_{v=\underline{v}:\overline{v}}$. This vector is the input of the detection algorithm which is detailed in the following.

Relevant objects shall be detected from a measurement $\mathbf{d}$ by a template matching approach in disparity space [37]. A template is understood as a parametrised representation of all possible measurement realisations. It is assumed that relevant objects are ideally described by vertical planes fronto-parallel to the camera with $x^c_{\mathrm{obj}} = \mathrm{const}$. This approximately maps to disparity values $d_{\mathrm{obj}} = \mathrm{const}$. as can be seen in Fig. 3.1(b) [152]. The template is thus defined by a single parameter d_{obj}.

Formally, this approach is expressed by a correspondence measure $g(\mathbf{d}, d_{\mathrm{obj}})$. Here, a Gaussian window is applied to the distances $(d_i - d_{\mathrm{obj}})$ and the sum over all image rows is taken. Scaling the distance with a parameter t_1 governs the tolerance to small deviations:

$$g\left(\mathbf{d}, d_{\mathrm{obj}}\right) = \sum_{i=1}^{n_{\Delta v}} \exp\left(-\frac{1}{2}\frac{\left(d_i - d_{\mathrm{obj}}\right)^2}{t_1^2}\right) . \tag{3.5}$$

Note that the sum appears outside of the Gaussian, whereas a likelihood as derived in [152] consists of a multiplication of Gaussians. This is because the approach pursued here does not assume that all disparity measurements belong to the vertical object contour but can as well occur on the road surface in front of the vehicle. Using (3.5), outlier values contribute with zero weight to the correspondence measure but do not penalise the result. For a likelihood-based objective function on the other hand, additional parameters have to be introduced to indicate the image rows within which the object is located.

As it is exemplified in Fig. 3.1(c), maximising (3.5) over the template parameter d_{obj} yields a realisation of the template that corresponds best to the measurements:

$$\hat{d}_{\mathrm{obj}} = \arg\max_{d_{\mathrm{obj}}} g\left(\mathbf{d}, d_{\mathrm{obj}}\right) \ . \tag{3.6}$$

Relevant objects are differentiated from false measurements or clutter by a detection threshold on the maximum correspondence measure

$$g_{\max} = \max_{d_{\mathrm{obj}}} g\left(\mathbf{d}, d_{\mathrm{obj}}\right) = g\left(\mathbf{d}, \hat{d}_{\mathrm{obj}}\right) \ . \tag{3.7}$$

Finally, distance estimates $\hat{x}^{w}_{\mathrm{obj}}$ are calculated by transforming $\hat{d}_{\mathrm{obj}}$ to Cartesian coordinates according to (3.1)-(3.3).

3.1.4 Statistical modelling

The detection algorithm from Sec. 3.1.3 considers the row-wise mean disparity from a narrow image region and estimates if a vertically extended contour exists. A statistical analysis of the algorithm's output quantities is developed in this section. The derivation comprises four different parts which are detailed in Sec. 3.1.4.1-3.1.4.4. An overview is visualised in Fig. 3.2.

Disparity measurements, which are the input quantities of this algorithm, have to be modelled first. To this end, one has to consider the stereo camera's measurement model as well as the physical object contour. Firstly, Sec. 3.1.4.1 details measurement models which have been reported in the literature. Secondly, the scope of Sec. 3.1.4.2 is how the relevant contour of physical objects can be efficiently modelled.

Thirdly, given a probabilistic descriptions of the detection algorithm's input quantities, an analytical propagation to the output is derived in Sec. 3.1.4.3. The objective function (3.5) is analysed in terms of the maximum correspondence measure $g_{\max}$ as well as the estimated object position $\hat{d}_{\mathrm{obj}}$ in disparity coordinates.

Fourthly, the transformation of this disparity estimate to a Cartesian distance $\hat{x}_{\mathrm{obj}}$ is studied in Sec. 3.1.4.4. A statistical test is used to assess the consistency of the joint transformation of mean and variance.

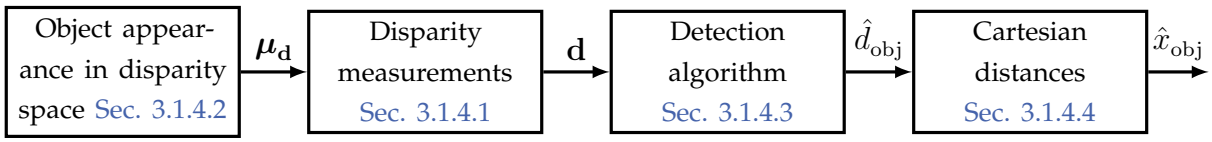


Figure 3.2 Overview of signal flow and modelling approach.

3.1.4.1 Disparity measurement error model

Stereo vision measurements according to Sec. 3.1.2 can be affected by systematic and stochastic errors. Firstly, inaccuracies during a camera calibration process cause uncertainty in the model parameters. A detailed discussion of the origin and propagation of

calibration errors is presented in [21]. These systematic errors lead to biased estimates and are not considered in the following. Secondly, the measured coordinates (u, v, d) in disparity space are affected by random noise. This is due to image quantisation effects as well as errors in the correspondence search for disparity estimation. A stochastic model will be employed to quantify these uncertainties.

Algorithms for disparity reconstruction with sub-pixel accuracy estimate the disparity from multiple adjacent pixels. This leads to correlated errors in neighbouring image columns. However, it is supposed that this correlation does not extend over image rows. As in [8, 135], a multivariate normal distribution is assumed to model the additive error $\mathbf{e}_{\mathbf{d}_v}$ in the measured disparity values $\mathbf{d}_v$ from one image row v:

$$\mathbf{d}_v = \boldsymbol{\mu}_{\mathbf{d}_v} + \mathbf{e}_{\mathbf{d}_v} \ , \qquad \mathbf{e}_{\mathbf{d}_v} \sim \mathcal{N}\left(\mathbf{0}, \boldsymbol{\Sigma}_{\mathbf{e},\mathbf{d}_v}\right) \ . \tag{3.8}$$

To parametrise the measurement error covariance $\boldsymbol{\Sigma}_{\mathbf{e},\mathbf{d}_v}$, quantisation errors can be geometrically studied and approximated by Gaussian densities [8, 25, 136]. However, as in [58, 206], these analyses focus only on the hardware parameters of the stereo system. Hence, they do not consider the influence of a specific algorithm for correspondence search on the estimated disparity. In order to obtain a comprehensive error model that includes a matching algorithm's influences, errors can be evaluated on real-world images as in [154] or synthetic data as in [164].

The input vector $\mathbf{d}$ of the detection algorithm consists of the row-wise mean values of $\mathbf{d}_v$ according to (3.4). Given (3.8), the mean values $\bar{d}_v$ are distributed as:

$$\bar{d}_v = \mu_{\bar{d}_v} + e_{\bar{d}} \ , \qquad e_{\bar{d}} \sim \mathcal{N}\left(0, \sigma^2_{e,\bar{d}}\right) \ , \qquad \sigma^2_{e,\bar{d}} = \frac{1}{n^2_{\Delta u}} \mathbf{1}_{1\times n_{\Delta u}} \boldsymbol{\Sigma}_{\mathbf{e},\mathbf{d}_v} \mathbf{1}_{n_{\Delta u}\times 1} \ . \tag{3.9}$$

A simple approximate expression for $\sigma^2_{e,\bar{d}}$ can be derived from (3.9), if a constant correlation coefficient $\rho_{e,d}$ between n_{corr} neighbouring pixels is assumed:

$$\sigma^2_{e,\bar{d}} = \frac{\sigma^2_{e,d}}{n_{\Delta u}} \left(1 + \rho_{e,d} L \frac{2n_{\Delta u} - L - 1}{n_{\Delta u}}\right) \ , \qquad L = \min\left(n_{\Delta u}, n_{\text{corr}}\right) \ . \tag{3.10}$$

Compared to the case of uncorrelated measurements with $\rho_{e,d} = 0$, the variance increases due to the second term. By assuming uncorrelated errors among image rows it follows that

$$\mathbf{d} \sim \mathcal{N}\left(\boldsymbol{\mu}_{\mathbf{d}}, \sigma^2_{e,\bar{d}} \mathbf{I}_{n_{\Delta v}\times n_{\Delta v}}\right) \ . \tag{3.11}$$

Finally, it should be noted that a correspondence search algorithms may fail entirely, for example at image regions with low texturisation. This causes invalid measurements and hence gaps in the dense disparity map, as seen in Fig. 3.1(a). Due to the high complexity of the interplay between lighting conditions, object surface and algorithm, this effect is not modelled explicitly but subsumed by a fixed percentage $0 < \eta_{\text{valid}} \leq 1$ of valid measurements.

3.1.4.2 Object appearance in disparity space

The appearance of objects in the narrow region of interest depends on object dimensions and the depth profile along the vertical axis. Due to the countless possible objects that might be visible, this analysis focusses on the detection of vehicle rear views only. In order to achieve a generic model, a probabilistic description of the depth distribution is introduced.

Coverage of image rows: The number of image rows $n_{\Delta v,\mathrm{obj}}$ that an object is mapped to depends on its height and distance to the camera. In the following, a deterministic expression is derived from geometrical considerations.

An object with height h_{obj} at distance x^{w}_{obj} is vertically confined by its ground contact position at $\mathbf{p}^{w-}_{\mathrm{obj}}$ and upper edge at $\mathbf{p}^{w+}_{\mathrm{obj}}$ in Cartesian world coordinates. Evaluating the projection to image coordinates (3.1)–(3.3) yields:

$$v^{+} = v_0 - \frac{c_{\mathrm{k}} z^{c-}_{\mathrm{obj}}}{x^{c-}_{\mathrm{obj}}} , \qquad v^{-} = v_0 - \frac{c_{\mathrm{k}} z^{c+}_{\mathrm{obj}}}{x^{c+}_{\mathrm{obj}}} . \tag{3.12}$$

These theoretical values might exceed the image's dimensions $v \in [\underline{v}, \overline{v}]$. This has to be considered when calculating the difference $n_{\Delta v,\mathrm{obj}}$ between upper and lower image row. Neglecting this effect yields a simplified formula:

$$n_{\Delta v,\mathrm{obj}} = \min\left(\overline{v}, v^{+}\right) - \max\left(\underline{v}, v^{-}\right) \tag{3.13a}$$

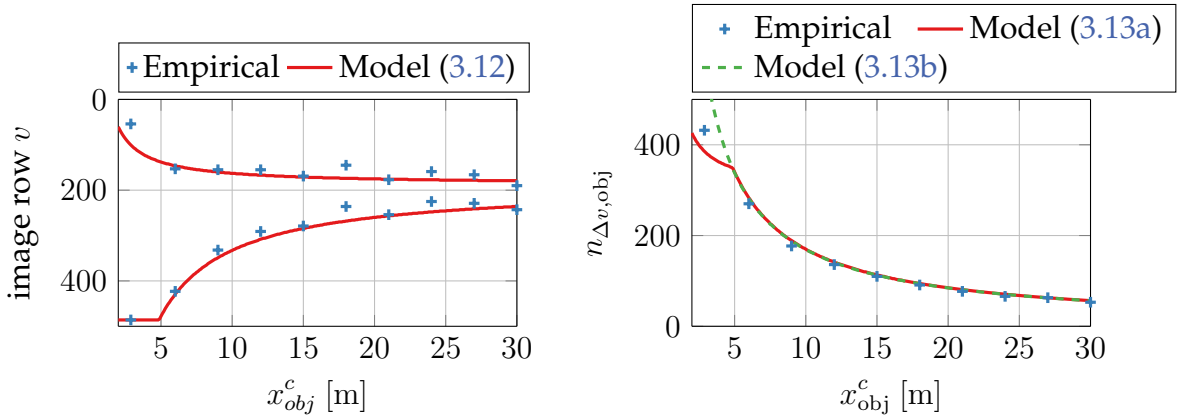
$$\approx \frac{c_{\mathrm{k}} h_{\mathrm{obj}}}{x^{c}_{\mathrm{obj}}} . \tag{3.13b}$$

The exact (3.13a) and approximate solution (3.13b) are shown in Fig. 3.3 for a vehicle with $h_{\mathrm{obj}} = 1.4\,\mathrm{m}$ and are compared to empirical values obtained at different distances x^{c}_{obj}. For the simplified approximation, deviations can be observed at small distances where upper and lower edges lie outside of the field of view of the camera.

Vehicle depth profile: The object detection algorithm from Sec. 3.1.3 compares the measured disparities $\mathbf{d}$ to a constant disparity value d_{obj} as is expected from a vertical plane. In order to obtain a model of the algorithm's output, it is necessary to describe a real vehicle's contour, that is the mean $\boldsymbol{\mu}_{\mathbf{d}}$ in (3.11), in contrast to this idealised assumption.

It is assumed that the observed vehicle's depth profile is independent of distance. Minor perspective effects are hereby neglected. Therefore, the contour is first modelled in Cartesian coordinates in terms of distances Δx relative to the vehicle rear location at x^{w}_{obj} as $\mu_x = x^{w}_{\mathrm{obj}} + \Delta x$.

One could describe the contour as pairs of depth over height values. However, this would complicate the model with a level of detail that is unnecessary for the professed goal. The detection algorithm (3.5) does not consider the image row v which a disparity measurement $\bar{d}_v$ originates from but solely the distribution of the entire sample



(a) Upper and lower image row.

(b) Number of image rows.

Figure 3.3 An object with height $h_{\mathrm{obj}} = 1.4\,\mathrm{m}$ at a distance x^c_{obj} to the camera is mapped to the upper and lower image rows shown in (a) and covers an absolute number $n_{\Delta v,\mathrm{obj}}$ of image rows shown in (b).

$\mathbf{d}$. Therefore, only the relative frequency of depth values Δx within a sample of $n_{\Delta v,\mathrm{obj}}$ image rows is modelled in the following.

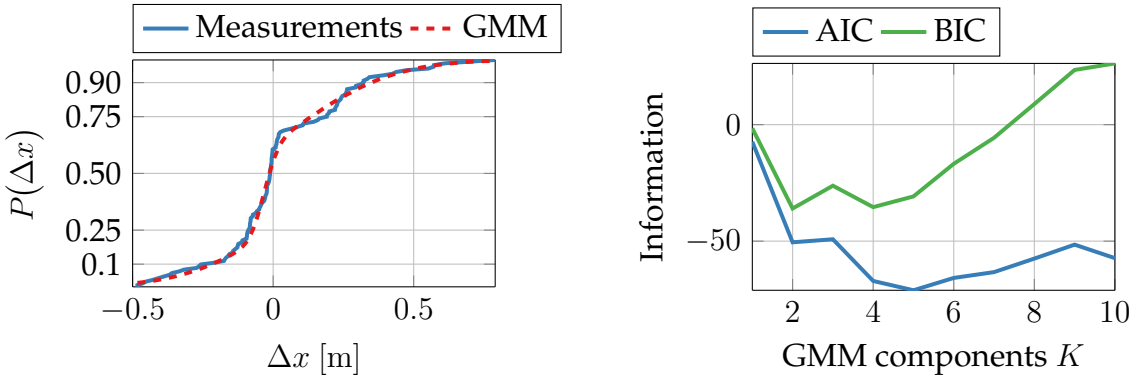
To this end, the contour depth values Δx are assumed to be independent[1] realisations of a random variable. It is therefore required that each sample $\mathbf{d}$ contains a sufficiently high number $n_{\Delta v,\mathrm{obj}}$ of measurements. A Gaussian mixture distribution is assumed:

$$p(\Delta x) = \sum_{k=1}^{K} \omega_{\Delta x,k} \cdot \mathcal{N}\left(\Delta x; \mu_{\Delta x,k}, \sigma^2_{\Delta x,k}\right) \;, \qquad \sum_{k=1}^{K} \omega_{\Delta x,k} = 1 \;, \qquad \omega_{\Delta x,k} \geq 0 \quad \forall k \;. \tag{3.14}$$

The model parameters $\mu_{\Delta x,k}$, $\sigma^2_{\Delta x,k}$ and $\omega_{\Delta x,k}$ can be estimated with the expectation maximisation principle [43]. Firstly though, a choice on the number of mixture components K has to be made. While a high-dimensional model improves the achievable fidelity, it may cause overfitting and increase the estimate's uncertainty. To find a sensible value, information theoretic criteria can be employed, such as *Akaike's information criterion (AIC)* [2] or the *Bayesian information criterion (BIC)* [176]. Thus, the parameter estimation is carried out for varying K and the values of the information criteria are compared. The minimum value corresponds to the best fitting model.

To exemplify the approach, a static vehicle rear is recorded by a stereo video sensor at a distance of $x^w_{\mathrm{obj}} = 6\,\mathrm{m}$. In order to remove the measurement noise influence, the average of the centred depth value Δx for each image row is calculated over $N_{\mathrm{frames}} = 243$ frames. The GMM parameters are then estimated from the averaged values. Fig. 3.4 shows the resulting model and the information criteria for $K = 1, \ldots, 10$ components. The chosen model order is $K = 2$ where the BIC measure attains an optimum.

[1] Since the physical contour is always continuous, the depth values are locally correlated. This is neglected because only the overall distribution is of interest.



(a) Cumulative distribution.

(b) Akaike's (AIC) and Bayesian (BIC) information criteria.

Figure 3.4 Approximation of a vehicle depth profile Δx from measurements at $x^w_{\mathrm{obj}} = 6\,\mathrm{m}$. The contour distribution and a GMM with $K = 2$ components are visualised in (a). The number of components is selected according to the information criteria shown in (b).

In a second step, the model (3.14) of the Cartesian distance μ_x is transformed to a GMM $p(\mu_d)$ in disparity domain. Firstly, the expected values $\mu_{\Delta x,k}$ are transformed by (3.3). Secondly, the variance components $\sigma^2_{\Delta x,k}$ are propagated by linearisation of the non-linear transformation (3.3) around $x^w_{\mathrm{obj}} + \mu_{\Delta x,k}$. The weights $\omega_{\Delta x,k}$ are left unchanged [185]. In summary, an object contour sample in disparity coordinates is described by:

$$p(\mu_d) = \sum_{k=1}^{K} \omega_{\mu_d,k} \cdot \mathcal{N}(\mu_d; \mu_{\mu_d,k}, \sigma^2_{\mu_d,k}) \tag{3.15a}$$

with

$$\mu_{\mu_d,k} = \frac{c_{\mathrm{k}} b_{\mathrm{w}}}{x^w_{\mathrm{obj}} + \mu_{\Delta x,k}}\,, \quad \sigma^2_{\mu_d,k} = \left(\frac{c_{\mathrm{k}} b_{\mathrm{w}}}{x^w_{\mathrm{obj}} + \mu_{\Delta x,k}}\right)^2 \sigma^2_{\Delta x,k}\,, \quad \omega_{\mu_d,k} = \omega_{\Delta x,k}\,. \tag{3.15b}$$

Fig. 3.5 shows how the contour model from Fig. 3.4, that was estimated at $x^w_{\mathrm{obj}} = 6\,\mathrm{m}$, is propagated to disparity space at distances $x^w_{\mathrm{obj}} \in [6\,\mathrm{m}, 30\,\mathrm{m}]$. Empirical measurements taken at the respective distances are shown for comparison. These show that in general, a satisfactory approximation can be achieved. However, especially the tails of the distribution are not fitted well at higher distances. This can be attributed to additional unmodeled effects caused by a change of perspective.

3.1.4.3 Propagation to the object detection algorithm

The goal of the following analysis is to analytically propagate the previously derived models for measurement errors and object appearance to the detection algorithm output. This comprises to model the estimated object position in disparity coordinates $\hat{d}_{\mathrm{obj}}$ as well as the maximum objective function value $g_{\max}$. Gaussian density functions are assumed as models for both quantities. This simplifies further analytical propagation of the results. Therefore, the first two moments, expected value and variance, of $\hat{d}_{\mathrm{obj}}$ and

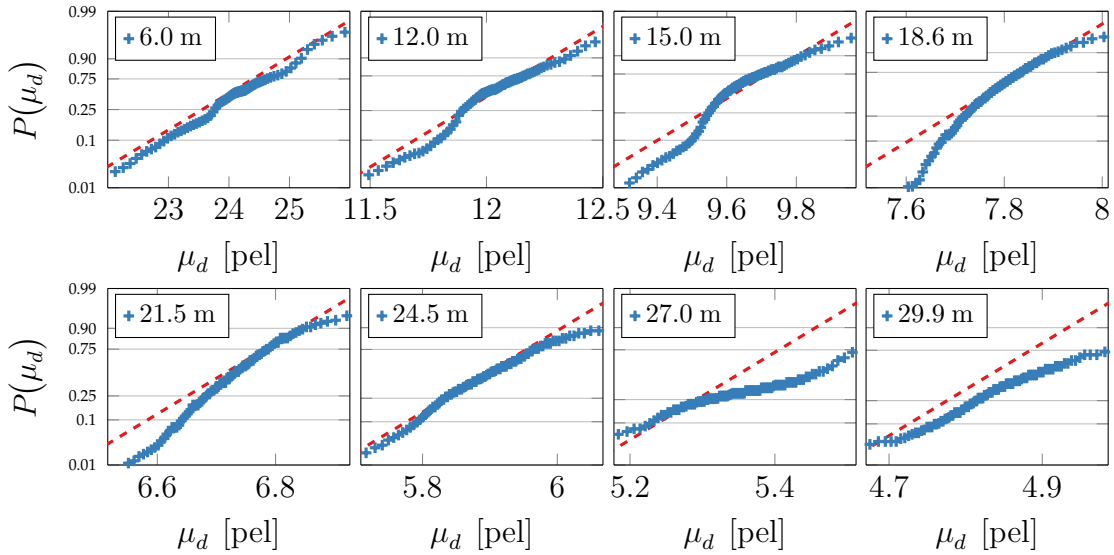


Figure 3.5 Propagation of a vehicle depth profile estimated at $x^w_{\mathrm{obj}} = 6\,\mathrm{m}$ from Fig. 3.4 to disparity space according to (3.15). Empirical measurements taken at the respective distances (blue) are compared to the analytical model (red). With increasing distance, the quality of the transformed model deteriorates due to additional unmodeled changes of perspective.

$g_{\max}$ have to be calculated. Monte-Carlo experiments will be used to verify the analytical expressions.

Probabilistic models have been derived for both the sensor measurement errors in Sec. 3.1.4.1 as well as the object contour in Sec. 3.1.4.2. However, solely measurement errors cause random fluctuations of the algorithm outputs. Whereas a probabilistic contour model describes the distribution of disparity values of a non-varying depth profile. The expected values and variances are thus derived depending on the contour $\boldsymbol{\mu}_{\mathbf{d}}$ first.

Subsequently, there are two approaches how the contour model is incorporated in the derivations: Either, the expectation of all results with respect to the GMM (3.15) is taken. Or, if only one specific contour with a known realisation is considered, $\boldsymbol{\mu}_{\mathbf{d}}$ is treated as a constant parameter.

Expected values We assume that a sample $\mathbf{d}$ of $n_{\Delta v,\mathrm{obj}}$ disparity values d which are independently and identically distributed according to $d \sim \mathcal{N}(\mu_d, \sigma^2_{e,\bar{d}})$ are given. The mean μ_d is described by the probabilistic contour model (3.15). The disparity measurement noise variance $\sigma_{e,\bar{d}}$ follows from Sec. 3.1.4.1. In the following, the expected values of the location $\hat{d}_{\mathrm{obj}}$ and maximum value $g_{\max}$ of the objective function are to be calculated.

As there is in general no closed-form expression for $g_{\max}$, the first simplification is to exchange the order of the expectation and maximisation operator. Due to the non-

linearity of the $\max(\cdot)$ operation, this is only an approximation:

$$\mathbb{E}\left[\max_{d_{\text{obj}}} g\left(\mathbf{d}, d_{\text{obj}}\right)\right] \approx \max_{d_{\text{obj}}} \sum_{i=1}^{n_{\Delta v,\text{obj}}} \mathbb{E}\left[g\left(d_i, d_{\text{obj}}\right)\right] \approx n_{\Delta v,\text{obj}} \mathbb{E}\left[g\left(d, \mathbb{E}\left[\hat{d}_{\text{obj}}\right]\right)\right] \ . \tag{3.16}$$

Taking the expectation using the probability density function $p\left(\mu_d\right)$ of the contour model from (3.15) and the measurement noise distribution $e_{\bar{d}} \sim \mathcal{N}\left(0, \sigma_{e,\bar{d}}^2\right)$ yields:

$$\begin{aligned}
&\mathbb{E}_{e_{\bar{d}},\mu_d}\left[g\left(\mu_d + e_{\bar{d}}, \mathbb{E}\left[\hat{d}_{\text{obj}}\right]\right)\right] \\
&= \int_{-\infty}^{\infty} g\left(d, \mathbb{E}\left[\hat{d}_{\text{obj}}\right]\right) \cdot \sum_{k=1}^{K} \frac{\omega_{\mu_d,k}}{\sqrt{2\pi\left(\sigma_{\mu_d,k}^2 + \sigma_{e,\bar{d}}^2\right)}} \exp\left(-\frac{1}{2}\frac{\left(d - \mu_{\mu_d,k}\right)^2}{\sigma_{\mu_d,k}^2 + \sigma_{e,\bar{d}}^2}\right) \mathrm{d}d \\
&= \sum_{k=1}^{K} \frac{\omega_{\mu_d,k} t_1}{\sqrt{t_1^2 + \sigma_{\mu_d,k}^2 + \sigma_{e,\bar{d}}^2}} \cdot \exp\left(-\frac{1}{2}\frac{\left(\mu_{\mu_d,k} - \mathbb{E}\left[\hat{d}_{\text{obj}}\right]\right)^2}{t_1^2 + \sigma_{\mu_d,k}^2 + \sigma_{e,\bar{d}}^2}\right) \ .
\end{aligned} \tag{3.17}$$

Unfortunately, maximising (3.17) in closed form is in general not possible. Therefore, the series expansion up to the quadratic term is studied instead. Taking the derivative of the quadratic approximation yields a unique maximum and one obtains:

$$\mathbb{E}\left[\hat{d}_{\text{obj}}\right] \approx \left[\sum_{k=1}^{K} \frac{\omega_{\mu_d,k}\mu_{\mu_d,k}}{\left(t_1^2 + \sigma_{\mu_d,k}^2 + \sigma_{e,\bar{d}}^2\right)^{\frac{3}{2}}}\right] \cdot \left[\sum_{k=1}^{K} \frac{\omega_{\mu_d,k}}{\left(t_1^2 + \sigma_{\mu_d,k}^2 + \sigma_{e,\bar{d}}^2\right)^{\frac{3}{2}}}\right]^{-1} \ . \tag{3.18}$$

Finally, the expected value of the maximum $g_{\max}$ of the objective function is given by (3.17) evaluated at $\mathbb{E}\left[\hat{d}_{\text{obj}}\right]$ from (3.18).

Variances The following analytical condition defines the location of a maximum of the objective function:

$$0 \overset{!}{=} \frac{\partial}{\partial d_{\text{obj}}} g\left(\mathbf{d}, d_{\text{obj}}\right)\Big|_{\hat{d}_{\text{obj}}} \ . \tag{3.19}$$

In order to find the variance of $\hat{d}_{\text{obj}}$, a linearisation of $\hat{d}_{\text{obj}}\left(\mathbf{d}\right)$ around the expected value $\boldsymbol{\mu}_{\mathbf{d}}$ is performed. This requires the gradient of $\hat{d}_{\text{obj}}\left(\mathbf{d}\right)$ with respect to $\mathbf{d}$. Since an explicit expression for this function is unavailable, the implicit function theorem can be used to obtain the gradient as a function of the partial derivatives of $g\left(\mathbf{d}, d_{\text{obj}}\right)$ from (3.5) [55]:

$$\frac{\partial \hat{d}_{\text{obj}}\left(\mathbf{d}\right)}{\partial d_i} = -\left(\frac{\partial^2 g\left(\mathbf{d}, d_{\text{obj}}\right)}{\partial^2 d_{\text{obj}}}\right)^{-1} \left(\frac{\partial^2 g\left(\mathbf{d}, d_{\text{obj}}\right)}{\partial d_i \partial d_{\text{obj}}}\right) \ , \qquad i = 1, \ldots, n_{\Delta v,\text{obj}} \ . \tag{3.20}$$

The measurement model $\mathbf{d} \sim \mathcal{N}\left(\boldsymbol{\mu}_{\mathbf{d}}, \sigma_{e,\bar{d}}^2 \mathbf{I}_{n_{\Delta v} \times n_{\Delta v}}\right)$ from (3.11) is thus propagated to the variance of $\hat{d}_{\text{obj}}$ as follows:

$$\begin{aligned}
\text{var}\left(\hat{d}_{\text{obj}}\right) &= \sum_{i=1}^{n_{\Delta v,\text{obj}}} \sigma_{e,\bar{d}}^2 \left(\left(\frac{\partial^2 g\left(\mathbf{d}, d_{\text{obj}}\right)}{\partial^2 d_{\text{obj}}} \right)^{-1} \left(\frac{\partial^2 g\left(\mathbf{d}, d_{\text{obj}}\right)}{\partial d_i \partial d_{\text{obj}}} \right) \Bigg|_{\boldsymbol{\mu}_{\mathbf{d}}, \mathbb{E}\left[\hat{d}_{\text{obj}}\right]} \right)^2 \\
&= \sigma_{e,\bar{d}}^2 \frac{\sum_{i=1}^{n_{\Delta v,\text{obj}}} \left[\exp\left(-\frac{1}{2t_1^2} \left(\mu_{d_i} - \mathbb{E}\left[\hat{d}_{\text{obj}}\right] \right)^2 \right) \left(t_1^2 - \left(\mu_{d_i} - \mathbb{E}\left[\hat{d}_{\text{obj}}\right] \right)^2 \right) \right]^2}{\left[\sum_{i=1}^{n_{\Delta v,\text{obj}}} \exp\left(-\frac{1}{2t_1^2} \left(\mu_{d_i} - \mathbb{E}\left[\hat{d}_{\text{obj}}\right] \right)^2 \right) \left(t_1^2 - \left(\mu_{d_i} - \mathbb{E}\left[\hat{d}_{\text{obj}}\right] \right)^2 \right) \right]^2} .
\end{aligned} \tag{3.21}$$

Similarly, the cross-covariance $\text{cov}\left(\hat{d}_{\text{obj}}, d_i\right)$, $i = 1, \ldots, n_{\Delta v,\text{obj}}$ is obtained:

$$\text{cov}\left(\hat{d}_{\text{obj}}, d_i\right) = \sigma_{e,\bar{d}}^2 \frac{\exp\left(-\frac{1}{2t_1^2} \left(\mu_{d_i} - \mathbb{E}\left[\hat{d}_{\text{obj}}\right] \right)^2 \right) \left(t_1^2 - \left(\mu_{d_i} - \mathbb{E}\left[\hat{d}_{\text{obj}}\right] \right)^2 \right)}{\sum_{i=1}^{n_{\Delta v,\text{obj}}} \exp\left(-\frac{1}{2t_1^2} \left(\mu_{d_i} - \mathbb{E}\left[\hat{d}_{\text{obj}}\right] \right)^2 \right) \left(t_1^2 - \left(\mu_{d_i} - \mathbb{E}\left[\hat{d}_{\text{obj}}\right] \right)^2 \right)} . \tag{3.22}$$

The variance depends on the contour $\boldsymbol{\mu}_{\mathbf{d}}$. This means that the same measurement errors affect the estimate $\hat{d}_{\text{obj}}$ differently for different vehicle rears. A lower bound on the variance that is independent of the contour is derived by applying the Cauchy-Schwarz inequality[2] to the denominator of (3.21):

$$\text{var}\left(\hat{d}_{\text{obj}}\right) \geq \frac{\sigma_{e,\bar{d}}^2}{n_{\Delta v,\text{obj}}} . \tag{3.23}$$

The interpretation of this lower bound is straight-forward: In the best case, the detection algorithm achieves the same variance as a simple averaging over independent measurements from $n_{\Delta v,\text{obj}}$ image rows. The latter results in the variance law that is given at the right-hand side of (3.23). If all measurements were independently and identically distributed and stem from the observed vehicle, taking the average disparity value yields a maximum likelihood estimate.

In practical situations though, outlier measurements occur, for example from the road surface close to the vehicle. It is the purpose of the objective function (3.5) to suppress the influence of these outliers by weighting with the Gaussian window function. The trade-off is that in the case of non-vertical vehicle contours, the gathered information reduces and the variance increases. It is easily shown that for the special case of an ideally vertical object with $\mu_{d_i} = d_{\text{obj}}\ \forall i$, both the location estimate $\hat{d}_{\text{obj}}$ from (3.6) as well as the variance in (3.21) simplify to the averaging strategy with equality.

[2] $\left(\sum_{i=1}^n x_i y_i\right)^2 \leq \left(\sum_{i=1}^n x_i^2\right) \cdot \left(\sum_{i=1}^n y_i^2\right) \overset{y_i=1}{\Longrightarrow} \left(\sum_{i=1}^n x_i\right)^2 \leq n \cdot \sum_{i=1}^n x_i^2$.

Finally, an expression for the variance of the maximum value $g_{\max}$ of the objective function is obtained by linearising $g\left(\mathbf{d}, d_{\text{obj}}\right)$ around $\boldsymbol{\mu}_{\mathbf{d}}$ and $\mathbb{E}\left[\hat{d}_{\text{obj}}\right]$:

$$\begin{aligned}\operatorname{var}\left(g_{\max}\right) = &\sum_{i=1}^{n_{\Delta v,\text{obj}}} \sigma_{e,\bar{d}}^2 \left(\frac{\partial g\left(\mathbf{d}, d_{\text{obj}}\right)}{\partial d_i}\bigg|_{\boldsymbol{\mu}_{\mathbf{d}}, \mathbb{E}\left[\hat{d}_{\text{obj}}\right]} \right)^2 + \operatorname{var}\left(\hat{d}_{\text{obj}}\right) \left(\frac{\partial g\left(\mathbf{d}, d_{\text{obj}}\right)}{\partial d_{\text{obj}}}\bigg|_{\boldsymbol{\mu}_{\mathbf{d}}, \mathbb{E}\left[\hat{d}_{\text{obj}}\right]} \right)^2 \\ &+ 2 \sum_{i=1}^{n_{\Delta v,\text{obj}}} \operatorname{cov}\left(\hat{d}_{\text{obj}}, d_i\right) \left(\frac{\partial g\left(\mathbf{d}, d_{\text{obj}}\right)}{\partial d_i} \frac{\partial g\left(\mathbf{d}, d_{\text{obj}}\right)}{\partial d_{\text{obj}}}\bigg|_{\boldsymbol{\mu}_{\mathbf{d}}, \mathbb{E}\left[\hat{d}_{\text{obj}}\right]} \right) .\end{aligned} \tag{3.24}$$

Numerical example In a Monte-Carlo simulation with $N_{\text{sim}} = 1 \times 10^3$ iterations, vehicle contour samples comprising $n_{\Delta v,\text{obj}} = 50$ values each are drawn from a GMM distribution $p\left(\Delta x\right)$ that was shown in Fig. 3.4. Each sample is mapped to disparity space for varying distances $x_{\text{obj}}^w \in [2\,\text{m}, 30\,\text{m}]$ according to (3.1)-(3.3) and correlated Gaussian measurement noise with $\sigma_d = 0.1\,\text{pel}$ according to the model from Sec. 3.1.4.1 is added. In every simulation run, the detection algorithm outputs (3.6) and (3.7) are calculated.

Firstly, the average of the simulated values of $\hat{d}_{\text{obj}}$ and $g_{\max}$ are calculated. These are compared to the closed-form expressions for the expected values in Fig. 3.6(a) and a good fit can be observed. As has been shown in Fig. 3.5, the disparity distribution narrows for higher distances and thus the peak value of the objective function increases. That is why $g_{\max}$ approaches the theoretical maximum $n_{\Delta v,\text{obj}}$ as it is expected for the ideal case $d_i = d_{\text{obj}}\ \forall i$.

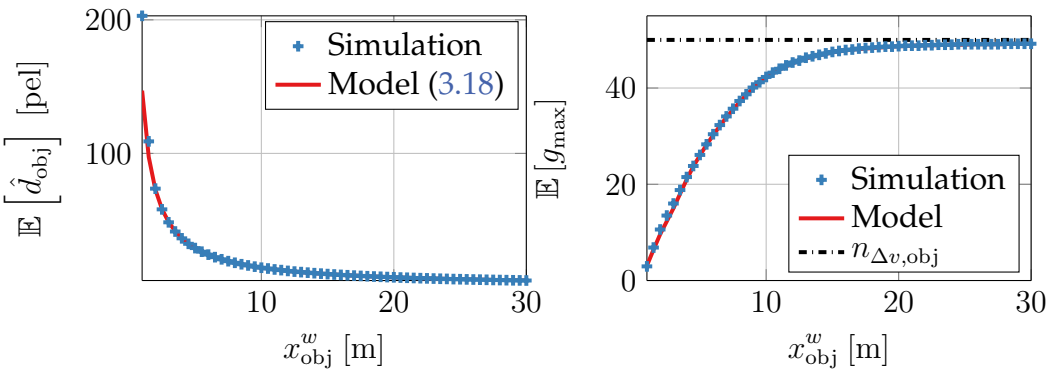
Secondly, the standard deviations in $\hat{d}_{\text{obj}}$ and $g_{\max}$ are shown in Fig. 3.6(b). In contrast to the previous simulation, only the realisations of the measurement noise are sampled whereas the vehicle contour distribution is drawn once and treated as a known constant parameter.

It can be observed that the difference between the true standard deviation and the analytical lower bound from (3.23) increases at short distances. This is again caused by a pronounced influence of the contour's depth distribution in disparity space at short distances. The lower bound does not depend on the depth distribution and approaches the observed values for distances $\geq 20\,\text{m}$ in this example.

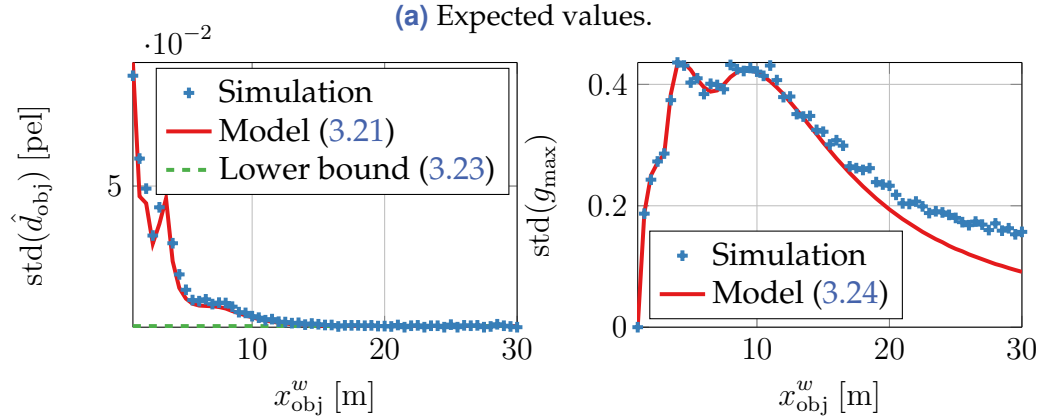
The variance of $g_{\max}$ shows a deviation between simulated and analytical results at high distances. As has been noted before, the objective function narrows in these cases. Thus, the quality of a linear approximation around the maximum, as it is employed to calculate the variance, decreases.

Summary Gaussian error propagation has been derived for the exemplary detection algorithm from Sec. 3.1.3. The closed-form expressions for expectation and variance have been verified in Monte-Carlo simulations.

The algorithm's output depends on a vehicle's contour in contrast to an ideally vertical plane. This effect is most prominent at short absolute distances, due to the inversely



(a) Expected values.



(b) Standard deviations.

Figure 3.6 Monte-Carlo simulation results of the algorithm from Sec. 3.1.3 compared to analytical models from Sec. 3.1.4.3. (a) shows the mean values of the estimated object position in disparity coordinates $\hat{d}_{\text{obj}}$ and the objective functions maximum value $g_{\max}$. Both quantities are well described by the analytical models. (b) visualises the standard deviation of the estimates for randomly sampled values.

proportional mapping between Cartesian coordinates and disparity space. It has been shown that this dependence is related to the algorithm's robustness against outlier measurements. This comes at the price of an increased variance in contrast to a maximum likelihood estimate in an ideal case without outliers.

One challenge in the derivation of the algorithm's statistical properties is that it is only implicitly defined in terms of an objective function. Apart from special cases, an analytical solution is not available and thus a numerical solver has to be applied. Considering the simplicity of the chosen exemplary algorithm, one should reasonably not expect that similar challenges will not occur in more sophisticated state of the art implementations.

3.1.4.4 Transformation to Cartesian coordinates

So far, position estimates of an object have been obtained as disparity values d. However, an object's motion is usually better modelled in Cartesian coordinates. Hence, for the

purpose of object tracking, one needs to transform the disparity values to Cartesian distances x.

Bayesian state estimation depends on precise knowledge of the measurement error model. A usual modelling assumption is that measurement errors are zero mean Gaussians with known variance. However, given a disparity estimate $\hat{d} = \mu_d + e_d$ with Gaussian errors $e_d \sim \mathcal{N}\left(0, \sigma_d^2\right)$, the exact probability density function of the distance $\hat{x}$ after transformation by (3.2) follows from (2.21) and is non-Gaussian [179]:

$$\hat{x} = \frac{c_\mathrm{k} b_\mathrm{w}}{\hat{d}} \quad \Longrightarrow \quad p\left(\hat{x}\right) = \frac{c_\mathrm{k} b_\mathrm{w}}{\sqrt{2\pi}\sigma_d} \frac{1}{\hat{x}^2} \exp\left(-\frac{1}{2} \frac{\left(\frac{c_\mathrm{k} b_\mathrm{w}}{\hat{x}} - \mu_d\right)^2}{\sigma_d^2}\right) . \tag{3.25}$$

In order to be used in a Bayes filter with a Gaussian representation of the states, a Gaussian approximation $\mathcal{N}\left(\hat{x}; \mu_x, \sigma_x^2\right)$ of (3.25) can be derived with [8]:

$$\mu_x = \frac{c_\mathrm{k} b_\mathrm{w}}{\mu_d} , \qquad \sigma_x^2 = \mathbb{E}\left[\left(\frac{\mathrm{d}}{\mathrm{d}d} \frac{c_\mathrm{k} b_\mathrm{w}}{d}\bigg|_{\mu_d} (d - \mu_d)\right)^2\right] \approx \left(\frac{c_\mathrm{k} b_\mathrm{w}}{\hat{d}^2}\right)^2 \sigma_d^2 . \tag{3.26}$$

Previous works [8, 179] have studied the goodness of this approximation in terms of a bias that is introduced by the non-linear transformation from disparity to Cartesian coordinates. An additional aspect concerns the question whether the Gaussian with $\left(\mu_x, \sigma_x^2\right)$ yields a consistent approximation of (3.25). Thereby, not only the transformation of the mean but also the variance from (3.26) is considered. The latter is based on a linearisation around the true, in practice unknown expected value μ_d. If instead, the noisy measurement $\hat{d}$ is used as a linearisation point, additional uncertainty is induced [120, 168].

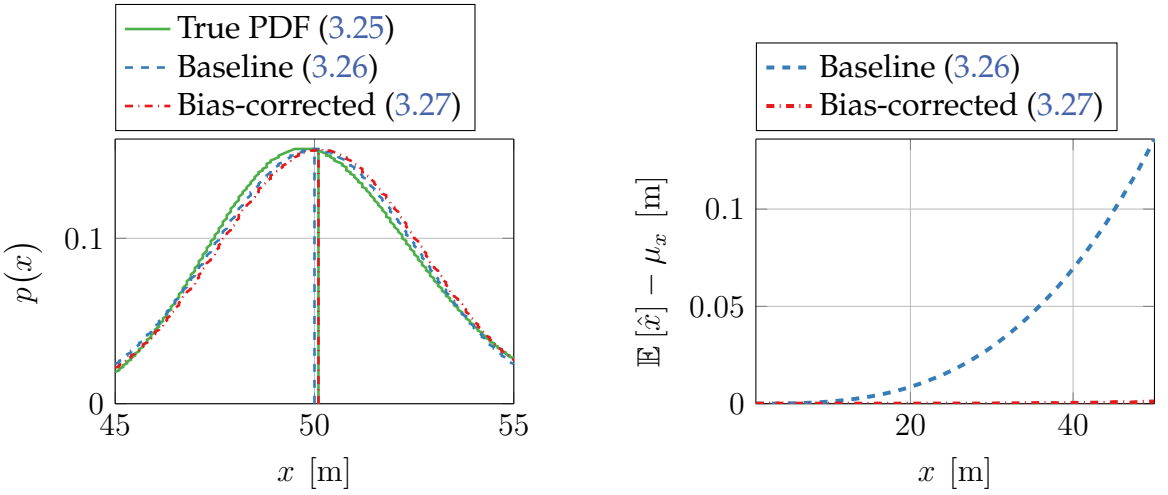
First, concerning the mean,[3] the following more exact transformation can be derived with a second order expansion of (3.2):

$$\mu_x = \frac{c_\mathrm{k} b_\mathrm{w}}{\mu_d} \left(1 + \frac{\sigma_d^2}{\mu_d^2}\right) . \tag{3.27}$$

In order to assess the difference between (3.26) and (3.27), a Monte-Carlo simulation with $N_\mathrm{sim} = 5 \times 10^8$ independent samples is employed. Virtual disparity measurements $\hat{d}^{(i)}$, $i = 1, \ldots, N_\mathrm{sim}$ are sampled from a Gaussian distribution $\mathcal{N}\left(\hat{d}; \mu_d, \sigma_d^2\right)$ with mean disparity values μ_d that correspond to distances $x = 1 \ldots 50\,\mathrm{m}$ and $\sigma_d = 0.15\,\mathrm{pel}$. The results are shown in Fig. 3.7 and demonstrate the usefulness of the corrected transformation.

Second, the transformation of the variance and the uncertainty introduced by the linearisation point are analysed. To this end, $\mu_d = \hat{d} - e_d$ is inserted into the variance

[3] Note that the works [8, 179] do not explicitly consider the calculation of the mean μ_x for the given transformation (3.2). Instead, they propose an alternative transformation to calculate $\hat{x}$ with the property that the transformed mean equals the mean of the transformed sample. The result differs only by the sign of the additive term in (3.27).



(a) Probability density functions of samples drawn from $\mathcal{N}\left(\mu_d, \sigma_d^2\right)$ and transformed to Cartesian distances. The distribution parameters are μ_d corresponding to 50 m and $\sigma_d = 0.15$ pel.

(b) The bias in (3.26) increases for higher distances and is always positive. With the corrected transformation of the mean (3.27), almost bias-free results are achieved.

Figure 3.7 Monte-Carlo simulation results of the baseline transformation (3.26) and the bias-corrected formula (3.27). The baseline Gaussian approximation (3.26) exhibits a bias in comparison to the exact probability density function (3.25).

formula in (3.26) and a second order Taylor expansion around $\mathbb{E}[e_d] = 0$ is performed. Taking the expectation with respect to e_d yields the expected variance $\mathbb{E}\left[\sigma_x^2 | \hat{d}\right]$ conditional on a measurement $\hat{d}$ [168]. Compared to the expression from (3.26) conditional on the true mean μ_d , the variance is increased:

$$\begin{aligned}\mathbb{E}\left[\sigma_x^2|\hat{d}\right] &\approx (c_\mathrm{k} b_\mathrm{w})^2 \cdot \mathbb{E}\left[\left.\frac{1}{\left(\hat{d}-e_d\right)^4}\right|_{e_d=0} - \left.\frac{4}{\left(\hat{d}-e_d\right)^5}\right|_{e_d=0} \cdot e_d + \frac{1}{2}\left.\frac{20}{\left(\hat{d}-e_d\right)^6}\right|_{e_d=0} \cdot e_d^2\right] \sigma_d^2 \\ &= \left(\frac{c_\mathrm{k} b_\mathrm{w}}{\hat{d}^2}\right)^2 \left(1 + \frac{10\sigma_d^2}{\hat{d}^2}\right) \sigma_d^2 \,. \end{aligned} \tag{3.28}$$

The consistency of the joint estimates $\left(\mu_x, \sigma_x^2\right)$ is evaluated in a Monte-Carlo simulation as detailed before. The variance $\sigma_x^{(i)}$ depends on the sample value $\hat{d}^{(i)}$ which serves as the linearisation point in (3.26) and (3.28), respectively. The expected value μ_x is transformed according to the baseline approach (3.26) or with an additional bias correction as in (3.27).

With these transformed distribution parameters, the normalised error squared (NES) is calculated [168]:

$$\bar{\psi} = \sum_{i=1}^{N_\mathrm{sim}} \left(\frac{\hat{x}^{(i)} - \mu_x}{\sigma_x^{(i)}}\right)^2 \,. \tag{3.29}$$

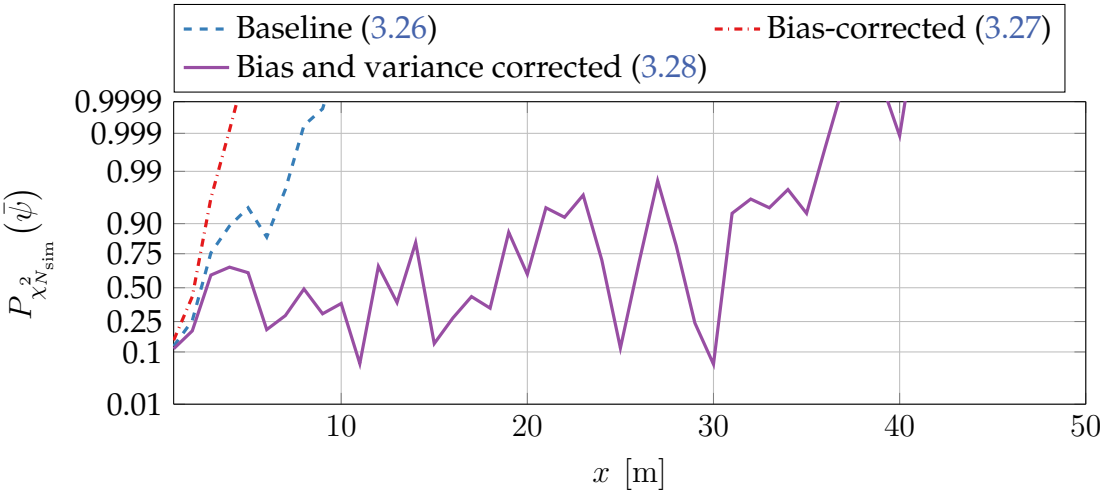


Figure 3.8 The normalised error squared (3.29) is used for consistency check of the transformation of the Gaussian distribution from disparity d to distance x. Two variants are compared to the baseline approach (3.26). These are a bias-corrected mean according to (3.27) and in conjunction with the proposed transformation of the variance (3.28). Widely consistent results are only achieved when taking bias and uncertainty in the linearisation into account.

The NES should be distributed according to a $\chi^2_{N_{\text{sim}}}$ distribution with N_{sim} degrees of freedom for a sample of N_{sim} independent Gaussian estimates $\hat{x}^{(i)} \sim \mathcal{N}\left(\mu_x, \sigma_x^{(i)^2}\right)$. Therefore, a realisation of $\bar{\psi}$ from a simulation run can be compared to the corresponding value of the cumulative distribution function $P_{\chi^2_{N_{\text{sim}}}}(\bar{\psi})$. A result that lies outside of an acceptance interval $[0.5\alpha, 1 - 0.5\alpha]$ is only expected in a proportion α of all cases.

This test gives an objective measure whether a particular transformation yields consistent distribution parameters. The results in Fig. 3.8 show that only the combination of a bias-corrected mean together with a transformation of the variance according to (3.28) produces mostly consistent results.

The scope of the aforementioned derivation is a consistent transformation of the Gaussian probability density $\mathcal{N}\left(\hat{d}; \mu_d, \sigma_d^2\right)$ of disparity values to an approximate Gaussian $\mathcal{N}\left(\hat{x}; \mu_x, \sigma_x^2\right)$ of the Cartesian distance. In the fashion of [8, 168, 179], an alternative bias-corrected transformation formula for the disparity values $\hat{d}$ can be used in the first place:

$$\hat{x} = \frac{c_{\text{k}} b_{\text{w}}}{\hat{d}} \left(1 - \frac{\sigma_d^2}{\hat{d}^2}\right) . \tag{3.30a}$$

With a similar approach as before, a transformation of the variance which takes into account the uncertainty of the linearisation point can be derived:

$$\sigma_x^2 = \left(\frac{c_{\text{k}} b_{\text{w}}}{\hat{d}^2}\right)^2 \left[\left(1 - \frac{3\sigma_d^2}{\hat{d}^2}\right)^2 + 2\left(5 - 63\frac{\sigma_d^2}{\hat{d}^2} + 162\frac{\sigma_d^4}{\hat{d}^4}\right)\frac{\sigma_d^2}{\hat{d}^2}\right]\sigma_d^2 . \tag{3.30b}$$

3.1.5 Evaluation

Having introduced the proposed framework for statistical modelling, a summative evaluation is now pursued. The model predictions are therefore compared to the results of the exemplary detection algorithm from Sec. 3.1.3 when applied to experimental stereo video recordings. It will be analysed how well the model explains the experimental data in terms of the objective function's maximum value $g_{\max}$ and the standard deviation of the estimated vehicle position $\hat{x}_{\mathrm{obj}}$.

The model relies on knowledge of the stereo video measurement model as presented in Sec. 3.1.4.1 and the object contour as well as height from Sec. 3.1.4.2. Some of these parameters are in general time varying, for example the measurement noise of vision-based systems depends on the lighting or weather conditions. Moreover, the object appearance differs among vehicle types. It is thus necessary to discuss whether these dependencies are reasonable to include and how unmodeled variations affect the results.

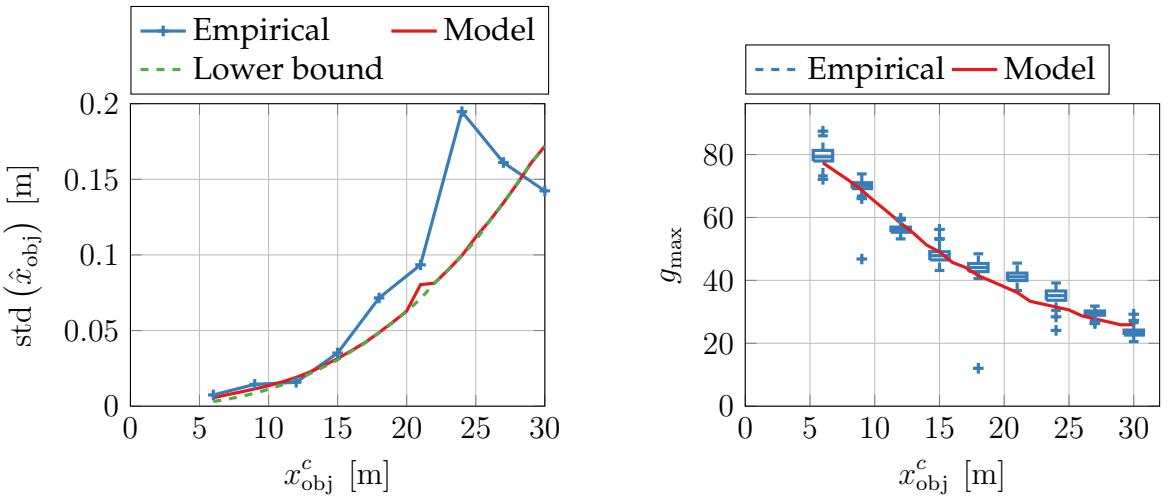
The first aspect will be answered using the recordings of a static scene with the identical vehicle positioned at different distances on a test track. With this setup, a minimal variation of model parameters is ensured. Measurements from the closest distance are used to calibrate the model and its generalisation to the remaining data is evaluated in Sec. 3.1.5.1. Secondly, dynamic recordings from an urban environment are analysed in Sec. 3.1.5.2. These comprise exposure to varying lighting conditions as well as different vehicles.

3.1.5.1 Static scenes

For the first evaluation, a stereo video sensor records sequences of a standing vehicle with height $h_{\mathrm{obj}} = 1.4\,\mathrm{m}$ in rear view at nine different positions $x^w_{\mathrm{obj}} \in \{6\,\mathrm{m}, 9\,\mathrm{m}, \ldots, 30\,\mathrm{m}\}$. Each sequence encompasses approximately 230 frames. Only the disparity measurements from $n_{\Delta u} = 10$ central image columns which comprise the vehicle rear are used in the objective function (3.5). Here, the Gaussian window is parametrised with $t_1 = 0.5$. The parametrisation of the contour model is estimated from measurements taken at $x^w_{\mathrm{obj}} = 6\,\mathrm{m}$ as detailed in Sec. 3.1.4.2. The resulting model has been shown in Fig. 3.5.

The object detection algorithm is applied to all images individually and the results are aggregated per sequence in terms of the standard deviation in $\hat{x}_{\mathrm{obj}}$. Since the distance remains constant per sequence, the sample average of the estimates is used to calculate the standard deviation [157]. The empirical results are visualised in Fig. 3.9 and compared to the model predictions. Overall, a good fit between modelled and observed values is obtained. However, a slight underestimation of the standard deviation can be recognised in Fig. 3.9(a). This can be attributed to a stochastic variation of the number of valid disparity measurements per image which was assumed as a constant ratio η_{valid} here. The mean values of $g_{\max}$ fit well to the analytical expectation.

Studying the relationship between $g_{\max}$ and distance in Fig. 3.9(b), especially for distances below $15\,\mathrm{m}$, reveals when it is necessary to explicitly model the vehicle contour. It has been previously shown in Fig. 3.3(b) that the number of image rows $n_{\Delta v,\mathrm{obj}}$ features



(a) Standard deviation in distance estimate. **(b)** Maximum value of objective function.

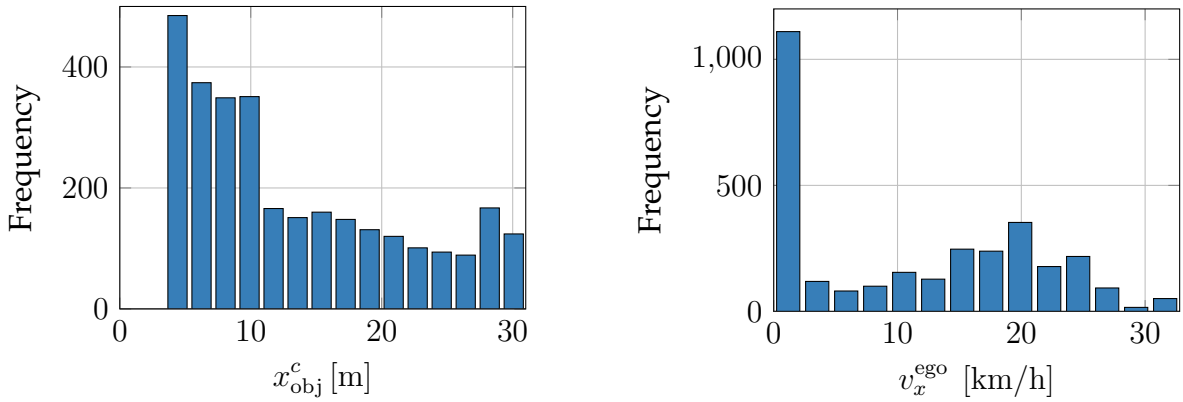
Figure 3.9 Evaluation results in static scenes on a test track compared to the analytical model from Sec. 3.1.4.3: (a) shows the standard deviation of the estimated distances. A slight underestimation at higher distances can be observed which is supposedly caused by additional unmodeled deteriorations of the image quality. The correspondence measure of the detection algorithm is visualised in (b). A good fit to the model is obtained since height and contour of the observed vehicle are known.

a hyperbolic dependence on the distance. However, although the objective function (3.5) integrates the measurements from all image rows *additively*, Fig. 3.9(b) does not reveal a similar, proportional characteristic. This effect can be explained by the influence of the vehicle's depth profile which causes wider disparity distributions at short distances.

3.1.5.2 Dynamic scenes

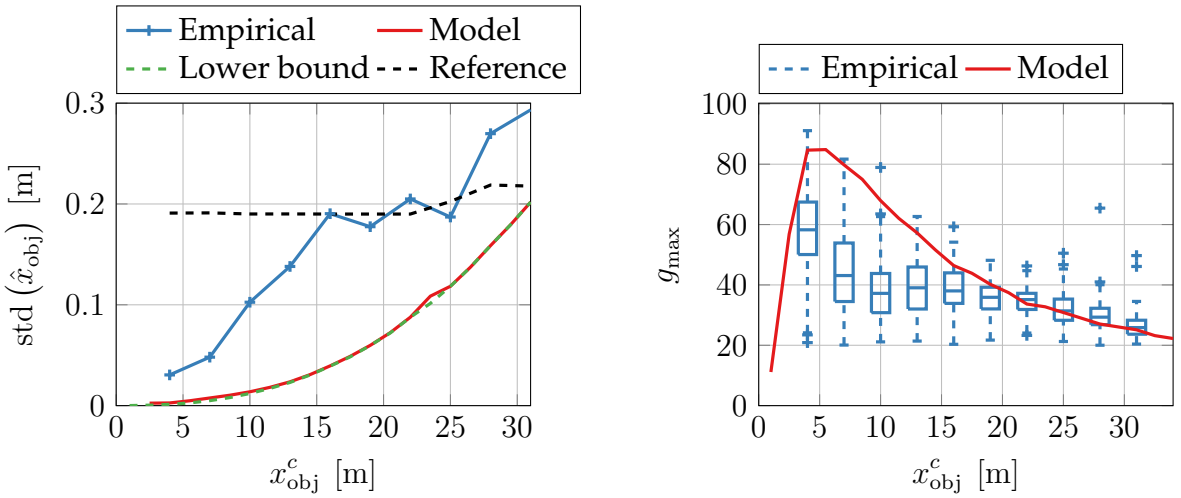
In a second evaluation, recordings from urban traffic taken out of a moving vehicle are used. From approximately 18 min of raw recordings, $N_{\mathrm{frames}} = 3088$ images with a preceding vehicle in the stereo sensor's field of view are selected for the evaluation. This experimental dataset is visualised in terms of distances and driven velocity in Fig. 3.10.

In order to obtain reference values for the distances to these vehicles, a laser scanner sensor is run in parallel. Further details on the generation of accurate reference measurements will be given in Ch. 6. However, the differences between the two measurement principles may cause systematic inaccuracies in the comparison. While the laser scanner approximates the entire vehicle by a rectangular shape and estimates the distance to its centroid, the stereo vision detections only represent a single point on a vehicle's surface. Therefore, a bias between both measurements may arise. Moreover, the effect is likely to vary over time in a dynamic scene with changing viewing angle. Similar issues are reported in [152]. Because a model-based approach to remove this bias from the evaluation requires unknown information on all objects' geometries, a moving average filter is used to estimate and remove the bias over a window of 30 frames.



(a) Distances. **(b)** Velocity of the ego-vehicle.

Figure 3.10 Visualisation of the evaluation dataset in terms of the distance to a preceding vehicle (a) and the velocity of the ego-vehicle (b). A high number of frames contain low distances at low speeds which is caused by urban traffic jams.



(a) Standard deviation in distance estimate. **(b)** Maximum value of objective function.

Figure 3.11 Evaluation results in dynamic scenes.

Due to the inherent distance dependence, all results are evaluated separately in bins with size $3\,\text{m}$ each. For each bin, the standard deviation $\text{std}\left(\hat{x}_{\text{obj}} - x_{\text{obj}}\right)$ is calculated with the laser scanner measurements as reference value x_{obj}. Similarly, the distribution of the maximum correspondence measure values $g_{\max}$ is evaluated per bin.

The results are depicted in Fig. 3.11. Firstly, a much higher standard deviation in the distance estimates is visible in Fig. 3.11(a), compared to the model values and the previous evaluation on static scenes. However, this increase can be partly attributed to inaccuracies in the reference measurements given by the laser scanner. The self-reported standard deviation indicates a non-neglectable uncertainty. Under the assumption that both sensors are statistically independent, the errors will add up when calculating the differences. Whereas the model only considers the stereo vision part.

Secondly, fundamental limitations of the modelling approach are seen from the distribution of the objective function's maximum as shown in Fig. 3.11(b). Stronger variations and deviations from the model predictions compared to static scenes with a single vehicle can be observed. These deviations are supposedly caused by the diversity of vehicles with varying height and contour that are encountered. The model takes only a single contour into account and thus fails to accurately describe the variation in the empirical data. These deviations are the worst at short distances and decrease for higher distances, where the influence of an object's depth profile is reduced.

3.1.6 Summary

This section has addressed environment perception based on a stereo video sensor. To this end, an exemplary algorithm for the detection of vehicles in dense stereo images has been introduced. The exemplary algorithm relies on the intentionally simple principle of a column-wise disparity aggregation.

For this algorithm, methods for analytical probabilistic modelling have been derived and evaluated in numerical simulations. The dividend of analytical models is that insight beyond a concrete parametrisation of an algorithm can be obtained. This has been exemplified by finding a lower bound on the variance of the estimated object position.

A second contribution concerns the transformation of measurements in disparity space to Cartesian coordinates in general. Previous works have studied this topic with regard to a bias that is due to the non-linear transformation. In this work, the consistency of a transformed estimate and its variance has been analysed with a statistical test. It has been shown that a linearisation-based transformation of the variance leads to widely inconsistent estimates if the uncertainty in the linearisation point is not taken into account. Therefore, a novel transformation formula with improved consistency has been proposed.

The derived model has been evaluated on two different datasets. Recordings with a static vehicle on a test track show the model's validity under situations with known conditions. Furthermore, these results emphasise that an accurate model of the detection algorithm requires modelling of the observed object's geometry, in this case the depth profile and height. However, evaluations on urban traffic scenes show that additional unmodeled variations, such as the diverse appearance of different vehicles, can have a significant effect.

Concerning the question whether it is possible to extend the presented modelling approach to more complex full-scale algorithms, it can be concluded that:

1. A linearisation-based variance propagation (as applied in Sec. 3.1.4.3) requires that an algorithm's output quantities can be formulated at least implicitly in terms of a twice differentiable cost function.

2. A column-wise image aggregation, as it is also performed by the approaches from [8, 12, 152], reduces the dimensions of the algorithm's input space and thus facilitates

the model development. The requirements on developing and parametrising such a model depend on the intended application and desired model accuracy.

Parametrising the model is certainly facilitated for applications where solely a small and well-defined set of objects are to be considered at all. One example is landmark-based localisation which will be studied in the following Sec. 3.2. The goal is to accurately localise the ego-vehicle by measuring the position of known, geo-referenced landmarks in the vehicle's surroundings, for example sign posts.

Instead of a model-based approach to characterising an algorithm for environment perception, the algorithm's behaviour can be evaluated empirically. To this end, a reference representation of the environment is needed. It will be analysed in Ch. 6 how such reference measurements can be obtained.

3.2 Feature-based localisation

This section develops analytical models of uncertainty in a second perception task, that is localisation of the ego-vehicle. First, an introduction to the relevance of feature-based localisation for automated vehicles is given in Sec. 3.2.1. After a review of related works on uncertainty modelling in Sec. 3.2.2, the localisation task is formulated in Sec. 3.2.3. The main contributions of this section are analytical models of uncertainty in the pose estimate which are derived in Sec. 3.2.4. These models are then evaluated and discussed with a numerical example in Sec. 3.2.5 before the section concludes with a summary in Sec. 3.2.6.

3.2.1 Introduction

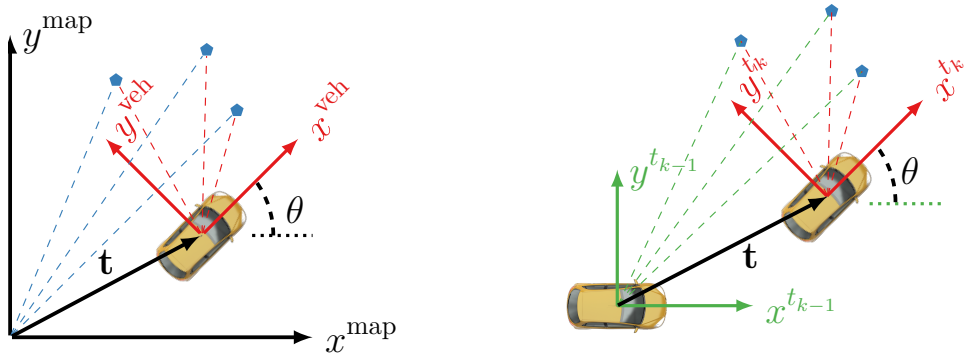
Accurately knowing the ego-vehicle's current pose is mandatory for a number of assistance functions with higher degrees of automation. For example, to navigate safely to a defined destination, an intelligent vehicle has to plan a trajectory that obeys traffic rules and thus needs to know its own position with at least lane-level accuracy. Moreover, if the ego-vehicle is localised on a detailed map of the road network, lane-information can be associated to other nearby vehicles. This may serve as valuable prior information for a prediction of the other vehicles' future trajectories, for instance at intersections [151].

To achieve a highly accurate localisation, satellite navigation alone is considered insufficient. Especially in urban environments, the localisation performance deteriorates due to multipath wave propagation of the satellite signals caused by reflections at buildings [116]. At certain places, for example in parking decks, satellite navigation might be entirely unavailable.

Therefore, feature-based localisation with exteroceptive sensors is a relevant research topic in the context of intelligent vehicles. The basic principle is to measure the positions of static, uniquely identifiable environment features relative to the vehicle. This is visualised in Fig. 3.12(a). Then, the vehicle's pose relative to a known map is estimated by comparing the relative feature positions to the corresponding locations in the map (map matching) [115, 155]. If a map is unavailable a-priori, one can alternatively estimate the relative displacements of the vehicle over time (pose tracking, see Fig. 3.12(b)). These incremental trajectory estimates can be fused with locally less accurate satellite-based position information to achieve globally accurate localisation.[4]

The core approach of feature-based localisation is twofold: At first, environment features have to be perceived and correspondences between the known landmarks from a map or previous observations have to be established. In a second step, the transformation parameters (rotation and translation) between both sets have to be estimated. The latter problem is usually independent of the employed sensor technology.

[4] A third method, that is not discussed in this work, is to perform pose tracking and store the measured features in a map which allows recognising places that have already been visited (simultaneous localisation and mapping, SLAM).



(a) Map matching.

(b) Pose tracking over time-steps t_{k-1} and t_k.

Figure 3.12 The core principle of feature-based (a) global localisation (map matching) and (b) relative localisation (pose tracking) is to estimate the transformation parameters between two different coordinate systems. These are the translation vector $\mathbf{t}$ and the rotation angle θ.

Since sensor measurements are affected by random noise, knowing the statistical properties of the estimated vehicle pose allows deriving sensor requirements. For example, the uncertainty is likely to depend on the size of the sensor's field of view since this influences the number of landmarks from which the pose estimate can be calculated. The relation between sensor parameters and achievable localisation accuracy will be studied for the specific example of a stereo vision sensor in Sec. 7.1.

Therefore, this section sets out to develop closed-form probabilistic models of uncertainty in pose estimates. In contrast to numerical Monte-Carlo methods, analytical models help to understand the influence of system parameters and to facilitate real-time sensor data fusion. The scope is the sensor-independent second step in the feature-based localisation framework, the estimation of the vehicle pose. To this end, it is assumed that the correspondences between the sets of features are ideally known and measurement noise can be modelled with Gaussian distributions. Moreover, under the usual assumption in the intelligent vehicles domain of a planar world, the problem is formulated in two-dimensional Cartesian coordinates. This greatly simplifies the estimation and yields explicit formulations of the probabilistic models.

In order to obtain an optimal estimate of the pose, the measurements' statistical properties have to be taken into account. This leads to different problem formulations with matrix or scalar weights. In the context of this dissertation, statistical models of the pose estimation uncertainty have been previously published for the unweighted problem in [217], for the scalar-weighted case in [211] and for the matrix-weighted case in [212]. In the following, the matrix-weighted problem is analysed first and other cases are subsequently derived from this general case.

3.2.2 Related work

Estimating a pose change from two sets of corresponding features is a relevant task in many fields and known as the *Procrustes problem*. Different solution formulae and approximate algorithms have been developed [6, 86, 87, 127, 197], see also [8] for a comprehensive review. Together with the problem formulation, these solutions will be further examined in Sec. 3.2.3.

Despite the widespread practical relevance of the Procrustes problem, only few works have been dedicated to probabilistic modelling of its solution under noisy input data. For the unweighted problem in 3-D, DORST [48] derives a first order series expansion and thus a closed-form expression for the variance. In this thesis, solely the 2-D case is considered. This allows developing a second order expansion for the scalar-weighted Procrustes problem in a compact form. Thus, as an extension of previous works, not only the variance but also the bias will be studied in Sec. 3.2.4.3.

The matrix-weighted Procrustes problem requires a conceptually different approach though, since no closed form of the estimate has been obtained to date [8]. A suitable method for Gaussian uncertainty propagation, that is applied to stereo vision sensors in [54, 205] and to laser range finders in [36], is to obtain the required Jacobians using the implicit function theorem. Along the same line, explicit results will be derived for a sensor-independent problem formulation in Sec. 3.2.4.2. A novelty in the derivation is given by a reformulation of the problem in terms of a centering operator. Properties of this operation are first proven in the general case and then applied to the specific expressions in order to systematically obtain a concise final result.

Probabilistic analyses of an estimated pose change are not to be confused with studies on error growth over entire trajectories. A number of works, for example [9, 51, 110, 147, 204], consider the question of how errors in the incremental pose change estimates accumulate to errors in a concatenated pose. However, previous works that aim for analytical models [110, 204] often assume a simple concatenation of pose increments (dead-reckoning), an approach that is well known to suffer from long-term instability.

More sophisticated system designs, for example based on Bayesian state estimation, are studied either in simulations [147] or empirically [9, 51, 68]. To obtain analytical models, the work [140] derives upper bounds on the expected covariance. In this thesis, the Bayes filters will be later analysed in a generic formulation in Sec. 4.1. The incremental pose updates before the filtering stage on the other are in the scope of the current chapter.

3.2.3 Problem formulation

This section details algorithms for pose change estimation whose statistical properties are later analysed in Sec. 3.2.4. It consists of three parts. The notation and a formalisation of the localisation and pose tracking problems will be first introduced in Sec. 3.2.3.1. Depending on the noise characteristics – anisotropic or isotropic – two different estimation problems with matrix or scalar weights arise. These will be subsequently outlined in Sec. 3.2.3.2 and Sec. 3.2.3.3.

3.2.3.1 Preliminaries

Two registered sets $\mathcal{P} = \{\,\mathbf{p}_i\,\}_{i=1:N}$ and $\mathcal{Q} = \{\,\mathbf{q}_i\,\}_{i=1:N}$ of $i = 1, \ldots, N$ feature positions $\mathbf{p}_i, \mathbf{q}_i \in \mathbb{R}^2$ in 2-D Cartesian coordinates form the basis of the estimation task. The relationship between both sets is given by a transformation that consists[5] of a rotation by an angle θ and a translation by $\mathbf{t} \in \mathbb{R}^2$:

$$\mathbf{q}_i = \mathbf{R}\mathbf{p}_i + \mathbf{t}\,, \qquad \mathbf{R} := \begin{bmatrix} c & -s \\ s & c \end{bmatrix} = \begin{bmatrix} \cos(\theta) & -\sin(\theta) \\ \sin(\theta) & \cos(\theta) \end{bmatrix}\,. \tag{3.31}$$

Given $\mathcal{P}$ and $\mathcal{Q}$, the task of estimating $(\theta, \mathbf{t})$ is known as the Procrustes problem.[6] The interpretation of these parameters depends on the application:

- In the context of global localisation against a known map (Fig. 3.12(a)), $\mathbf{p}_i$ are feature positions in the ground-fixed coordinate system of the map. The same features, perceived in the moving coordinate system of the ego-vehicle, are denoted by $\mathbf{q}_i$. Thus, the translation $\mathbf{t}$ is the vehicle's absolute position in the map and θ its orientation.
- In the case of a pose tracking task (Fig. 3.12(b)), $\mathbf{p}_i, \mathbf{q}_i$ stand for feature positions in the coordinate system of the sensor from two different time steps. Then, θ and $\mathbf{t}$ define changes in orientation and position of the ego-vehicle.

Without noise, solving for the three transformation parameters is trivial if at least two feature vectors are known (assuming that the problem is not ill-posed). In the relevant case of noisy measurements though, a parameter estimation problem has to be defined and solved. It is assumed that the measurement errors are additive, mutually independent, have zero mean and follow known Gaussian distributions. Thus, $\mathbf{p}_i \sim \mathcal{N}\big(\boldsymbol{\mu}_{\mathbf{p}_i}, \boldsymbol{\Sigma}_{\mathbf{p}_i}\big)$ and $\mathbf{q}_i \sim \mathcal{N}\big(\boldsymbol{\mu}_{\mathbf{q}_i}, \boldsymbol{\Sigma}_{\mathbf{q}_i}\big)$.[7] Then, a suitable approach to estimate $(\theta, \mathbf{t})$ is to minimise the weighted sum of the quadratic residuals J_i [146]:

$$J = \sum_{i=1}^{N} \underbrace{(\mathbf{q}_i - \mathbf{R}\mathbf{p}_i - \mathbf{t})^{\mathrm{T}}\, \mathbf{W}_i\, (\mathbf{q}_i - \mathbf{R}\mathbf{p}_i - \mathbf{t})}_{=:J_i}\,, \qquad \mathbf{W}_i = \left(\mathbf{R}\boldsymbol{\Sigma}_{\mathbf{p}_i}\mathbf{R}^{\mathrm{T}} + \boldsymbol{\Sigma}_{\mathbf{q}_i}\right)^{-1}\,. \tag{3.32}$$

The symmetric weighting matrices $\mathbf{W}_i$ are inversely proportional to the covariance of the measurement errors.

It should be remarked that while this approach resembles the familiar definition of a least squares estimator, it is not in general an optimal approach if noise is prevalent in

[5] Most of the subsequent derivations can be similarly performed if an additional scale change is part of the transformation. However, this is not relevant for the applications that are considered in this work.

[6] Instead of estimating the angle θ, it is often more straight-forward to solve the equivalent problem of finding the entries c, s of the rotation matrix $\mathbf{R}$, subject to the constraint that $c^2 + s^2 = 1$.

[7] Throughout this chapter, the general case that both $\mathbf{q}_i$ and $\mathbf{p}_i$ are affected by noise is assumed. In a localisation task, the $\mathbf{p}_i$ constitute the a-priori known map and it can be reasoned that these are given without uncertainty. An application example corresponding to this case will be later studied in Sec. 7.1.

both variables $\mathbf{q}_i$ and $\mathbf{p}_i$. The correct way of handling such errors-in-variables problems would be a *total* least squares approach. However, for the Procrustes problem, the cost function of a total least squares estimator reduces to (3.32) [146].

In the following, it will be discussed how J can be minimised. At first, the general case of arbitrary covariance matrices (anisotropic noise) is studied, which unfortunately does not allow for an exact closed-form solution. Secondly, the special case of equal variance in all coordinates (isotropic noise) is analysed and analytical solution formulae are reviewed.

3.2.3.2 Anisotropic noise: Matrix-weighted Procrustes problem

An important intermediate result towards minimising the matrix-weighted objective function (3.32) is the matrix-weighted centroid-coincidence theorem:

Theorem 3.1 (Matrix-weighted centroid-coincidence theorem [197])
If the weighting matrices $\mathbf{W}_i$ do not depend on the unknown parameters $(\mathbf{R}, \mathbf{t})$, then the estimates $(\hat{\mathbf{R}}, \hat{\mathbf{t}})$ which minimise (3.32) satisfy the following condition:

$$\sum_{i=1}^{N} \mathbf{W}_i \mathbf{q}_i = \sum_{i=1}^{N} \mathbf{W}_i \left(\hat{\mathbf{R}} \mathbf{p}_i + \hat{\mathbf{t}} \right) \ . \tag{3.33}$$

Thus, an explicit expression for the optimal translation estimate $\hat{\mathbf{t}}$ is obtained, given that the rotation matrix $\hat{\mathbf{R}}$ is known:

$$\hat{\mathbf{t}} = \left(\sum_{i=1}^{N} \mathbf{W}_i \right)^{-1} \cdot \left(\sum_{i=1}^{N} \mathbf{W}_i \mathbf{q}_i - \mathbf{W}_i \hat{\mathbf{R}} \mathbf{p}_i \right) \ . \tag{3.34}$$

Noting that the first factor serves as a normalisation constant, the abbreviation $\mathbf{W}_\mathrm{s} := \sum_{i=1}^{N} \mathbf{W}_i$ is used hereafter to refer to this sum of the weighting matrices.

In order to apply this theorem to (3.32), it is suggested in [197] to obtain a tentative estimate of $\hat{\mathbf{R}}$ first, which is only used to define the matrix weights $\mathbf{W}_i$. These become independent of $\mathbf{R}$ then. Another suggestion, relevant to pose tracking tasks, is to assume that only small rotations occur and therefore to approximate $\mathbf{R} \approx \mathbf{I}_{2\times 2}$ in (3.32).

Now, (3.34) is inserted into (3.32), which yields the following cost function:

$$J = \sum_{i=1}^{N} \mathbf{a}_i^\mathrm{T} \mathbf{W}_i \mathbf{a}_i \ , \quad \mathbf{a}_i = \underbrace{\left(\mathbf{q}_i - \mathbf{W}_\mathrm{s}^{-1} \sum_{j=1}^{N} \mathbf{W}_j \mathbf{q}_j \right)}_{=: \mathbf{c}_i\left(\{\mathbf{q}_j\}_{j=1:N}\right)} - \underbrace{\left(\mathbf{R}\mathbf{p}_i - \mathbf{W}_\mathrm{s}^{-1} \sum_{j=1}^{N} \mathbf{W}_j \mathbf{R}\mathbf{p}_j \right)}_{=: \mathbf{c}_i\left(\{\mathbf{R}\mathbf{p}_j\}_{j=1:N}\right)} \ . \tag{3.35}$$

As a means for keeping the upcoming derivations concise and interpretable, a centering operator $\mathbf{c}_i(\cdot)$ is introduced. This set function maps a set $\{\mathbf{p}_j\}_{j=1:N}$ of N vectors $\mathbf{p}_j \in \mathbb{R}^n$ on a vector $\mathbf{c}_i(\cdot) \in \mathbb{R}^n$.[8] Given a set of symmetric weighting matrices $\mathbf{W}_j$ of appropriate dimensions and let the sum thereof be denoted by $\mathbf{W}_\mathrm{s}$, the centering operation subtracts the centroid from the ith element of the set of vectors:

$$\mathbf{c}_i\big(\{\mathbf{p}_j\}_{j=1:N}\big) \coloneqq \mathbf{p}_i - \mathbf{W}_\mathrm{s}^{-1} \sum_{j=1}^{N} \mathbf{W}_j \mathbf{p}_j \ . \tag{3.36}$$

Before proceeding to solve the Procrustes problem, relevant properties of this function will be briefly collected in the following.

1. **Additivity:** Since the matrix product is distributive, it holds that

$$\mathbf{c}_i\big(\{\mathbf{p}_j + \mathbf{q}_j\}_{j=1:N}\big) = \mathbf{c}_i\big(\{\mathbf{p}_j\}_{j=1:N}\big) + \mathbf{c}_i\big(\{\mathbf{q}_j\}_{j=1:N}\big) \ . \tag{3.37}$$

2. **Scale invariance:** With a scalar factor $\alpha \in \mathbb{R}$ it holds that

$$\mathbf{c}_i\big(\{\alpha\mathbf{p}_j\}_{j=1:N}\big) = \alpha\mathbf{c}_i\big(\{\mathbf{p}_j\}_{j=1:N}\big) \ . \tag{3.38}$$

   In the special case where the weighting matrices $\mathbf{W}_j$ are scalars w_j, $j = 1, \ldots, N$, this property furthermore extends to matrix scale factors $\mathbf{A} \in \mathbb{R}^{n\times n}$.

3. **Offset invariance:** As the name of the centering function indicates, its result is invariant under a constant offset $\mathbf{b} \in \mathbb{R}^n$:

$$\mathbf{c}_i\big(\{\mathbf{p}_j + \mathbf{b}\}_{j=1:N}\big) = \mathbf{c}_i\big(\{\mathbf{p}_j\}_{j=1:N}\big) \ . \tag{3.39}$$

   This follows from (3.36) since $\mathbf{W}_\mathrm{s}^{-1} \sum_{j=1}^{N} \mathbf{W}_j = \mathbf{I}_{n\times n}$.

4. **Concatenation:** Applying (3.36) on a previously centred set $\mathcal{C} = \{\mathbf{c}_i(\mathcal{P})\}_{i=1:N}$ is equal to the same operation on the original set $\mathcal{P}$:

$$\begin{aligned}
\mathbf{c}_k(\mathcal{C}) &= \mathbf{c}_k(\mathcal{P}) - \mathbf{W}_\mathrm{s}^{-1} \sum_{l=1}^{N} \mathbf{W}_l \mathbf{c}_l(\mathcal{P}) \\
&= \mathbf{c}_k(\mathcal{P}) - \mathbf{W}_\mathrm{s}^{-1} \sum_{l=1}^{N} \mathbf{W}_l \left(\mathbf{p}_l - \mathbf{W}_\mathrm{s}^{-1} \sum_{j=1}^{N} \mathbf{W}_j \mathbf{p}_j \right) \\
&= \mathbf{c}_k(\mathcal{P}) - \underbrace{\mathbf{W}_\mathrm{s}^{-1} \left(\sum_{l=1}^{N} \mathbf{W}_l \mathbf{p}_l - \sum_{j=1}^{N} \mathbf{W}_j \mathbf{p}_j \right)}_{=\mathbf{0}} \ .
\end{aligned} \tag{3.40}$$

[8] In the context of this chapter, the operation is used only for two-dimensional inputs $\mathbf{p}_j \in \mathbb{R}^2$.

We now return to the estimation task. The goal is to minimise the cost function (3.35) subject to the condition that $\mathbf{R}$ defines a rotation matrix. Unfortunately, $\mathbf{R}$ is a non-linear function of the rotation angle θ and the optimisation cannot be solved in closed form. Thus, one has to find the minimum either numerically or content oneself with an approximately optimal estimate. One possible approximation is based on an unconstrained estimate of $\mathbf{R}$ as the solution of a linear least squares problem. This initial estimate is then used to find the closest true rotation matrix [197].

Nevertheless, although an explicit solution is not available, a model of the variance of the optimal estimate can be obtained. This and the subsequent propagation to the translation estimate $\hat{\mathbf{t}}$ from (3.34) will be derived in Sec. 3.2.4.2.

3.2.3.3 Isotropic noise: Scalar-weighted Procrustes problem

A scalar-weighted cost function arises from (3.32) for the case of isotropic noise. That is, the error covariances read $\mathbf{\Sigma}_{\mathbf{p}_i} = \sigma_{p_i}^2 \mathbf{I}_{2\times 2}$ and $\mathbf{\Sigma}_{\mathbf{q}_i} = \sigma_{q_i}^2 \mathbf{I}_{2\times 2}$. This special case of the matrix-weighted Procrustes problem has the practically relevant property that an exact closed-form solution exists.[9]

Firstly, note that the weights[10] $\mathbf{W}_i = w_i \mathbf{I}_{2\times 2} = \left(\sigma_{p_i}^2 + \sigma_{q_i}^2\right)^{-1} \mathbf{I}_{2\times 2}$ are always independent of $\mathbf{R}$ now and thus, the requirements of theorem 3.1 are fulfilled. As a second consequence, it follows that the centering operator is scale invariant (3.38) with respect to a matrix multiplication, that is $\mathbf{c}_i\left(\left\{\mathbf{R}\mathbf{p}_j\right\}_{j=1:N}\right) = \mathbf{R}\mathbf{c}_i(\mathcal{P})$. Expanding the cost function (3.35) yields:

$$J = \sum_{i=1}^{N} w_i \left[\mathbf{c}_i^{\mathrm{T}}(\mathcal{Q})\,\mathbf{c}_i(\mathcal{Q}) - 2\mathbf{c}_i^{\mathrm{T}}(\mathcal{Q})\,\mathbf{R}\mathbf{c}_i(\mathcal{P}) + \mathbf{c}_i(\mathcal{P})^{\mathrm{T}}\,\mathbf{c}_i(\mathcal{P})\right] . \tag{3.41}$$

Since the first and third term are independent of $\mathbf{R}$, these can be neglected. Therefore, (3.41) reduces to

$$J = -\sum_{i=1}^{N} w_i \mathbf{c}_i^{\mathrm{T}}(\mathcal{Q})\,\mathbf{R}\mathbf{c}_i(\mathcal{P}) = -\operatorname{tr}\left(\sum_{i=1}^{N} w_i \mathbf{c}_i(\mathcal{P})\,\mathbf{c}_i^{\mathrm{T}}(\mathcal{Q})\,\mathbf{R}\right) = -\operatorname{tr}\left(\mathbf{\Sigma}_{\mathcal{P}\mathcal{Q}}\mathbf{R}\right) . \tag{3.42}$$

The notation of $\mathbf{\Sigma}_{\mathcal{P}\mathcal{Q}}$ is motivated by the fact that $\sum_{i=1}^{N} \mathbf{c}_i(\mathcal{P})\,\mathbf{c}_i^{\mathrm{T}}(\mathcal{Q})$ is the scatter matrix of the vectors in $\mathcal{P}$ and $\mathcal{Q}$. If a normalisation with $w_i = \frac{1}{N-1}$ is chosen, this yields an estimate of the cross-covariance $\operatorname{cov}(\mathbf{p}, \mathbf{q})$.

Finding a rotation matrix $\mathbf{R}$ that minimises (3.42) can be achieved in a number of ways which lead to different forms of the same closed-form solution [52]. Commonly employed

[9] In scenarios, where the noise is in fact anisotropic but numerically solving the matrix-weighted problem from Sec. 3.2.3.2 is computationally infeasible, the approximations $\sigma_{p_i}^2 = \det\left(\mathbf{\Sigma}_{\mathbf{p}_i}\right)$ and $\sigma_{q_i}^2 = \det\left(\mathbf{\Sigma}_{\mathbf{q}_i}\right)$ can be employed [38]. Thereby, instead of taking the shape of the covariance ellipses into account, only the area information is considered.

[10] The scalar weights now serve the purpose to deal with heteroscedastic noise. If the noise processes are homoscedastic, that is $\sigma_{p_i}^2 = \sigma_p^2$ and $\sigma_{q_i}^2 = \sigma_q^2\ \forall i$, one obtains the unweighted Procrustes problem. In this case, the exact values of the variances are not needed for the estimation.

methods are based on decomposing the matrix $\mathbf{\Sigma}_{\mathcal{PQ}}$, for example with a singular value decomposition [6], a polar decomposition [87] or a quaternion representation [86]. These approaches are applicable to both the three- and the two-dimensional problem. With the method of Lagrange multipliers though, a more explicit solution can be obtained for the 2-D case [127], which will be briefly derived in the following.

To this end, $\mathbf{R}$ is expressed element-wise in terms of c and s as in (3.31). Then, the following Lagrangian is obtained from (3.42):

$$\mathcal{L}(c, s, \lambda) = -c \cdot \underbrace{\left(\Sigma_{\mathcal{PQ},11} + \Sigma_{\mathcal{PQ},22}\right)}_{=:f_1} - s \cdot \underbrace{\left(\Sigma_{\mathcal{PQ},12} - \Sigma_{\mathcal{PQ},21}\right)}_{=:f_2} + \lambda\left(c^2 + s^2 - 1\right) . \tag{3.43}$$

One can differentiate $\mathcal{L}(c, s, \lambda)$ with respect to its arguments and thus obtains estimates for the entries of $\mathbf{R}$:

$$\hat{c} = \frac{f_1}{\sqrt{f_1^2 + f_2^2}} , \qquad \hat{s} = \frac{f_2}{\sqrt{f_1^2 + f_2^2}} , \qquad \hat{\theta} = \arctan\left(\frac{f_1}{f_2}\right) . \tag{3.44a}$$

The translation estimate $\hat{\mathbf{t}}$ follows from the centroid-coincidence theorem (3.34):

$$\hat{\mathbf{t}} = \underbrace{w_{\mathrm{s}}^{-1} \sum_{i=1}^{N} w_i \mathbf{q}_i}_{=:\bar{\mathbf{q}}} - \hat{\mathbf{R}} \underbrace{w_{\mathrm{s}}^{-1} \sum_{i=1}^{N} w_i \mathbf{p}_i}_{=:\bar{\mathbf{p}}} . \tag{3.44b}$$

Thus, the translation is determined by the vector difference between the two centroids $\bar{\mathbf{q}}$ and $\bar{\mathbf{p}}$ with a change of orientation taken into account.

The simplicity of the solution (3.44) permits to easily develop a series expansion. This will be studied in Sec. 3.2.4.3, where expressions for the variance and bias of the estimates are derived and a novel debiased formula is proposed. Moreover, the explicit solution shows practical advantages over ones that involve matrix decompositions in terms of run time.[11] This has been analysed analytically and empirically in [217].

To conclude the problem formulation, an overview of the main characteristics of the Procrustes problem is given in Tab. 3.1.

3.2.4 Statistical modelling

In the following, statistical models of uncertainty in the pose estimates are developed. These present the main contributions of this chapter. Firstly, the overall approach will be laid out in Sec. 3.2.4.1. In the sequel, closed-form models are derived for the matrix-weighted Procrustes problem in Sec. 3.2.4.2 and the scalar-weighted case in Sec. 3.2.4.3.

[11] If the correspondences between features in the two sets are initially unknown and not provided by sensory information, a random sampling approach can be used to estimate both the registration and pose transformation [57]. Thus, solving the Procrustes problem has to be carried out not once per system time step but for a significant number of iterations. It is therefore advisable to perform the underlying estimation as efficiently as possible in order to a achieve real-time capable system [72].

Table 3.1 Comparison of the matrix- and scalar-weighted Procrustes problem.

	Matrix-weighted	**Scalar-weighted**
Noise model	Independent, heteroscedastic anisotropic Gaussian	Independent, heteroscedastic isotropic Gaussian
Objective function	(3.35)	(3.42)
Solution	▪ Iterative [197] ▪ Approximate [197]	▪ Singular value decomposition [6] ▪ Polar decomposition [87] ▪ Quaternion [86] ▪ 2-D: Lagrangian (3.44) [127]

3.2.4.1 Preliminaries

The previous section has reviewed algorithms for optimal estimation of the pose parameters $\left(\hat{\theta}, \hat{\mathbf{t}}\right)$ from two sets of feature position measurements $(\mathcal{P}, \mathcal{Q})$. Since the input quantities of the algorithms are assumed to be Gaussian random variables, the estimates are random as well. Thus, the purpose of this section is to find expressions for the estimation error variances $\sigma_\theta^2 := \operatorname{var}\left(\hat{\theta}\right)$ and $\mathbf{\Sigma}_\mathbf{t} := \operatorname{cov}\left(\hat{\mathbf{t}}\right)$.

The chosen approach is to linearise[12] the estimates around the expected values and propagate the covariance of the measurement errors to the estimate. In a first step it is thus necessary to find the gradients of $\hat{\theta}$ with respect to $\mathbf{p}_k$ and $\mathbf{q}_k$, $k = 1, \ldots, N$. Under the assumption of independent measurement errors, the variance then follows as:

$$\sigma_\theta^2 = \sum_{k=1}^{N} \left(\nabla_{\mathbf{p}_k} \hat{\theta}\right) \mathbf{\Sigma}_{\mathbf{p}_k} \left(\nabla_{\mathbf{p}_k} \hat{\theta}\right)^{\mathrm{T}} + \left(\nabla_{\mathbf{q}_k} \hat{\theta}\right) \mathbf{\Sigma}_{\mathbf{q}_k} \left(\nabla_{\mathbf{q}_k} \hat{\theta}\right)^{\mathrm{T}} . \tag{3.45}$$

In a second step, the covariance matrix of the estimated translation $\hat{\mathbf{t}}$ can be obtained by linearisation of (3.34):

$$\mathbf{\Sigma}_\mathbf{t} = \left(\frac{\partial}{\partial \theta} \hat{\mathbf{t}}\right) \sigma_\theta^2 \left(\frac{\partial}{\partial \theta} \hat{\mathbf{t}}\right)^{\mathrm{T}} + \sum_{k=1}^{N} \left(\nabla_{\mathbf{p}_k} \hat{\mathbf{t}}\right) \mathbf{\Sigma}_{\mathbf{p}_k} \left(\nabla_{\mathbf{p}_k} \hat{\mathbf{t}}\right)^{\mathrm{T}} + \left(\nabla_{\mathbf{q}_k} \hat{\mathbf{t}}\right) \mathbf{\Sigma}_{\mathbf{q}_k} \left(\nabla_{\mathbf{q}_k} \hat{\mathbf{t}}\right)^{\mathrm{T}} . \tag{3.46}$$

Having formulated the approach, the aim is to obtain the necessary derivatives in the following. Given that no closed-form solution exists for the matrix-weighted Procrustes problem – in contrast to the scalar-weighted one – the two cases are treated separately.

[12] According to the general approach (2.24), all derivatives are evaluated at the expected values. For notational brevity, an explicit indication in the form of $\frac{\partial}{\partial x}(\cdot)|_{\mu_x}$ is omitted in this section.

3.2.4.2 Matrix-weighted Procrustes problem

Due to the fact that a closed-form estimate $\hat{\theta}$ which minimises the objective function J from (3.35) is not available, the implicit function theorem is used to find the required derivatives [55]:

$$\nabla_{\mathbf{p}_k}\hat{\theta} = \left(\frac{\partial^2}{\partial\theta^2}J\right)^{-1} \cdot \left(\frac{\partial}{\partial\theta}\nabla_{\mathbf{p}_k}J\right) , \qquad \nabla_{\mathbf{q}_k}\hat{\theta} = \left(\frac{\partial^2}{\partial\theta^2}J\right)^{-1} \cdot \left(\frac{\partial^2}{\partial\theta}\nabla_{\mathbf{q}_k}J\right) \tag{3.47}$$

Since J is composed of a sum $\sum_{i=1}^{N} J_i$, the differentiation of the individual J_i is studied first. The first and second derivatives with respect to θ are obtained from (3.35):

$$\frac{\partial}{\partial\theta}J_i = 2\mathbf{a}_i^{\mathrm{T}}\mathbf{W}_i\frac{\partial}{\partial\theta}\mathbf{a}_i , \qquad \frac{\partial^2}{\partial\theta^2}J_i = 2\frac{\partial}{\partial\theta}\mathbf{a}_i^{\mathrm{T}}\mathbf{W}_i\frac{\partial}{\partial\theta}\mathbf{a}_i + 2\mathbf{a}_i^{\mathrm{T}}\mathbf{W}_i\frac{\partial^2}{\partial\theta^2}\mathbf{a}_i . \tag{3.48}$$

Thereby, the derivative of $\mathbf{a}_i$ from (3.35) with respect to θ is required:

$$\frac{\partial}{\partial\theta}\mathbf{a}_i = -\mathbf{c}_i\left(\left\{\mathbf{R}'\mathbf{p}_j\right\}_{j=1:N}\right) , \qquad \mathbf{R}' \coloneqq \begin{bmatrix} -\sin(\theta) & -\cos(\theta) \\ \cos(\theta) & -\sin(\theta) \end{bmatrix} . \tag{3.49}$$

Note that the derivatives in (3.45) are to be evaluated at the expected values. Thus, it holds that $\mathbf{a}_i = \mathbf{0}$ in (3.48):

$$\mathbf{a}_i \overset{(3.37),(3.39)}{=} \mathbf{c}_i\left(\left\{\boldsymbol{\mu}_{\mathbf{q}_j} - \mathbf{R}\boldsymbol{\mu}_{\mathbf{p}_j} - \mathbf{t}\right\}_{j=1:N}\right) = \mathbf{0} . \tag{3.50}$$

With these two results, the final expression is obtained by summing up the individual derivatives from (3.48):

$$\frac{\partial^2}{\partial\theta^2}J = 2\sum_{i=1}^{N}\mathbf{c}_i^{\mathrm{T}}\left(\left\{\mathbf{R}'\mathbf{p}_j\right\}_{j=1:N}\right)\cdot\mathbf{W}_i\cdot\mathbf{c}_i\left(\left\{\mathbf{R}'\mathbf{p}_j\right\}_{j=1:N}\right) . \tag{3.51}$$

Secondly, the mixed derivatives in (3.47) are calculated:

$$\begin{aligned}\frac{\partial}{\partial\theta}\nabla_{\mathbf{p}_k}J_i &= 2\left(\mathbf{W}_i\frac{\partial}{\partial\theta}\mathbf{a}_i\right)^{\mathrm{T}}\left(\nabla_{\mathbf{p}_k}\mathbf{a}_i\right) + 2\mathbf{a}_i^{\mathrm{T}}\left(\nabla_{\mathbf{p}_k}\mathbf{W}_i\frac{\partial}{\partial\theta}\mathbf{a}_i\right) \\ &= \begin{cases} -2\left(\mathbf{W}_k\frac{\partial}{\partial\theta}\mathbf{a}_k\right)^{\mathrm{T}}\left(\mathbf{R} - \mathbf{W}_{\mathrm{s}}^{-1}\mathbf{W}_k\mathbf{R}\right) & k = i \\ -2\left(\mathbf{W}_i\frac{\partial}{\partial\theta}\mathbf{a}_i\right)^{\mathrm{T}}\left(-\mathbf{W}_{\mathrm{s}}^{-1}\mathbf{W}_k\mathbf{R}\right) & k \neq i \end{cases} .\end{aligned} \tag{3.52}$$

The sum over $i = 1, \ldots, N$ reads:

$$\begin{aligned}\frac{\partial}{\partial\theta}\nabla_{\mathbf{p}_k}J &= -2\left[\mathbf{R}^{\mathrm{T}}\mathbf{W}_k\left(\frac{\partial}{\partial\theta}\mathbf{a}_k - \mathbf{W}_{\mathrm{s}}^{-1}\sum_{i=1}^{N}\mathbf{W}_i\frac{\partial}{\partial\theta}\mathbf{a}_i\right)\right]^{\mathrm{T}} \\ &= -2\mathbf{c}_k\left(\left\{\frac{\partial}{\partial\theta}\mathbf{a}_i\right\}_{i=1:N}\right)^{\mathrm{T}}\mathbf{W}_k\mathbf{R} .\end{aligned} \tag{3.53}$$

In a similar way, the derivative with respect to $\mathbf{q}_k$ is obtained. Note that the partial derivative of $\mathbf{a}_i$ from (3.49) contains a further centering operation. Utilising the properties of scale invariance (3.38) and concatenation (3.40), simplifies the results:

$$\frac{\partial}{\partial\theta}\nabla_{\mathbf{p}_k} J = 2\mathbf{c}_k^{\mathrm{T}}\left(\left\{\mathbf{R}'\mathbf{p}_j\right\}_{j=1:N}\right)\mathbf{W}_k\mathbf{R}\,, \qquad \frac{\partial}{\partial\theta}\nabla_{\mathbf{q}_k} J = -2\mathbf{c}_k^{\mathrm{T}}\left(\left\{\mathbf{R}'\mathbf{p}_j\right\}_{j=1:N}\right)\mathbf{W}_k\,. \tag{3.54}$$

Finally, (3.51) and (3.54) can be inserted into (3.47). This yields the derivatives of $\hat{\theta}$ with respect to the measurements $\mathbf{p}_k$ and $\mathbf{q}_k$. According to (3.45), the variance of the estimate is then obtained.

Compared to the differentiation of $\hat{\theta}$, it is straight-forward to find the derivatives of $\hat{\mathbf{t}}$ with respect to θ, $\mathbf{p}_k$ and $\mathbf{q}_k$ from the explicit definition (3.34):

$$\frac{\partial}{\partial\theta}\hat{\mathbf{t}} = -\mathbf{W}_{\mathrm{s}}^{-1}\sum_{i=1}^{N}\mathbf{W}_i\mathbf{R}'\mathbf{p}_i\,, \quad \nabla_{\mathbf{p}_k}\hat{\mathbf{t}} = -\mathbf{W}_{\mathrm{s}}^{-1}\mathbf{W}_k\mathbf{R}\,, \quad \nabla_{\mathbf{q}_k}\hat{\mathbf{t}} = \mathbf{W}_{\mathrm{s}}^{-1}\mathbf{W}_k\,. \tag{3.55}$$

Inserting these into (3.46) yields a model of the variance of $\hat{\mathbf{t}}$. It is remarkable that, due to the first term in (3.55), this expression depends on the absolute feature positions $\mathbf{p}_i$. All other derivatives are either constant or depend on the feature locations relative to each other as produced by the centering operator.

3.2.4.3 Scalar-weighted Procrustes problem

The scalar-weighted Procrustes problem from Sec. 3.2.3.3 has a solution (3.44) that is an explicit function of the scatter matrix $\mathbf{\Sigma}_{\mathcal{PQ}}$. In order to derive the estimate's statistical properties, the entries of the scatter matrix are studied in a first step. Secondly, a second-order series expansion of the estimate is developed. This allows not only to calculate the variance but also the bias. Based on these results, a novel bias-corrected algorithm will be proposed.

Firstly, the distribution of $\mathbf{f} \coloneqq \begin{bmatrix} f_1 & f_2 \end{bmatrix}^{\mathrm{T}}$ from (3.43) is considered. This variable is modelled with a Gaussian $\mathbf{f} \sim \mathcal{N}(\boldsymbol{\mu}_{\mathbf{f}}, \mathbf{\Sigma}_{\mathbf{f}})$. Due to the assumption of zero mean noise, the expectation $\boldsymbol{\mu}_{\mathbf{f}}$ can be calculated from the expectations of the feature positions $\mathcal{P}_\mu \coloneqq \{\,\boldsymbol{\mu}_{\mathbf{p}_i}\,\}_{i=1:N}$ and $\mathcal{Q}_\mu \coloneqq \{\,\boldsymbol{\mu}_{\mathbf{q}_i}\,\}_{i=1:N}$.

Since the scalar-weighted Procrustes problem only yields an optimal solution for isotropic noise, it is assumed in the following that $\mathbf{\Sigma}_{\mathbf{p}_i} = \sigma_{p_i}^2\mathbf{I}_{2\times2}$ and $\mathbf{\Sigma}_{\mathbf{q}_i} = \sigma_{q_i}^2\mathbf{I}_{2\times2}$. In order to obtain the covariance $\mathbf{\Sigma}_{\mathbf{f}}$, $\mathbf{f}$ is expressed in terms of $\mathbf{\Sigma}_{\mathcal{PQ}}$:

$$f_1 = \Sigma_{\mathcal{PQ},11} + \Sigma_{\mathcal{PQ},22} = \sum_{i=1}^{N} w_i\mathbf{c}_i^{\mathrm{T}}\left(\mathcal{Q}\right)\mathbf{c}_i\left(\mathcal{P}\right) \tag{3.56a}$$

$$f_2 = \Sigma_{\mathcal{PQ},12} - \Sigma_{\mathcal{PQ},21} = \sum_{i=1}^{N} w_i\mathbf{c}_i^{\mathrm{T}}\left(\mathcal{Q}\right)\begin{bmatrix} 0 & 1 \\ -1 & 0 \end{bmatrix}\mathbf{c}_i\left(\mathcal{P}\right)\,. \tag{3.56b}$$

Using the property of additivity (3.37), the effect of additive stochastic errors on the elements of $\mathcal{P}$ and $\mathcal{Q}$ can be propagated. The assumption of mutual independence of the noise thereby allows evaluating the sum in (3.56a) element-wise. Note that it is possible to write $\mathbf{c}_i(\mathcal{Q}_\mu) = \mathbf{R}\mathbf{c}_i(\mathcal{P}_\mu)$ using (3.39)-(3.38):

$$\begin{aligned}\operatorname{var}(f_1) &= \sum_{i=1}^{N} w_i^2 \mathbf{c}_i^{\mathrm{T}}(\mathcal{Q}_\mu)\,\mathbf{c}_i(\mathcal{Q}_\mu)\,\sigma_{p_i}^2 + w_i^2 \mathbf{c}_i^{\mathrm{T}}(\mathcal{P}_\mu)\,\mathbf{c}_i(\mathcal{P}_\mu)\,\sigma_{q_i}^2 + 2w_i^2\sigma_{p_i}^2\sigma_{q_i}^2 \\ &= \sum_{i=1}^{N} w_i^2 \|\mathbf{c}_i(\mathcal{P}_\mu)\|^2 \left(\sigma_{p_i}^2 + \sigma_{q_i}^2\right) + 2w_i^2\sigma_{p_i}^2\sigma_{q_i}^2 \,. \end{aligned} \tag{3.57}$$

In a similar manner, the variance $\operatorname{var}(f_2) = \operatorname{var}(f_1) =: \sigma_f^2$ is obtained from (3.56b). Since errors in both sets are assumed to be independent, the cross-covariance $\operatorname{cov}(f_1, f_2)$ vanishes. Hence, the covariance matrix of $\mathbf{f}$ reads $\mathbf{\Sigma}_\mathbf{f} = \sigma_f^2 \mathbf{I}_{2\times 2}$.

Secondly, the perturbations in $\mathbf{f}$ are propagated to the estimates of $\hat{c}$, $\hat{s}$ and $\hat{\theta}$. To this end, the following gradient and Hessians are calculated from (3.44):

$$\nabla_\mathbf{f}\hat{\theta} = \begin{bmatrix} \frac{f_2}{f_1^2+f_2^2} & -\frac{f_1}{f_1^2+f_2^2} \end{bmatrix} \tag{3.58a}$$

$$\Delta_\mathbf{f}^\mathbf{f}\hat{c} = \frac{1}{\left(f_1^2+f_2^2\right)^{\frac{5}{2}}} \begin{bmatrix} -3f_1f_2^2 & 2f_1^2f_2 - f_2^3 \\ 2f_1^2f_2 - f_2^3 & 2f_1f_2^2 - f_1^3 \end{bmatrix} \tag{3.58b}$$

$$\Delta_\mathbf{f}^\mathbf{f}\hat{s} = \frac{1}{\left(f_1^2+f_2^2\right)^{\frac{5}{2}}} \begin{bmatrix} 2f_1^2f_2 - f_2^3 & 2f_1f_2^2 - f_1^3 \\ 2f_1f_2^2 - f_1^3 & -3f_1^2f_2 \end{bmatrix} \,. \tag{3.58c}$$

As before, a linearisation yields a first-order approximation of the variance σ_θ^2:

$$\sigma_\theta^2 = \left(\nabla_\mathbf{f}\hat{\theta}\right)\mathbf{\Sigma}_\mathbf{f}\left(\nabla_\mathbf{f}\hat{\theta}\right)^{\mathrm{T}} = \frac{\sum_{i=1}^{N} w_i^2 \|\mathbf{c}_i(\mathcal{P}_\mu)\|^2 \left(\sigma_{p_i}^2 + \sigma_{q_i}^2\right) + 2w_i^2\sigma_{p_i}^2\sigma_{q_i}^2}{\sum_{i=1}^{N} w_i^2 \|\mathbf{c}_i(\mathcal{P}_\mu)\|^2} \,. \tag{3.59}$$

The last equality follows from explicitly calculating $f_1^2 + f_2^2$ from (3.56). Therefore, the variance of $\hat{\theta}$ depends on the spacing of the features as described by $\sum_{i=1}^{N} w_i^2 \|\mathbf{c}_i(\mathcal{P}_\mu)\|^2$ but is invariant against the true rotation angle θ.

With the Hessians (3.58b)-(3.58c) one can furthermore calculate the bias in $\hat{c}$ and $\hat{s}$:

$$\mathbb{E}[\hat{c} - c] = \frac{1}{2}\operatorname{tr}\left(\left(\Delta_\mathbf{f}^\mathbf{f}\hat{c}\right)\mathbf{\Sigma}_\mathbf{f}\right) = -\frac{1}{2}\frac{\sigma_f^2}{\mu_{f_1}^2 + \mu_{f_2}^2}\cos(\theta) \tag{3.60a}$$

$$\mathbb{E}[\hat{s} - s] = \frac{1}{2}\operatorname{tr}\left(\left(\Delta_\mathbf{f}^\mathbf{f}\hat{s}\right)\mathbf{\Sigma}_\mathbf{f}\right) = -\frac{1}{2}\frac{\sigma_f^2}{\mu_{f_1}^2 + \mu_{f_2}^2}\sin(\theta) \,. \tag{3.60b}$$

These expressions show that even in the case of zero mean measurement errors, the estimates are biased proportionally to the true values. The relative estimation bias is $\lambda_f := -\frac{1}{2}\sigma_f^2\left(\mu_{f_1}^2 + \mu_{f_2}^2\right)^{-1}$. It has to be noted that this bias is not caused by an incorrect

least squares approach but is an inherent property of maximum likelihood estimators for non-linear parameter models [26].[13]

It is straight-forward to define alternative estimator equations that enhance (3.44) with a multiplicative de-biasing:

$$\hat{c}_{\text{new}} = \frac{f_1}{\sqrt{f_1^2 + f_2^2}} \frac{1}{1 - \lambda_f} \,, \qquad \hat{s}_{\text{new}} = \frac{f_2}{\sqrt{f_1^2 + f_2^2}} \frac{1}{1 - \lambda_f} \,. \tag{3.61}$$

However, the bias correction has been analytically derived using the expected values μ_{f_1} and μ_{f_2}. In practice, these are not known and only the corresponding noisy quantities can be used. The effect of this bias correction will be explored in Sec. 3.2.5.

Lastly, the variance of the estimated translation $\hat{\mathbf{t}}$ is given by (3.46). The required derivatives are a special case of the ones for the matrix-weighted Procrustes problem from (3.55). One obtains with the abbreviation $\bar{\boldsymbol{\mu}}_{\mathbf{q}} := w_{\text{s}}^{-1} \sum_{i=1}^{N} w_i \boldsymbol{\mu}_{\mathbf{q}_i}$ for the centroid of $\mathcal{Q}_\mu$:

$$\boldsymbol{\Sigma}_{\mathbf{t}} = \begin{bmatrix} 0 & -1 \\ 1 & 0 \end{bmatrix} \left(\bar{\boldsymbol{\mu}}_{\mathbf{q}} - \mathbf{t}\right) \left(\bar{\boldsymbol{\mu}}_{\mathbf{q}} - \mathbf{t}\right)^{\mathrm{T}} \begin{bmatrix} 0 & 1 \\ -1 & 0 \end{bmatrix} \sigma_\theta^2 + \frac{1}{w_{\text{s}}^2} \sum_{i=1}^{N} w_i^2 \mathbf{I}_{2\times 2} \left(\sigma_{p_i}^2 + \sigma_{q_i}^2\right) \,. \tag{3.62}$$

In summary, the variance of the estimated rotation and translation parameters is given by (3.59) and (3.62). The estimated rotation matrix $\mathbf{R}$ carries a multiplicative bias (3.60) due to the non-linearity of the parameter model. This is remedied by the proposed de-biased algorithm (3.61).

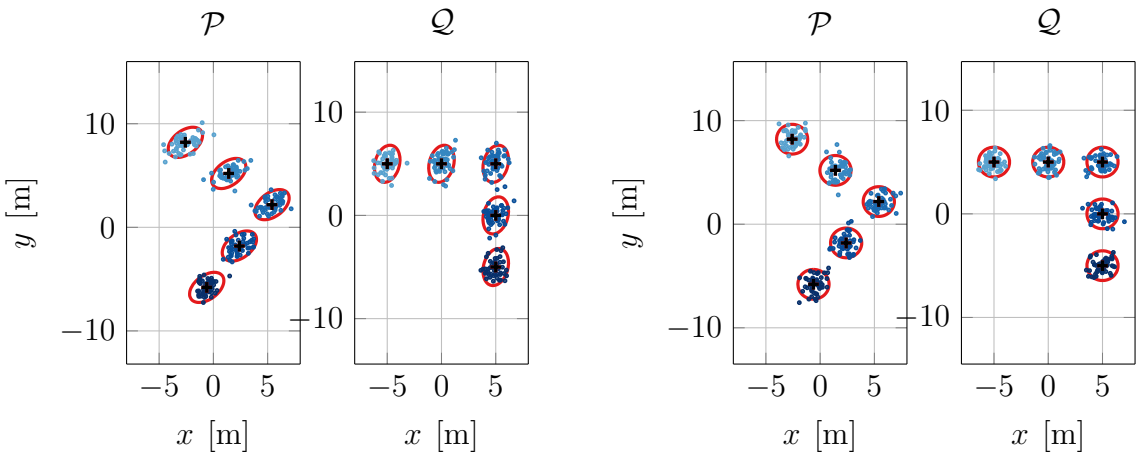
3.2.5 Numerical example

To verify the analytical models, these are compared to the results from Monte-Carlo simulations. A scenario with $N = 5$ features that are spread as seen in Fig. 3.13 is chosen. While the translation remains constant with $\mathbf{t} = \begin{bmatrix} 2\,\text{m} & 0 \end{bmatrix}$, the rotation angle varies with $\theta = 0 \ldots 180°$.

The measured feature positions from the set $\mathcal{P}$ before and after the transformation $\mathcal{Q}$ are corrupted by additive Gaussian noise. Simulations are carried out individually for anisotropic and isotropic noise. Thereby, equal covariances over all features are selected as

$$\begin{aligned} \boldsymbol{\Sigma}_{\mathbf{p},\text{anisotropic}} &= \boldsymbol{\Sigma}_{\mathbf{q},\text{anisotropic}} = \begin{bmatrix} 0.3\,\text{m}^2 & 0.1\,\text{m}^2 \\ 0.1\,\text{m}^2 & 0.7\,\text{m}^2 \end{bmatrix} \,, \\ \boldsymbol{\Sigma}_{\mathbf{p},\text{isotropic}} &= \boldsymbol{\Sigma}_{\mathbf{q},\text{isotropic}} = \sqrt{0.2}\text{m}^2 \mathbf{I}_{2\times 2} \,. \end{aligned} \tag{3.63}$$

[13] The non-linearity is slightly obfuscated if the rotation matrix is written in terms of c and s. However, in order to incorporate the necessary constraint, the Lagrangian (3.43) is again a non-linear model in c and s. The unconstrained problem (3.42) defined by θ on the other hand is clearly non-linear due to the trigonometric expressions in $\mathbf{R}$.



(a) Anisotropic noise. **(b)** Isotropic noise.

Figure 3.13 Illustration of the locations of $N = 5$ features in a simulated scenario with $\theta = 37°$. Exemplary realisations of the measurements with (a) anisotropic and (b) isotropic Gaussian noise are shown together with the 90% confidence ellipses.

In both cases, the covariance ellipses span an equal area of $\sqrt{\det(\mathbf{\Sigma})} = \sqrt{0.2}\mathrm{m}^2$. It is thus interesting to compare the accuracy of the respective estimates from the matrix-weighted and scalar-weighted Procrustes problem. If a difference occurs, this is due to the directional characteristic of the noise.

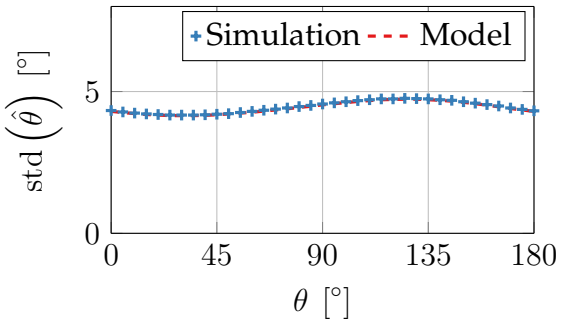
For the scenarios with anisotropic noise, solving the matrix-weighted Procrustes problem by minimisation of the objective function (3.35) yields an optimal estimate in the least-squares sense. A numerical non-linear optimisation algorithm is used to this end. In the case of isotropic noise, the closed-form solution (3.44) and the proposed de-biased version (3.61) are evaluated. For every combination of rotation angle and noise characteristic, $N_{\mathrm{sim}} = 1 \times 10^6$ independent realisations of the measurement noise are simulated.

The standard deviation of the calculated estimates is visualised in Fig. 3.14 and compared to the analytical models. A number of noteworthy aspects can be observed:

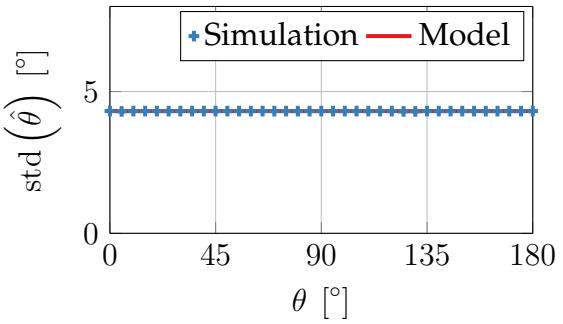
- While in the case of anisotropic noise (Fig. 3.14(a)) a clear dependence of the variance of $\hat{\theta}$ on the rotation angle θ can be observed, constant values result under isotropic noise (Fig. 3.14(b)) as indicated by (3.59). The invariance of $\mathbf{\Sigma}_{\mathbf{t}}$ against θ in the scalar-weighted case as observed in (3.62) can be noted in Fig. 3.14(d).

- The errors in the estimated translation $\hat{\mathbf{t}}$ in Fig. 3.14(c)-(d) are unequal in the two directional components, even under isotropic noise. This relates to the dependence on the absolute feature positions as seen in (3.55) and (3.62).

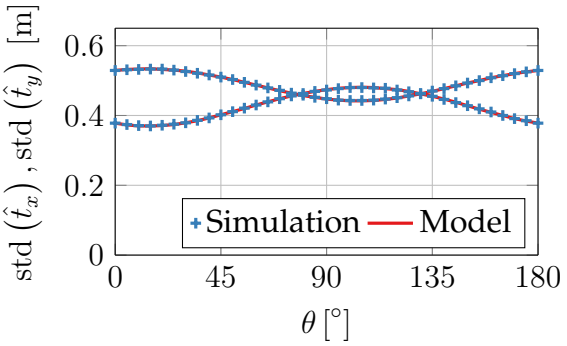
- Despite the variation over θ that is observed for the matrix-weighted Procrustes problem, the standard deviations are in the same order of magnitude for both noise characteristics. Further investigations of the analytical models could reveal



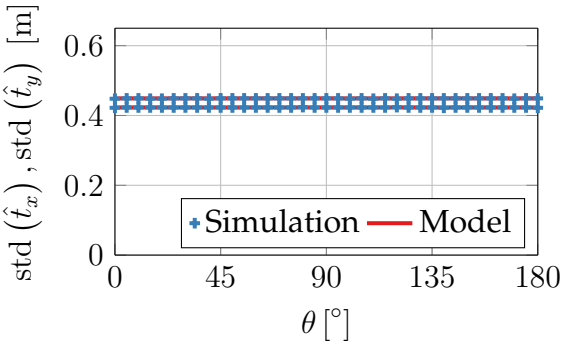
(a) Standard deviation of $\hat{\theta}$ in the matrix-weighted Procrustes problem.



(b) Standard deviation of $\hat{\theta}$ in the scalar-weighted Procrustes problem.



(c) Standard deviation of the components of $\hat{\mathbf{t}}$ for the matrix-weighted Procrustes problem.



(d) Standard deviation of the components of $\hat{\mathbf{t}}$ for the scalar-weighted Procrustes problem.

Figure 3.14 Monte-Carlo simulation results of the matrix-weighted Procrustes problem (left) according to Sec. 3.2.3.2 and the scalar-weighted case (right) from Sec. 3.2.3.3 in the anisotropic and isotropic noise scenarios as seen in Fig. 3.13. The models from Sec. 3.2.4.2 and Sec. 3.2.4.3 are shown for comparison and a good prediction of the respective standard deviation can be observed.

conditions on the noise processes and feature spacing, under which these take on equal average values.

Furthermore, Fig. 3.15 depicts the bias in the estimated entries of the rotation matrix for the scalar-weighted Procrustes problem. The simulated results agree with the analytical expectations from (3.60). Moreover, the proposed de-biased estimator (3.61) is capable of significantly reducing the bias.

3.2.6 Summary

This section has addressed uncertainty in low-level algorithms for feature-based localisation. In the intelligent vehicles domain, localisation is usually performed under a planar world assumption and requires solving the two-dimensional Procrustes problem. Depending on the measurement noise characteristics, either a matrix- or scalar-weighted problem formulation arises.

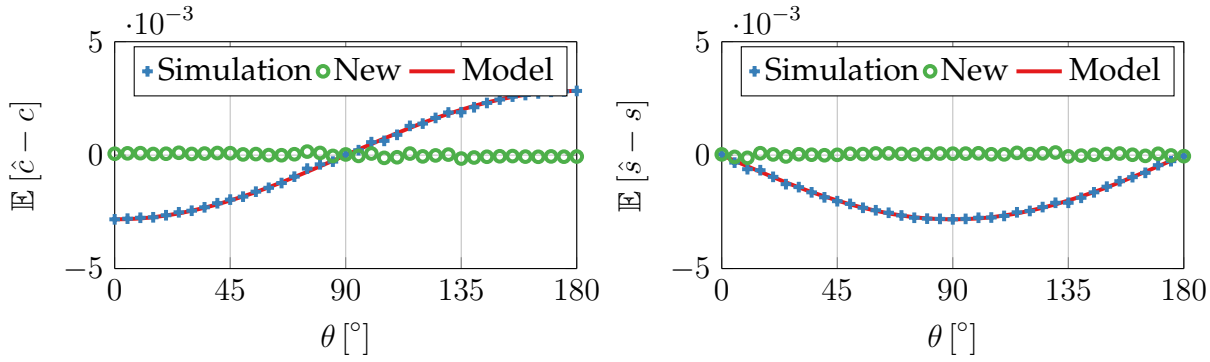


Figure 3.15 Monte-Carlo simulation results of the bias in the scalar-weighted Procrustes problem. The observed bias is proportional to the absolute values of $\cos(\theta)$ and $\sin(\theta)$, respectively, as it indicated in the analytical model (3.60). With the novel estimator (3.61), almost bias-free results are obtained.

For both cases, analytical models of the estimation uncertainty have been derived. Previous works are extended in terms of comprehensiveness and conciseness of the model derivations. Additionally, a de-biased solution of the 2-D scalar-weighted Procrustes problem has been proposed. This yields improved estimates which are free of an otherwise multiplicative bias. The closed-form expressions are useful to optimally parametrise a Bayes filter for tracking the position over time. Foundations of these filters and statistical models thereof will be discussed in the subsequent Sec. 4.1.

Moreover, analytical models can be employed to efficiently derive sensor performance characteristics which are required to reach a desired localisation accuracy. This topic will be studied in greater detail for a stereo vision based system in Sec. 7.1.

4 Modelling of algorithms for state estimation and prediction

4.1 Object tracking and state estimation

In contrast to the previously detailed perception layer, the origin of observations over time is now explicitly considered. Given a model of the time evolution of an underlying dynamic process, one can associate individual measurements (tracking) and reduce the estimation errors (filtering).

Statistical models of the filtered estimates are presented in this section. The overall approach is firstly motivated in Sec. 4.1.1, followed by a survey of related publications in Sec. 4.1.2. A formal definition of the tracking and state estimation problem is then given in Sec. 4.1.3 and a number of popular algorithms are discussed. Sec. 4.1.4 presents analytical lower bounds on the state estimation uncertainty. These are illustrated with numerical examples in Sec. 4.1.5. Finally, a summary with potential future research directions concludes this section in Sec. 4.1.6.

4.1.1 Introduction

Sensor measurements, as introduced in the previous Sec. 3.1-3.2, represent observations at specific points in time that originate from a in general dynamic process. In the context of advanced driver assistance systems, these dynamics refer for example to the motion of traffic participants. The goal of a tracking algorithm is to associate measurements from different time-steps. To this end, a model of the underlying dynamics is assumed. Subsequently, the combined information can be used to estimate unobservable state variables in the dynamic model. This is known as filtering or state estimation.

However, in general, neither the measurements nor the dynamic system model are exact but affected by stochastic uncertainties. Consequently, the state estimates are uncertain as well. The goal of this section is to find analytic models of this uncertainty in order to be able to propagate them to subsequent algorithms.

An important aspect will be to understand the impact of the state history. Given that the measurements are realisations of a random variable at each time step, modelling the tracking filter results for all possible combinations from multiple time steps becomes challenging. The approach pursued here is to study the Cramér-Rao lower bound (CRB) on the best attainable state estimation variance. This bound is independent of the assumptions and parametrisations of sub-optimal filters which allows drawing generic conclusions.

In the context of this thesis, the purpose of deriving analytical bounds on the state estimation variance is twofold. The performance bounds can be used for the selection of sensor requirements and algorithms for ADAS. Sub-optimal filters can be compared to the theoretical bound and the room for improvement is revealed. Furthermore, it is possible to analytically propagate the uncertainty from the perception layer to subsequent algorithms. This will be applied for the example of an AEB system in Sec. 7.2.

Since a large body of works on performance bounds in general tracking problems exists, it is out of the scope of this chapter to further develop this theory. However, as

will be seen from the survey in the next section, only a few examples of applications in the driver assistance domain have been reported yet.

4.1.2 Related work

Before the advent of environment perception technologies in the automotive domain, tracking the motion of objects has been considered for example in maritime applications or aviation [10, 123, 124]. Similar theoretical foundations are applied to estimate a dynamic motion state with automotive sensors, for example radar [104], laser scanner [103], stereo vision [12] or the fused information from multiple sensors [101, 148, 153].

The performance of a tracking algorithm can be evaluated with artificial data in a simulation environment [80]. The estimation errors can be evaluated since the ground truth trajectories are known. One shortcoming of this numerical approach is the effort in sampling from different trajectories and measurement noise realisations [65]. Another issue is that in order to objectively evaluate the performance of a tracking filter, one has to define a meaningful reference for comparison, for example an optimal filtering solution. Often though, such an optimal filter cannot be realised with finite computational power.

Therefore, it has become an active research topic to analytically study the theoretical limit on the state estimation errors as given by the Cramér-Rao lower bound. This allows gaining further insight into the problem. For example, the filter performance becomes independent of the actual state trajectory under certain conditions and thus reduces to a function of time only.

The theoretical foundations of the CRB have been developed in [16, 74, 188, 190]. Applications of this theory are for example the derivation of performance bounds in angle-only tracking [88] or the deployment of multiple sensors [78]. In an automotive context, examples for the use of the CRB have been reported in cooperative localisation [163] and as a reference for performance evaluation [8, 24]. Since these examples are limited in number and depth, the goal of this work is to provide a more complete overview of the CRB and illustrate its use for the development of driver assistance systems.

4.1.3 State estimation problem formulation and solutions

Formally, the goal is to estimate the probability density $p\left(\mathbf{x}_{1:k}\middle|\mathbf{y}_{1:k}\right)$ of the state sequence $\mathbf{x}_{1:k} \coloneqq \mathbf{x}_1, \ldots, \mathbf{x}_k$ with $\mathbf{x}_k \in \mathbb{R}^n \; \forall k$ using a sequence of measurements $\mathbf{y}_{1:k} \coloneqq \mathbf{y}_1, \ldots, \mathbf{y}_k$ with $\mathbf{y}_k \in \mathbb{R}^m \; \forall k$ from the time-steps $1, \ldots, k$. To this end, the following models are assumed. Firstly, the time evolution of $\mathbf{x}_k$ is governed by a stochastic process model

$$\mathbf{x}_{k+1} \sim p\left(\mathbf{x}_{k+1}\middle|\mathbf{x}_{1:k}, \mathbf{u}_{1:k}\right) \tag{4.1a}$$

where $\mathbf{u}_{1:k} \coloneqq \mathbf{u}_1, \ldots, \mathbf{u}_k$ is a sequence of known, deterministic input signals.

Secondly, each measurement $\mathbf{y}_k$ is related to the sequence of system states through the measurement equation:

$$\mathbf{y}_k \sim p\left(\mathbf{y}_k\middle|\mathbf{x}_{1:k}\right) \; . \tag{4.1b}$$

However, in this general notation, the estimation problem is hardly tractable. Two assumptions are the key to finding recursive or even analytic expressions for $p(\mathbf{x}_{1:k}|\mathbf{y}_{1:k})$. First, under the Markov assumption, the recursive Bayes filter will be studied in Sec. 4.1.3.1. The Bayes filter can be numerically implemented for general probability density functions in the particle filter framework, which will be introduced in Sec. 4.1.3.2. Second, closed-form filter equations can be obtained under the additional assumptions of linear system dynamics and Gaussian densities. These are known as the Kalman filter and will be derived in Sec. 4.1.3.3. A similar algorithm for non-linear systems, the extended Kalman filter, can be obtained by linearisation and will be presented in Sec. 4.1.3.4. Lastly, the state estimation problem for multiple targets will be introduced in Sec. 4.1.3.5.

4.1.3.1 Bayes filter for Markovian systems

As a first simplification, the Markov assumption is introduced in (4.1). A new state $\mathbf{x}_{k+1}$ depends only on its previous value $\mathbf{x}_k$ and the input $\mathbf{u}_k$ instead of the full state history $\mathbf{x}_{1:k}$. Similarly, all information which leads to the measurement $\mathbf{y}_k$ is included in the current state $\mathbf{x}_k$:

$$\mathbf{x}_{k+1} \sim p(\mathbf{x}_{k+1}|\mathbf{x}_k, \mathbf{u}_k) \tag{4.2a}$$
$$\mathbf{y}_k \sim p(\mathbf{y}_k|\mathbf{x}_k) \ . \tag{4.2b}$$

For this system, the desired density $p(\mathbf{x}_{k+1}|\mathbf{y}_{1:k+1})$ can be recursively calculated from $p(\mathbf{x}_k|\mathbf{y}_{1:k})$. Firstly, applying Bayes' rule (2.8) yields the a posteriori density $p(\mathbf{x}_{k+1}|\mathbf{y}_{1:k+1})$ as a function of the measurement model (4.2b) and the a priori density $p(\mathbf{x}_{k+1}|\mathbf{y}_{1:k})$ [196]:

$$p(\mathbf{x}_{k+1}|\mathbf{y}_{1:k+1}) = \frac{p(\mathbf{y}_{k+1}|\mathbf{x}_{k+1})p(\mathbf{x}_{k+1}|\mathbf{y}_{1:k})}{p(\mathbf{y}_{k+1}|\mathbf{y}_{1:k})} = A^{-1}p(\mathbf{y}_{k+1}|\mathbf{x}_{k+1})p(\mathbf{x}_{k+1}|\mathbf{y}_{1:k}) \ . \tag{4.3}$$

Note that the denominator $p(\mathbf{y}_{k+1}|\mathbf{y}_{1:k})$ is independent of $\mathbf{x}_{k+1}$ and therefore serves as a mere normalisation constant A. Secondly, employing the law of total probability (2.7), one can write the a priori density $p(\mathbf{x}_{k+1}|\mathbf{y}_{1:k})$ as follows [196]:

$$p(\mathbf{x}_{k+1}|\mathbf{y}_{1:k}) = \int p(\mathbf{x}_{k+1}|\mathbf{x}_k, \mathbf{u}_k)p(\mathbf{x}_k|\mathbf{y}_{1:k}) \ \mathrm{d}\mathbf{x}_k \ . \tag{4.4}$$

This expression only depends on the system equation (4.2a) and the a posteriori density $p(\mathbf{x}_k|\mathbf{y}_{1:k})$ from the previous time step. Therefore, these two steps termed *prediction* (4.4) and *innovation* (4.3) provide a way to recursively estimate $p(\mathbf{x}_{k+1}|\mathbf{y}_{1:k+1})$, given an initial state distribution $p(\mathbf{x}_0)$. A point estimate $\hat{\mathbf{x}}_{k+1}$ of the state can then be constructed from this density according to various criteria, for example the maximum likelihood principle.

Note that no assumption on the actual probability densities has been introduced yet. In practice, two representations are commonly used: A non-parametric distribution, described by a finite set of samples, is the foundation of a particle filter as it will be

introduced in Sec. 4.1.3.2. Parametric distributions on the other hand play an important role as they allow further simplifications of the general equations (4.3)-(4.4). In the frequent case of a Gaussian distribution $\mathbf{x}_k \sim \mathcal{N}(\hat{\mathbf{x}}_k, \mathbf{\Sigma}_k)$ described by the mean $\hat{\mathbf{x}}_k$ and covariance $\mathbf{\Sigma}_k$, the popular Kalman filter equations can be derived which will be shown in Sec. 4.1.3.3.

4.1.3.2 Particle filter

The approach of a particle filter is to represent the involved probability densities by a set of N_s samples and numerically propagate these through (4.3)-(4.4) with Monte-Carlo methods. In theory, an arbitrary density function can be represented by $N_\mathrm{s} \to \infty$ particles. In a practical implementation though, only a finite set can be used. Various strategies exist in order to maintain an accurate approximation [5, 166]. The following derivation of a naïve implementation primarily intends to illustrate the concept as presented in [196].

The posterior state distribution $p\left(\mathbf{x}_k \middle| \mathbf{y}_{1:k}\right)$ is represented by $i = 1, \ldots, N_\mathrm{s}$ samples $\mathbf{x}_k^{(i)}$ and weights $\omega_k^{(i)}$ as [5]:

$$p\left(\mathbf{x}_k \middle| \mathbf{y}_{1:k}\right) \approx \sum_{i=1}^{N_\mathrm{s}} \omega_k^{(i)} \delta\left(\mathbf{x}_k - \mathbf{x}_k^{(i)}\right) , \qquad \sum_{i=1}^{N} \omega_k^{(i)} = 1 , \qquad \omega_k^{(i)} \geq 0 \quad \forall i . \tag{4.5}$$

With this definition, the prediction step (4.4) reads:

$$p\left(\mathbf{x}_{k+1} \middle| \mathbf{y}_{1:k}\right) = \sum_{i=1}^{N_\mathrm{s}} \omega_k^{(i)} p\left(\mathbf{x}_{k+1} \middle| \mathbf{x}_k^{(i)}, \mathbf{u}_k\right) . \tag{4.6}$$

The a priori density $p\left(\mathbf{x}_{k+1} \middle| \mathbf{y}_{1:k}\right)$ is described by samples $\mathbf{x}_{k+1|k}^{(i)}$ which are drawn from $p\left(\mathbf{x}_{k+1} \middle| \mathbf{x}_k^{(i)}, \mathbf{u}_k\right)$. Inserting the a priori density in particle representation into the innovation step (4.3) yields the a posteriori density $p\left(\mathbf{x}_{k+1} \middle| \mathbf{y}_{1:k+1}\right)$ as a re-weighted set of samples [196]:

$$p\left(\mathbf{x}_{k+1} \middle| \mathbf{y}_{1:k+1}\right) = \sum_{i=1}^{N_\mathrm{s}} \omega_{k+1}^{(i)} \delta\left(\mathbf{x}_{k+1} - \mathbf{x}_{k+1|k}^{(i)}\right) \tag{4.7}$$

where $\omega_{k+1}^{(i)} = A^{-1} \omega_k^{(i)} p\left(\mathbf{y}_{k+1} \middle| \mathbf{x}_{k+1|k}^{(i)}\right)$ and $A = \sum_{i=1}^{N_\mathrm{s}} \omega_k^{(i)} p\left(\mathbf{y}_{k+1} \middle| \mathbf{x}_{k+1|k}^{(i)}\right)$. In practice, the particle representation of $p\left(\mathbf{x}_{k+1} \middle| \mathbf{y}_{1:k+1}\right)$ can degenerate over time, that is most particles eventually contribute with negligible weights. One possible countermeasure is to periodically create a new representation by drawing N_s particles with replacement from $p\left(\mathbf{x}_{k+1} \middle| \mathbf{y}_{1:k+1}\right)$ and resetting all weights to $\omega_k^{(i)} = N_\mathrm{s}^{-1}$ [5].

The advantage of the particle filter framework is that arbitrary system and measurement models can be used. However, a drawback is the computational effort since in order to accurately represent the distribution of a multidimensional state vector, it is required to propagate a sufficiently high number of particles.

4.1.3.3 Kalman filter

In this section, the special case of linear system and measurement models is considered. The model contains zero mean Gaussian process noise $\mathbf{w}_k$ and measurement noise $\mathbf{v}_k$:

$$\mathbf{x}_{k+1} = \mathbf{A}_k\mathbf{x}_k + \mathbf{B}_k\mathbf{u}_k + \mathbf{w}_k \;, \qquad \mathbf{w}_k \sim \mathcal{N}(\mathbf{0}, \mathbf{Q}_k) \tag{4.8a}$$

$$\mathbf{y}_k = \mathbf{C}_k\mathbf{x}_k + \mathbf{v}_k \;, \qquad \mathbf{v}_k \sim \mathcal{N}(\mathbf{0}, \mathbf{R}_k) \;. \tag{4.8b}$$

Unless stated otherwise, *white* and *independent* noise processes are assumed in the remainder of this work:

$$\mathbb{E}\left[\mathbf{w}_k\mathbf{w}_{k-\kappa}^{\mathrm{T}}\right] = \mathbf{0}_{n\times n} \quad \forall k \;, \forall\kappa \neq 0 \;, \qquad \mathbb{E}\left[\mathbf{v}_k\mathbf{v}_{k-\kappa}^{\mathrm{T}}\right] = \mathbf{0}_{m\times m} \quad \forall k \;, \forall\kappa \neq 0 \;, \tag{4.9a}$$

$$\mathbb{E}\left[\mathbf{w}_k\mathbf{v}_{k-\kappa}^{\mathrm{T}}\right] = \mathbf{0}_{n\times m} \quad \forall k \;, \forall\kappa \;. \tag{4.9b}$$

If the initial state is a Gaussian $\mathbf{x}_0 \sim \mathcal{N}(\mathbf{x}_0; \hat{\mathbf{x}}_0, \mathbf{\Sigma}_0)$ and independent of $\mathbf{w}_k$ and $\mathbf{v}_k$, the state density $p(\mathbf{x}_k|\mathbf{y}_{1:k})$ remains a Gaussian $\mathcal{N}(\mathbf{x}_k; \hat{\mathbf{x}}_k, \mathbf{\Sigma}_k)$ due to the linearity.[1] The conditional probabilities (4.2) are given as:

$$p(\mathbf{x}_{k+1}|\mathbf{x}_k, \mathbf{u}_k) = \mathcal{N}(\mathbf{x}_{k+1}; \mathbf{A}_k\mathbf{x}_k + \mathbf{B}_k\mathbf{u}_k, \mathbf{Q}_k) \tag{4.10a}$$

$$p(\mathbf{y}_k|\mathbf{x}_k) = \mathcal{N}(\mathbf{y}_k; \mathbf{C}_k\mathbf{x}_k, \mathbf{R}_k) \;. \tag{4.10b}$$

With these assumptions, it is possible to find closed-form expressions for the Gaussian a priori density $\mathcal{N}(\mathbf{x}_{k+1}; \hat{\mathbf{x}}_{k+1|k}, \mathbf{\Sigma}_{k+1|k})$ from (4.4) and the a posteriori density $\mathcal{N}(\mathbf{x}_{k+1}; \hat{\mathbf{x}}_{k+1}, \mathbf{\Sigma}_{k+1})$ from (4.3). These equations are known as the linear Kalman filter. In the following, a brief outline of the full derivation as found, for example in [187], is given.[2]

First, the prediction step (4.4) is worked out using (2.13):

$$\begin{aligned} p(\mathbf{x}_{k+1}|\mathbf{y}_{1:k}) &= \int \mathcal{N}(\mathbf{x}_{k+1}; \mathbf{A}_k\mathbf{x}_k + \mathbf{B}_k\mathbf{u}_k, \mathbf{Q}_k)\,\mathcal{N}(\mathbf{x}_k; \hat{\mathbf{x}}_k, \mathbf{\Sigma}_k)\;\mathrm{d}\mathbf{x}_k \\ &= \mathcal{N}\Big(\mathbf{x}_{k+1}; \underbrace{\mathbf{A}_k\hat{\mathbf{x}}_k + \mathbf{B}_k\mathbf{u}_k}_{=\hat{\mathbf{x}}_{k+1|k}}, \underbrace{\mathbf{A}_k\mathbf{\Sigma}_k\mathbf{A}_k^{\mathrm{T}} + \mathbf{Q}_k}_{=\mathbf{\Sigma}_{k+1|k}}\Big) \;. \end{aligned} \tag{4.11}$$

Next, the innovation step (4.3) is the product of Gaussian density functions which yields a Gaussian after combination of the exponents [187]:

$$\begin{aligned} p(\mathbf{x}_{k+1}|\mathbf{y}_{1:k+1}) &= A^{-1}\mathcal{N}(\mathbf{y}_{k+1}; \mathbf{C}_{k+1}\hat{\mathbf{x}}_{k+1}, \mathbf{R}_{k+1})\,\mathcal{N}\left(\mathbf{x}_{k+1}; \hat{\mathbf{x}}_{k+1|k}, \mathbf{\Sigma}_{k+1|k}\right) \\ &= \mathcal{N}\Big(\mathbf{x}_{k+1}; \underbrace{\hat{\mathbf{x}}_{k+1|k} + \mathbf{K}_{k+1}\left(\mathbf{y}_{k+1} - \mathbf{C}_{k+1}\hat{\mathbf{x}}_{k+1|k}\right)}_{=\hat{\mathbf{x}}_{k+1}}, \underbrace{\left(\mathbf{I}_{n\times n} - \mathbf{K}_{k+1}\mathbf{C}_{k+1}\right)\mathbf{\Sigma}_{k+1|k}}_{=\mathbf{\Sigma}_{k+1}}\Big) \;, \end{aligned} \tag{4.12a}$$

[1] If not explicitly stated otherwise, a covariance matrix $\mathbf{\Sigma}$ always refers to the state $\mathbf{x}$ in this section. An additional subscript as in $\mathbf{\Sigma}_\mathbf{x}$ is omitted for improved readability.

[2] Starting the derivations from the general Bayes filter equations is in line with the previous formulations. Other derivations rely for example on an orthogonality principle as in the original work [99].

where the abbreviation $\mathbf{K}_{k+1}$ for the Kalman gain matrix

$$\mathbf{K}_{k+1} = \boldsymbol{\Sigma}_{k+1|k}\mathbf{C}_{k+1}^{\mathrm{T}}\underbrace{\left(\mathbf{C}_{k+1}\boldsymbol{\Sigma}_{k+1|k}\mathbf{C}_{k+1}^{\mathrm{T}} + \mathbf{R}_{k+1}\right)}_{=:\boldsymbol{\Sigma}_{\boldsymbol{\epsilon}_{k+1}}}^{-1} \tag{4.12b}$$

is introduced. This gain matrix relates the measurement covariance $\mathbf{R}_{k+1}$ to the predicted covariance $\boldsymbol{\Sigma}_{k+1|k}$. A state estimate $\hat{\mathbf{x}}_{k+1}$ is constructed from the prediction by adding the weighted *innovation*, that is the difference between the expected and observed measurement $\boldsymbol{\epsilon}_{k+1} = \mathbf{y}_{k+1} - \mathbf{C}_{k+1}\hat{\mathbf{x}}$.

In summary, the Bayes filter for linear Gaussian systems consists of explicit matrix operations for calculating the state predictions $\left(\hat{\mathbf{x}}_{k+1|k}, \boldsymbol{\Sigma}_{k+1|k}\right)$ and estimates $(\hat{\mathbf{x}}_{k+1}, \boldsymbol{\Sigma}_{k+1})$. The estimated mean and covariance fully describe the Gaussian a posteriori density $p\left(\mathbf{x}_{k+1}\middle|\mathbf{y}_{1:k+1}\right)$. Moreover, $\hat{\mathbf{x}}_{k+1}$ features the smallest variance $\mathrm{cov}\left(\hat{\mathbf{x}}_{k+1} - \mathbf{x}_{k+1}\right)$ compared to any unbiased estimator of $\mathbf{x}_{k+1}$ and thus is an efficient estimate [108].

However, these favourable properties are closely related to the assumption of a linear system and Gaussian noise processes. In the more general case of non-linear systems, it is in general not possible to find closed-form expressions for the propagated densities. Hence, one can either resort to a numerical approach such as the particle filter from Sec. 4.1.3.2. Or, instead of propagating the true non-Gaussian densities, these can be approximated by Gaussians in each filter step. This can be achieved for example by a linearisation of the system equations, which leads to an extended Kalman filter.

4.1.3.4 Extended Kalman filter

As a more general variant of the model (4.8), a non-linear system is assumed:

$$\mathbf{x}_{k+1} = \mathbf{f}_k\left(\mathbf{x}_k, \mathbf{u}_k, \mathbf{w}_k\right) \ , \qquad \mathbf{w}_k \sim \mathcal{N}(\mathbf{0}, \mathbf{Q}_k) \tag{4.13a}$$

$$\mathbf{y}_k = \mathbf{h}_k\left(\mathbf{x}_k, \mathbf{v}_k\right) \ , \qquad \mathbf{v}_k \sim \mathcal{N}(\mathbf{0}, \mathbf{R}_k) \ . \tag{4.13b}$$

The noise processes are independent Gaussians and the initial state is $\mathbf{x}_0 \sim \mathcal{N}(\hat{\mathbf{x}}_0, \boldsymbol{\Sigma}_0)$.

In the previous section, the linear Kalman filter has been derived as the optimal state estimator for a known system model which is linear but potentially time-varying. A natural extension to a non-linear system is therefore to consider a linearised version of it at each time-step. This yields the extended Kalman filter (EKF) algorithm [108]. Firstly, instead of the prediction step (4.11), one has:

$$\hat{\mathbf{x}}_{k+1|k} = \mathbf{f}_k\left(\hat{\mathbf{x}}_k, \mathbf{u}_k, \mathbf{0}\right) \ , \tag{4.14a}$$

$$\boldsymbol{\Sigma}_{k+1|k} = \mathbf{A}_k\boldsymbol{\Sigma}_k\mathbf{A}_k^{\mathrm{T}} + \mathbf{G}_k\mathbf{Q}_k\mathbf{G}_k^{\mathrm{T}} \ , \tag{4.14b}$$

where

$$\mathbf{A}_k = \nabla_{\mathbf{x}_k}\,\mathbf{f}_k\left(\mathbf{x}_k, \mathbf{u}_k, \mathbf{w}_k\right)\big|_{\mathbf{x}_k=\hat{\mathbf{x}}_k, \mathbf{w}_k=\mathbf{0}} \ , \tag{4.14c}$$

$$\mathbf{G}_k = \nabla_{\mathbf{w}_k}\,\mathbf{f}_k\left(\mathbf{x}_k, \mathbf{u}_k, \mathbf{w}_k\right)\big|_{\mathbf{x}_k=\hat{\mathbf{x}}_k, \mathbf{w}_k=\mathbf{0}} \ . \tag{4.14d}$$

Secondly, the innovation step reads:

$$\hat{\mathbf{x}}_{k+1} = \hat{\mathbf{x}}_{k+1|k} + \mathbf{K}_{k+1}\left(\mathbf{y}_{k+1} - \mathbf{h}_{k+1}\left(\hat{\mathbf{x}}_{k+1|k}, \mathbf{0}\right)\right) \ , \tag{4.15a}$$

$$\boldsymbol{\Sigma}_{k+1} = \left(\mathbf{I}_{n\times n} - \mathbf{K}_{k+1}\mathbf{C}_{k+1}\right)\boldsymbol{\Sigma}_{k+1|k} \ , \tag{4.15b}$$

where

$$\mathbf{K}_{k+1} = \boldsymbol{\Sigma}_{k+1|k}\mathbf{C}_{k+1}^{\mathrm{T}}\left(\mathbf{C}_{k+1}\boldsymbol{\Sigma}_{k+1|k}\mathbf{C}_{k+1}^{\mathrm{T}} + \mathbf{L}_{k+1}\mathbf{R}_{k+1}\mathbf{L}_{k+1}^{\mathrm{T}}\right)^{-1} \ , \tag{4.15c}$$

$$\mathbf{C}_{k+1} = \nabla_{\mathbf{x}_{k+1}}\, \mathbf{h}_k\left(\mathbf{x}_{k+1}, \mathbf{v}_{k+1}\right)\big|_{\mathbf{x}_{k+1}=\hat{\mathbf{x}}_{k+1|k}, \mathbf{v}_{k+1}=\mathbf{0}} \ , \tag{4.15d}$$

$$\mathbf{L}_{k+1} = \nabla_{\mathbf{v}_{k+1}}\, \mathbf{h}_k\left(\mathbf{x}_{k+1}, \mathbf{v}_{k+1}\right)\big|_{\mathbf{x}_{k+1}=\hat{\mathbf{x}}_{k+1|k}, \mathbf{v}_{k+1}=\mathbf{0}} \ . \tag{4.15e}$$

Due to the use of a linearised model, the sub-optimality of the filter is obvious. It will be studied in Sec. 4.1.4 how the performance deteriorates compared to the theoretical optimum. Note that using a first-order linearisation of the system is only one possible way to obtain a closed form of the Bayes filter (4.3)-(4.4). More sophisticated approaches [166] can lead to better approximations.

4.1.3.5 Extension to multi-target tracking

So far, only the tracking of a single object has been considered. The object's motion is represented as a dynamic system and a state estimate is obtained from sensor measurements which originate from this system. In practical automotive applications, multiple objects are visible for the sensor at the same time and it is of interest to simultaneously estimate their motion states.

However, since the measurements are usually not uniquely identifiable, estimating the states by separate filters in parallel is not straight-forward. This is especially true in an automotive context, where the distances between vehicles are typically in the same order as their dimensions and thus measurements are not cleanly separated. Therefore, the joint problem of estimating the association between multiple measurements over time and the dynamic states of multiple objects needs to be solved, which is known as multi-target tracking. Furthermore, one has to cope with the notorious case of missing or false positive measurements.

As with the single target case introduced at the beginning of Sec. 4.1.3, performing a batch estimation over all time steps is hardly possible due to the growing computational demand. This is especially relevant for the association problem with its combinatorial explosion. Therefore, additional assumptions are made in order to obtain a sub-optimal recursive but feasible solution to a multi-target tracking problem.

The main principle of a recursive association scheme is to sequentially decide on how the current measurements are integrated in the estimation without altering the use of previous measurement information [23]. Moreover, if the different targets are assumed as moving independently from each other, the motion estimation can be performed with individual state estimation filters.

In a first step, an association between measurements and the previously tracked objects has to be found. This can be a hard decision, where each measurement is exclusively assigned, or a weighted assignment as used in probabilistic data association [23].

Before the assignment between all measurements and all objects is considered though, a gating step is usually performed in order to exclude unlikely combinations. For each

currently tracked object, the filter prediction is compared with the current measurements in terms of a distance function. In the case of a Gaussian state representation, the Kalman filter predictions (4.11) can be used to determine the residual $\boldsymbol{\epsilon}_{k+1} = \mathbf{y}_{k+1} - \mathbf{C}_{k+1}\hat{\mathbf{x}}_{k+1}$ which is expected to be a Gaussian $\mathcal{N}\left(\mathbf{0}, \boldsymbol{\Sigma}_{\boldsymbol{\epsilon}_{k+1}}\right)$. A suitable scalar measure for the fit between measurement and prediction is then given by the Mahalanobis distance $\boldsymbol{\epsilon}_{k+1}^{\mathrm{T}} \boldsymbol{\Sigma}_{\boldsymbol{\epsilon}_{k+1}}^{-1} \boldsymbol{\epsilon}_{k+1} \sim \chi_m^2$. It is therefore possible to define an ellipsoidal gate in the measurement space by means of a hypothesis test with an $1-\alpha$ acceptance probability. Measurements that origin from other objects with a different mean or outlier detections that do not follow a Gaussian distribution can be rejected if the gate is sufficiently tight.

Depending on the situation, the gating step alone will not eliminate all ambiguities of the assignment problem in multi-target tracking. Moreover, the list of currently tracked objects needs to be maintained, that is new filters have to be initiated and obsolete ones terminated. Further details are given for example in [10, 23]. Still, the introductory overview that is given here suffices to discuss the extension of optimal performance bounds from single to multi-target tracking in Sec. 4.1.4.5.

4.1.4 Cramér-Rao bound for state estimation problems

Having introduced the formulation and approaches to solve the state estimation problem, a lower bound on the estimation error covariance is now derived. First, non-linear, possibly time-variant state space models of the following form are considered:

$$\mathbf{x}_{k+1} = \mathbf{f}_k\left(\mathbf{x}_k, \mathbf{u}_k, \mathbf{w}_k\right) \,, \qquad \mathbf{w}_k \sim p_k\left(\mathbf{w}_k\right) \,, \qquad \mathbb{E}\left[\mathbf{w}_k\right] = \mathbf{0} \tag{4.16a}$$

$$\mathbf{y}_k = \mathbf{h}_k\left(\mathbf{x}_k, \mathbf{v}_k\right) \,, \qquad \mathbf{v}_k \sim p_k\left(\mathbf{v}_k\right) \,, \qquad \mathbb{E}\left[\mathbf{v}_k\right] = \mathbf{0} \,. \tag{4.16b}$$

Here, the discrete time noise processes $\mathbf{w}_k$ are $\mathbf{v}_k$ assumed as independent and white with arbitrary probability density functions and zero mean.

It is assumed that an unbiased estimator $\hat{\mathbf{x}}_k\left(\mathbf{y}_{1:k}\right)$ of $\mathbf{x}_k$ exists:

$$\mathbb{E}\left[\hat{\mathbf{x}}_k\left(\mathbf{y}_{1:k}\right) - \mathbf{x}_k\right] = \mathbf{0} \,. \tag{4.17}$$

The goal is then to find a tight lower bound on the achievable estimation error covariance $\boldsymbol{\mathcal{I}}_k^{-1}$ so that

$$\mathrm{cov}\left(\hat{\mathbf{x}}_k\left(\mathbf{y}_{1:k}\right) - \mathbf{x}_k\right) \succeq \boldsymbol{\mathcal{I}}_k^{-1} \tag{4.18}$$

holds for any unbiased estimator. Depending on whether the state $\mathbf{x}_k$ is regarded as a non-random parameter or, in a Bayesian framework, as random itself, the expectations in (4.17)-(4.18) are taken using the densities $p\left(\mathbf{y}_{1:k}\middle|\mathbf{x}_k\right)$ or $p\left(\mathbf{x}_k, \mathbf{y}_{1:k}\right)$, respectively.

Recall from Sec. 2.1.3 that for the classical problem of estimating a constant, non-random parameter $\boldsymbol{\theta}$ from measurement information $\mathbf{y} \sim p\left(\mathbf{y}\middle|\boldsymbol{\theta}\right)$, such a bound is provided by the inverse of the Fisher information matrix $\boldsymbol{\mathcal{I}}_{\mathbf{y}}\left(\boldsymbol{\theta}\right)$ from (2.15). This result is known as the Cramér-Rao bound (CRB).

In contrast to the estimation of a static parameter $\boldsymbol{\theta}$, the dynamic state $\mathbf{x}_k$ is time-dependent and so will be the CRB $\boldsymbol{\mathcal{I}}_k^{-1}$ for this problem. Moreover, in the Bayesian state

estimation framework, the sequence of states $\mathbf{x}_{1:k}$ is not deterministic but a realisation of a stochastic process. Two different approaches are known for handling these challenges.

1. Similar to the classical parameter estimation theory, a *parametric* bound can be obtained if the state sequence $\mathbf{x}_{1:k}$ is treated as a known, deterministic parameter. Thus, a bound is derived for specific realisations of the state. A recursive formulation of the parametric CRB will be shown in Sec. 4.1.4.1.

2. The *posterior* CRB is derived from a Bayesian point of view and takes the probabilistic nature of the state evolution into account. Effectively, the information matrix is not calculated for a particular state trajectory but averaged over the state distribution. This bound will be explained in Sec. 4.1.4.2.

Further explicit results and relevant properties for special cases, for example the linear (4.8) or non-linear systems (4.13), will be highlighted in Sec. 4.1.4.3-4.1.4.4. Rigorous derivations and a comparison of both formulations of the CRB are given in [16]. The main results are summarised in the following.

Lastly, Sec. 4.1.4.5 discusses the challenges when the CRB for estimating the state of a single target shall be extended to the multi-target tracking problem from Sec. 4.1.3.5.

4.1.4.1 Parametric CRB

To derive the parametric bound, the sequence of system states $\mathbf{x}_{1:k}$ is written as a stacked vector and treated as a parameter $\mathbf{X}_{1:k} = \begin{bmatrix}\mathbf{x}_1^{\mathrm{T}} & \dots & \mathbf{x}_k^{\mathrm{T}}\end{bmatrix}^{\mathrm{T}}$ with a true value $\mathbf{X}_{1:k}^*$. The corresponding realisation of the process noise $\mathbf{W}_k^*$ is similarly assumed as deterministic. Likewise, the sequence of measurements $\mathbf{y}_{1:k}$ is written as $\mathbf{Y}_{1:k} = \begin{bmatrix}\mathbf{y}_1^{\mathrm{T}} & \dots & \mathbf{y}_k^{\mathrm{T}}\end{bmatrix}^{\mathrm{T}}$. The density $p\left(\mathbf{Y}_{1:k}\middle|\mathbf{X}_{1:k}\right)$ of the measurements conditional on $\mathbf{X}_{1:k}$ can be retrieved from (4.16b). Then, the Fisher information matrix is obtained from (2.15) as

$$\boldsymbol{\mathcal{I}}_{\mathbf{Y}_{1:k}}\left(\mathbf{X}_{1:k}\right) = -\mathbb{E}_{\mathbf{Y}_{1:k}|\mathbf{X}_{1:k}}\left[\Delta_{\mathbf{X}_{1:k}}^{\mathbf{X}_{1:k}} \log p\left(\mathbf{Y}_{1:k}\middle|\mathbf{X}_{1:k}\right)\right] . \tag{4.19}$$

Due to the time-dependence of $\mathbf{x}_k$, the dimensions of $\mathbf{X}_{1:k}$ grow unlimitedly which is impractical for calculating the inverse of $\boldsymbol{\mathcal{I}}_{\mathbf{Y}_{1:k}}\left(\mathbf{X}_{1:k}\right)$. To avoid this issue, a recursive expression for $\boldsymbol{\mathcal{I}}_k := \boldsymbol{\mathcal{I}}_{\mathbf{Y}_{1:k}}\left(\mathbf{x}_k\right)$ has been developed in [181]. This recursion of the parametric CRB is given by [16, 74]:

$$\boldsymbol{\mathcal{I}}_{k+1} = \mathbf{C}_{k+1}^{\mathrm{T}}\mathbf{R}_{k+1}^{-1}\mathbf{C}_{k+1} + \left(\mathbf{A}_k\boldsymbol{\mathcal{I}}_k^{-1}\mathbf{A}_k^{\mathrm{T}} + \mathbf{G}_k\mathbf{Q}_k\mathbf{G}_k^{\mathrm{T}}\right)^{-1} , \tag{4.20a}$$

where

$$\mathbf{A}_k = \nabla_{\mathbf{x}_k}\,\mathbf{f}_k\left(\mathbf{x}_k, \mathbf{u}_k, \mathbf{w}_k\right)\big|_{\mathbf{x}_k^*, \mathbf{w}_k^*} , \tag{4.20b}$$

$$\mathbf{G}_k = \nabla_{\mathbf{w}_k}\,\mathbf{f}_k\left(\mathbf{x}_k, \mathbf{u}_k, \mathbf{w}_k\right)\big|_{\mathbf{x}_k^*, \mathbf{w}_k^*} , \tag{4.20c}$$

$$\mathbf{Q}_k^{-1} = -\mathbb{E}_{\mathbf{x}_k}\left[\Delta_{\mathbf{w}_k}^{\mathbf{w}_k} \log p\left(\mathbf{x}_k, \mathbf{w}_k\right)\big|_{\mathbf{w}_k^*}\right] , \tag{4.20d}$$

$$\mathbf{C}_{k+1}^{\mathrm{T}}\mathbf{R}_{k+1}^{-1}\mathbf{C}_{k+1} = -\mathbb{E}_{\mathbf{y}_{k+1}}\left[\Delta_{\mathbf{x}_{k+1}}^{\mathbf{x}_{k+1}} \log p\left(\mathbf{y}_{k+1}\middle|\mathbf{x}_{k+1}\right)\big|_{\mathbf{x}_{k+1}^*}\right] . \tag{4.20e}$$

4.1.4.2 Posterior CRB

In contrast to the previous section, the probability distribution of the state trajectory $\mathbf{x}_{1:k}$ is now considered instead of a specific realisation only. Instead of a conditional density $p\left(\mathbf{Y}_{1:k}\middle|\mathbf{X}_{1:k}\right)$, one has the joint density $p\left(\mathbf{Y}_{1:k},\mathbf{X}_{1:k}\right)$ and the objective is to find the *Bayesian information matrix*

$$\boldsymbol{\mathcal{I}}_{\mathbf{Y}_{1:k}}\left(\mathbf{X}_{1:k}\right)=-\mathbb{E}_{\mathbf{X}_{1:k},\mathbf{Y}_{1:k}}\left[\Delta_{\mathbf{X}_{1:k}}^{\mathbf{X}_{1:k}}\log p\left(\mathbf{Y}_{1:k},\mathbf{X}_{1:k}\right)\right]\ . \tag{4.21}$$

This bound has been derived in batch matrix form $\boldsymbol{\mathcal{I}}_{\mathbf{Y}_{1:k}}\left(\mathbf{X}_{1:k}\right)$ in [190]. The recursive expressions presented in the following have been reported in [16, 188]:

$$\boldsymbol{\mathcal{I}}_{k+1}=\mathbf{D}_{22,k}-\mathbf{D}_{21,k}\left(\boldsymbol{\mathcal{I}}_{k}+\mathbf{D}_{11,k}\right)^{-1}\mathbf{D}_{12,k}\ , \tag{4.22a}$$

where

$$\mathbf{D}_{11,k}=-\mathbb{E}_{\mathbf{x}_k,\mathbf{w}_k}\left[\Delta_{\mathbf{x}_k}^{\mathbf{x}_k}\log p\left(\mathbf{x}_{k+1}\middle|\mathbf{x}_k\right)\right]\ , \tag{4.22b}$$

$$\mathbf{D}_{12,k}=\mathbf{D}_{21,k}^{\mathrm{T}}=-\mathbb{E}_{\mathbf{x}_k,\mathbf{w}_k}\left[\Delta_{\mathbf{x}_k}^{\mathbf{x}_{k+1}}\log p\left(\mathbf{x}_{k+1}\middle|\mathbf{x}_k\right)\right]\ , \tag{4.22c}$$

$$\mathbf{D}_{22,k}=-\mathbb{E}_{\mathbf{x}_k,\mathbf{w}_k}\left[\Delta_{\mathbf{x}_{k+1}}^{\mathbf{x}_{k+1}}\log p\left(\mathbf{x}_{k+1}\middle|\mathbf{x}_k\right)\right]-\mathbb{E}_{\mathbf{x}_{k+1},\mathbf{y}_{k+1}}\left[\Delta_{\mathbf{y}_{k+1}}^{\mathbf{y}_{k+1}}\log p\left(\mathbf{y}_{k+1}\middle|\mathbf{x}_{k+1}\right)\right]\ . \tag{4.22d}$$

The expectation is now taken over measurements $\mathbf{Y}_{1:k}$ and states $\mathbf{X}_{1:k}$. Instead of evaluating the expressions for a single state trajectory $\mathbf{X}_{1:k}^{*}$, the expectation over the joint density is calculated. Depending on the system equations, finding this density and evaluating the expectations in closed form might be impossible and numerical methods such as the particle filter from Sec. 4.1.3.2 can be used [182].[3]

In the following Sec. 4.1.4.3-4.1.4.4, two special cases of the system (4.16) will be analysed. These are non-linear systems with additive Gaussian noise and linear systems with additive non-Gaussian noise. Here, closed-form expressions for the parametric (4.20) and posterior CRB (4.22) are available.

4.1.4.3 CRB for non-linear systems with additive Gaussian noise

Consider the non-linear system with additive Gaussian noise processes[4] and Gaussian initial state $\mathbf{x}_0\sim\mathcal{N}(\hat{\mathbf{x}}_0,\boldsymbol{\Sigma}_0)$:

$$\mathbf{x}_{k+1}=\mathbf{f}\left(\mathbf{x}_k,\mathbf{u}_k\right)+\mathbf{w}_k\ ,\qquad \mathbf{w}_k\sim\mathcal{N}\left(\mathbf{0},\mathbf{Q}'\right) \tag{4.23a}$$

$$\mathbf{y}_k=\mathbf{h}\left(\mathbf{x}_k\right)+\mathbf{v}_k\ ,\qquad \mathbf{v}_k\sim\mathcal{N}\left(\mathbf{0},\mathbf{R}'\right)\ . \tag{4.23b}$$

[3] At first sight, it is unsatisfying that a theoretical lower bound on the estimator covariance depends on numerical estimates from a particle filter. However, it should be noted that this is still a significant improvement over a purely numerical evaluation where a particle filter is used as a reference filter. In this case, one has to repeat the execution of the particle filter multiple times in order to find the covariance of these reference estimates.

[4] To avoid confusion with the notation used in the parametric CRB (4.20), the covariances are denoted by $(\cdot)'$.

For the sake of clarity, both the system equations and noise covariances are assumed as time-invariant. One obtains for the expressions of the parametric CRB from (4.20) [16]:

$$\mathbf{Q}^{-1} = -\mathbb{E}_{\mathbf{x}_k}\left[\Delta_{\mathbf{w}_k}^{\mathbf{w}_k} - \frac{1}{2}\left(\mathbf{w}_k - \mathbf{f}\left(\mathbf{x}_k, \mathbf{u}_k\right)\right)^{\mathrm{T}} \mathbf{Q}'^{-1}\left(\mathbf{w}_k - \mathbf{f}\left(\mathbf{x}_k, \mathbf{u}_k\right)\right)\Big|_{\mathbf{w}_k^*}\right] = \mathbf{Q}'^{-1} \tag{4.24a}$$

$$\mathbf{C}_{k+1}^{\mathrm{T}}\mathbf{R}_{k+1}^{-1}\mathbf{C}_{k+1} = \left(\nabla_{\mathbf{x}_{k+1}}\mathbf{h}\left(\mathbf{x}_{k+1}\right)\Big|_{\mathbf{x}_{k+1}^*}\right)^{\mathrm{T}} \mathbf{R}'^{-1}\left(\nabla_{\mathbf{x}_{k+1}}\mathbf{h}\left(\mathbf{x}_{k+1}\right)\Big|_{\mathbf{x}_{k+1}^*}\right) . \tag{4.24b}$$

The resulting recursion of the inverse information matrix $\boldsymbol{\mathcal{I}}_{k+1}^{-1}$ resembles the covariance propagation of an extended Kalman filter from Sec. 4.1.3.4 [74]. The main difference lies in the point around which the linearisation is performed which is the true state $\mathbf{x}_k^*$ for the parametric CRB and the current estimate $\hat{\mathbf{x}}_k$ in the filter equations.

The posterior CRB (4.22) is defined by the following expressions [188]:

$$\mathbf{D}_{11,k} = \mathbb{E}_{\mathbf{x}_k,\mathbf{w}_k}\left[\left(\nabla_{\mathbf{x}_k}\mathbf{f}\left(\mathbf{x}_k, \mathbf{u}_k\right)\right)^{\mathrm{T}} \mathbf{Q}'^{-1}\left(\nabla_{\mathbf{x}_k}\mathbf{f}\left(\mathbf{x}_k, \mathbf{u}_k\right)\right)\right] \tag{4.25a}$$

$$\mathbf{D}_{12,k} = -\mathbb{E}_{\mathbf{x}_k,\mathbf{w}_k}\left[\left(\nabla_{\mathbf{x}_k}\mathbf{f}\left(\mathbf{x}_k, \mathbf{u}_k\right)\right)^{\mathrm{T}}\right] \mathbf{Q}'^{-1} \tag{4.25b}$$

$$\mathbf{D}_{22,k} = \mathbf{Q}'^{-1} + \mathbb{E}_{\mathbf{x}_{k+1},\mathbf{y}_{k+1}}\left[\left(\nabla_{\mathbf{x}_{k+1}}\mathbf{h}\left(\mathbf{x}_{k+1}\right)\right)^{\mathrm{T}} \mathbf{R}'^{-1}\left(\nabla_{\mathbf{x}_{k+1}}\mathbf{h}\left(\mathbf{x}_{k+1}\right)\right)\right] . \tag{4.25c}$$

In the general case of non-linear system and measurement equations, the Jacobians depend on the state $\mathbf{x}_k$. Often, no closed-form expressions for the expected values of the non-linear transformations in (4.25) are available. The special case of a linear system will be further elaborated on in the next section.

4.1.4.4 CRB for linear systems with additive non-Gaussian noise

The system equations are linear and time-invariant:

$$\mathbf{x}_{k+1} = \mathbf{A}\mathbf{x}_k + \mathbf{B}\mathbf{u}_k + \mathbf{w}_k , \quad \mathbf{w}_k \sim p\left(\mathbf{w}_k\right) , \quad \mathbb{E}\left[\mathbf{w}_k\right] = \mathbf{0} , \quad \operatorname{cov}\left(\mathbf{w}_k\right) = \mathbf{Q}' \tag{4.26a}$$

$$\mathbf{y}_k = \mathbf{C}\mathbf{x}_k + \mathbf{v}_k , \quad \mathbf{v}_k \sim p\left(\mathbf{v}_k\right) , \quad \mathbb{E}\left[\mathbf{v}_k\right] = \mathbf{0} , \quad \operatorname{cov}\left(\mathbf{v}_k\right) = \mathbf{R}' . \tag{4.26b}$$

The intrinsic accuracies (2.18) of the independent noise processes are denoted by $\boldsymbol{\mathcal{I}}_{\mathbf{w}}$ and $\boldsymbol{\mathcal{I}}_{\mathbf{v}}$. A distribution $\hat{\mathbf{x}}_0 \sim p\left(\mathbf{x}_0\right)$ with intrinsic accuracy $\boldsymbol{\mathcal{I}}_0$ is assumed for the initial state.

The parametric bound (4.20) for this system is found as [74]:

$$\boldsymbol{\mathcal{I}}_{k+1} = \mathbf{C}^{\mathrm{T}}\boldsymbol{\mathcal{I}}_{\mathbf{v}}\mathbf{C} + \left(\mathbf{A}\boldsymbol{\mathcal{I}}_k^{-1}\mathbf{A}^{\mathrm{T}} + \boldsymbol{\mathcal{I}}_{\mathbf{w}}^{-1}\right)^{-1} , \tag{4.27a}$$

since

$$\mathbf{A}_k = \nabla_{\mathbf{x}_k}\left[\mathbf{A}\mathbf{x}_k + \mathbf{B}\mathbf{u}_k + \mathbf{w}_k\right]|_{\mathbf{x}_k^*,\mathbf{w}_k^*} = \mathbf{A} , \tag{4.27b}$$

$$\mathbf{G}_k = \nabla_{\mathbf{w}_k}\left[\mathbf{A}\mathbf{x}_k + \mathbf{B}\mathbf{u}_k + \mathbf{w}_k\right]|_{\mathbf{x}_k^*,\mathbf{w}_k^*} = \mathbf{I}_{n\times n} , \tag{4.27c}$$

$$\mathbf{Q}_k^{-1} = -\mathbb{E}_{\mathbf{x}_k}\left[\Delta_{\mathbf{w}_k}^{\mathbf{w}_k} \log p\left(\mathbf{x}_k, \mathbf{w}_k\right)\Big|_{\mathbf{w}_k^*}\right] = \boldsymbol{\mathcal{I}}_{\mathbf{w}} , \tag{4.27d}$$

$$\mathbf{C}_{k+1}^{\mathrm{T}}\mathbf{R}_{k+1}^{-1}\mathbf{C}_{k+1} = -\mathbb{E}_{\mathbf{y}_{k+1}}\left[\Delta_{\mathbf{x}_{k+1}}^{\mathbf{x}_{k+1}} \log p\left(\mathbf{y}_{k+1}\middle|\mathbf{x}_{k+1}\right)\Big|_{\mathbf{x}_{k+1}^*}\right] = \mathbf{C}^{\mathrm{T}}\boldsymbol{\mathcal{I}}_{\mathbf{v}}\mathbf{C} . \tag{4.27e}$$

The recursion resembles the covariance propagation of a linear Kalman filter from Sec. 4.1.3.3. For the posterior CRB, one obtains the following expressions for (4.22) [74]:

$$\mathbf{D}_{11} = \mathbf{A}^{\mathrm{T}} \boldsymbol{\mathcal{I}}_{\mathbf{w}} \mathbf{A} \,, \qquad \mathbf{D}_{12} = -\mathbf{A}^{\mathrm{T}} \boldsymbol{\mathcal{I}}_{\mathbf{w}} \,, \qquad \mathbf{D}_{22} = \boldsymbol{\mathcal{I}}_{\mathbf{w}} + \mathbf{C}^{\mathrm{T}} \boldsymbol{\mathcal{I}}_{\mathbf{v}} \mathbf{C} \,. \tag{4.28a}$$

Note that after inserting these into the recursion for $\boldsymbol{\mathcal{I}}_{k+1}$, one has the same result as the parametric CRB (4.27a):

$$\boldsymbol{\mathcal{I}}_{k+1} = \mathbf{C}^{\mathrm{T}} \boldsymbol{\mathcal{I}}_{\mathbf{v}} \mathbf{C} + \underbrace{\boldsymbol{\mathcal{I}}_{\mathbf{w}} - \boldsymbol{\mathcal{I}}_{\mathbf{w}} \mathbf{A} \left(\boldsymbol{\mathcal{I}}_k + \mathbf{A}^{\mathrm{T}} \boldsymbol{\mathcal{I}}_{\mathbf{w}} \mathbf{A} \right)^{-1} \mathbf{A}^{\mathrm{T}} \boldsymbol{\mathcal{I}}_{\mathbf{w}}}_{\overset{(A.9)}{=} \left(\mathbf{A} \boldsymbol{\mathcal{I}}_k^{-1} \mathbf{A}^{\mathrm{T}} + \boldsymbol{\mathcal{I}}_{\mathbf{w}}^{-1} \right)^{-1}} \,. \tag{4.29}$$

An explanation for the equality between the parametric and posterior bound is that in linear systems, the Jacobians of the state transition and measurement functions do not depend on the state. Thus, it is irrelevant whether a specific trajectory of the state or, from a Bayesian perspective, the entire state distribution is considered [74].

Another observation is that the result (4.29) is independent of the state trajectory, for example the manoeuvre driven by an observed vehicle. The bound only depends on the time step k. This finding confirms the initial motivation for analytical modelling from Sec. 1.2, namely that careful inspection of a problem may help to find invariant aspects and thus reduces the modelling effort.

Moreover, the recursion given by (4.22) can be recognised as a discrete-time algebraic Riccati equation with constant coefficients after reformulation [188]:

$$\begin{aligned}
\boldsymbol{\mathcal{I}}_{k+1} &= \mathbf{D}_{22} - \mathbf{D}_{21} \left(\boldsymbol{\mathcal{I}}_k + \mathbf{D}_{11} \right)^{-1} \mathbf{D}_{12} \\
&= \mathbf{D}_{22} - \mathbf{D}_{21} \mathbf{D}_{11}^{-1} \left(\mathbf{D}_{11} - \left(\mathbf{D}_{11} - \mathbf{D}_{11} \left(\boldsymbol{\mathcal{I}}_k + \mathbf{D}_{11} \right)^{-1} \mathbf{D}_{11} \right) \right) \mathbf{D}_{11}^{-1} \mathbf{D}_{12} \\
&= \mathbf{D}_{21} \mathbf{D}_{11}^{-1} \boldsymbol{\mathcal{I}}_k \mathbf{D}_{11}^{-1} \mathbf{D}_{12} + \mathbf{D}_{21} \mathbf{D}_{11}^{-1} \boldsymbol{\mathcal{I}}_k \left(\boldsymbol{\mathcal{I}}_k + \mathbf{D}_{11} \right)^{-1} \boldsymbol{\mathcal{I}}_k \mathbf{D}_{11}^{-1} \mathbf{D}_{12} \\
&\quad + \mathbf{D}_{22} - \mathbf{D}_{21} \mathbf{D}_{11}^{-1} \mathbf{D}_{12} \\
&= \tilde{\mathbf{A}}^{\mathrm{T}} \boldsymbol{\mathcal{I}}_k \tilde{\mathbf{A}} - \left(\tilde{\mathbf{A}}^{\mathrm{T}} \boldsymbol{\mathcal{I}}_k \tilde{\mathbf{B}} \right) \left(\tilde{\mathbf{R}} + \tilde{\mathbf{B}}^{\mathrm{T}} \boldsymbol{\mathcal{I}}_k \tilde{\mathbf{B}} \right)^{-1} \left(\tilde{\mathbf{B}}^{\mathrm{T}} \boldsymbol{\mathcal{I}}_k \tilde{\mathbf{A}} \right) + \tilde{\mathbf{Q}} \\
\text{with } \tilde{\mathbf{A}} &\coloneqq \mathbf{D}_{11}^{-1} \mathbf{D}_{12} \,, \quad \tilde{\mathbf{B}} \coloneqq \mathbf{I}_{n \times n} \,, \quad \tilde{\mathbf{R}} \coloneqq \mathbf{D}_{11} \,, \quad \tilde{\mathbf{Q}} \coloneqq \mathbf{D}_{22} - \mathbf{D}_{21} \mathbf{D}_{11}^{-1} \mathbf{D}_{12} \,.
\end{aligned} \tag{4.30}$$

It is thus possible to calculate the asymptotic steady state solution $\boldsymbol{\mathcal{I}}_\infty$ for $k \to \infty$, if it exists. This single quantity can be used as a metric for comparing different sensors with their measurement quality described by the intrinsic accuracy $\boldsymbol{\mathcal{I}}_{\mathbf{v}}$ [74].

Similarly, the difference between the optimal Bayes filter and a Kalman filter can be investigated. The latter only incorporates the covariance matrices $\mathbf{Q}$ and $\mathbf{R}$ of the potentially non-Gaussian prediction and measurement noise. Recall from (2.19), that describing a non-Gaussian noise process $\mathbf{e}$ solely by its second order moment $\operatorname{cov}(\mathbf{e})$ carries less information than the intrinsic accuracy $\boldsymbol{\mathcal{I}}_{\mathbf{e}}$. It is shown in [74] that this inequality propagates through the CRB recursion. Consequently, employing the full noise information leads to more accurate state estimates than the approximation by a Gaussian.

While this result is not surprising, the CRB and its steady-state value enable to analytically quantify this difference and to reason whether a more accurate implementation of a Bayes filter is worth the effort over a Kalman filter. This will be exemplified in the following Sec. 4.1.5 for the use of stereo vision for the estimation of vehicle motion. The non-Gaussianity of these measurements has been discussed in Sec. 3.1.4.4.

In practical cases, the noise processes are often time-dependent. An approach to draw conclusions on the asymptotic time evolution is presented in [140]. Time-invariant enclosures of the covariance matrices have to be defined for which the steady state value of the CRB can be derived.

4.1.4.5 Extending the posterior CRB to multi-target tracking

Having studied the Cramér-Rao bound for the state estimation of a single target, a natural but challenging question concerns the extension to multiple targets and measurements with uncertain origin as introduced in Sec. 4.1.3.5. The central objective is of course not only to formulate this bound but to obtain tractable expressions, as it is the case for the elegant recursion of the posterior CRB (4.22). The general case of multi-target tracking must be considered as yet unsolved in this regard but solutions to partial problems or approximations have been obtained.

Two cases can be distinguished here: On the one hand, the single target tracking problem can be extended to measurement models with missed detections and false positive observations. It has been found, that this degradation of the measurement information can be included as an additional scalar *information reduction factor* in the calculation of the CRB [79]. However, this remarkably simple result is overly optimistic since the order in which nominal and false measurements may occur is not considered [78]. A tight bound can be derived with this taken into account but at the cost of more complex formulations [77, 138].

While fruitful work has been devoted to the case of a single or multiple, sufficiently separated targets, the CRB for a multi-target tracking problem on the other hand remains only partially solved. With different approaches and assumptions, recent results have been obtained in [93, 121, 159]. The main question here is to establish how much the achievable covariance of the state estimates increases if the association between measurements needs to be estimated at the same time.

As has been shown in [44], the assumption of a perfectly known association always leads to an overly optimistic bound. This means that at least a lower variance bound can be obtained from the single target tracking case by simply considering each object separately. Such an optimistic bound is still useful as it provides necessary conditions on the sensor accuracy in order to achieve a state estimation uncertainty below a maximal tolerable value. A tight bound on the other hand is of great interest for the development of multi-target tracking algorithms, in order to quantify the gap between the performance of an approximate algorithm and the theoretical limit.

4.1.5 Numerical example: Vehicle motion estimation

The theoretical foundations are now illustrated with the example of estimating a vehicle's motion from range sensor measurements. Thus, the state estimation problem is based on the following discrete-time dynamic system of a one-dimensional motion with distance x and constant velocity v_x:

$$\begin{bmatrix} x_{k+1} \\ v_{x,k+1} \end{bmatrix} = \begin{bmatrix} 1 & T_\mathrm{S} \\ 0 & 1 \end{bmatrix} \begin{bmatrix} x_k \\ v_{x,k} \end{bmatrix} + \mathbf{w}_k \ , \qquad \mathbf{w}_k \sim \mathcal{N}(\mathbf{0}, \mathbf{Q}) \ . \tag{4.31}$$

Two different measurement models will be studied in the following. State estimators will be applied to the system and the performance of these filters will be compared to the theoretical CRB.

Firstly, the intrinsic accuracy $\mathcal{I}_\mathbf{v}$ in the measurements will be derived and compared to the information in Gaussian approximations in Sec. 4.1.5.1. Based on these results, the CRB is subsequently analysed in Sec. 4.1.5.2.

4.1.5.1 Information in sensor measurement distributions

Two measurement models will be analysed: Firstly, measurements of the Cartesian distance are obtained by a stereo vision sensor as explained in Sec. 3.1.4.4. Secondly, distance measurements that are affected by additive Gaussian noise with outlier values are studied. A Gaussian mixture distribution with two components is employed to model outliers which are not explained by the expected fluctuation of an ordinary measurement.

Distance measurements by stereo vision: Measurements y_k of a true distance x_k are assumed to be obtained by a stereo video sensor with Gaussian disparity errors:

$$y_k = \frac{c_\mathrm{k} b_\mathrm{w}}{d_k} \ , \qquad d_k \sim \mathcal{N}\left(\frac{c_\mathrm{k} b_\mathrm{w}}{x_k}, \sigma_d^2 \right) \ . \tag{4.32}$$

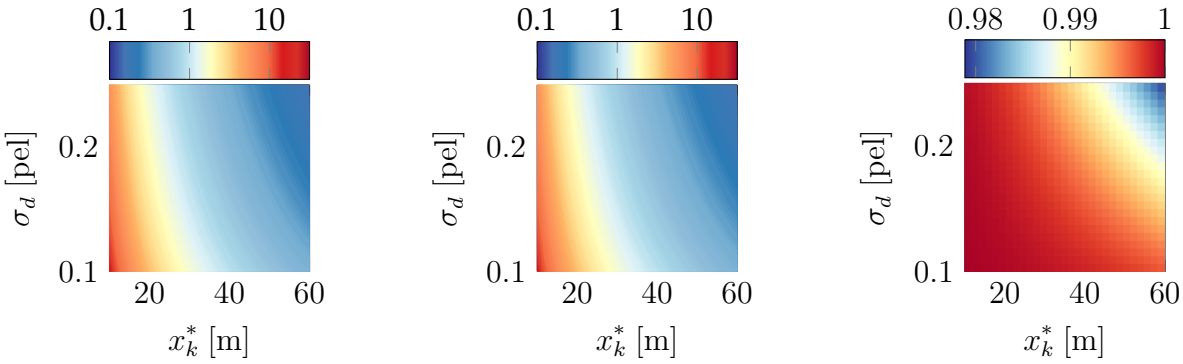
According to (3.25), one has the following non-Gaussian measurement model:

$$p\left(y_k \middle| x_k\right) = \frac{c_\mathrm{k} b_\mathrm{w}}{\sqrt{2\pi}\sigma_d} \frac{1}{y_k^2} \exp\left(-\frac{1}{2} \left(\frac{c_\mathrm{k} b_\mathrm{w}}{\sigma_d} \right)^2 \left(\frac{1}{y_k} - \frac{1}{x_k} \right)^2 \right) \ . \tag{4.33}$$

The parametric CRB[5] (4.20) will be derived for this model. To this end, (4.33) is inserted into (4.20e) with a deterministic true distance x_k^*:

$$\begin{aligned} \mathcal{I}_{y_k}(x_k) &= -\mathbb{E}_{y_k}\left[\Delta_{x_k}^{x_k} -\frac{1}{2} \left(\frac{c_\mathrm{k} b_\mathrm{w}}{\sigma_d} \right)^2 \left(\frac{1}{y_k} - \frac{1}{x_k} \right)^2 \Bigg|_{x_k^*} \right] \\ &= -\mathbb{E}_{y_k}\left[\frac{c_\mathrm{k} b_\mathrm{w}}{y_k} \right] \cdot \frac{2}{(x_k^*)^3} \frac{c_\mathrm{k} b_\mathrm{w}}{\sigma_d^2} + \frac{3}{(x_k^*)^4} \frac{(c_\mathrm{k} b_\mathrm{w})^2}{\sigma_d^2} = \left(\frac{c_\mathrm{k} b_\mathrm{w}}{\sigma_d} \right)^2 \cdot \frac{1}{(x_k^*)^4} \ . \end{aligned} \tag{4.34}$$

[5] Numerical methods have to be employed to calculate the posterior CRB from (4.22).



(a) Intrinsic accuracy $\mathcal{I}_{y_k}(x_k)$ from (4.34).

(b) Inverse variance $\operatorname{var}(y_k)^{-1}$ from (4.36).

(c) Inverse relative accuracy $\Psi_{d_k}^{-1}$.

Figure 4.1 Comparison of the intrinsic accuracy of Gaussian disparity measurements (a) to the variance of an approximated measurement model (b). The ratio between both quantities is shown in (c). The variance $\operatorname{var}(y_k)$ is numerically evaluated with $N_{\text{sim}} = 1 \times 10^7$ Monte-Carlo iterations.

The result depends on σ_d and x_k^* and is visualised over a range of value pairs in Fig. 4.1(a). Increasing the measurement noise standard deviation or the distance leads to less informative measurements.

Implementing the optimal Bayes filter requires the particle filter framework from Sec. 4.1.3.2. This might be undesirable from a computational point of view. Instead, an extended Kalman filter from Sec. 4.1.3.4 can be used where the measurement density (4.33) is approximated by a Gaussian in each step:

$$p(y_k|x_k) \approx \mathcal{N}(y_k; x_k, \sigma_{y_k}^2) \ , \qquad y_k = \frac{c_\text{k} b_\text{w}}{d_k} \ , \qquad \sigma_{y_k}^2 = \left(\frac{c_\text{k} b_\text{w}}{x_k^2}\right)^2 \sigma_d^2 \ . \tag{4.35}$$

The goodness of this approximation can be evaluated by comparing the variance $\operatorname{var}(y_k)$ to the intrinsic accuracy (4.34). The variance is numerically evaluated and displayed in Fig. 4.1(b). A second order series expansion yields the following approximation [74]:

$$\operatorname{var}(y_k) = \mathbb{E}_{d_k}\left[\left(\frac{c_\text{k} b_\text{w}}{d_k} - \mathbb{E}_{d_k}\left[\frac{c_\text{k} b_\text{w}}{d_k}\right]\right)^2\right] \approx \left(\frac{\sigma_d}{c_\text{k} b_\text{w}}\right)^2 \cdot (x_k^*)^4 + 2\left(\frac{\sigma_d}{c_\text{k} b_\text{w}}\right)^4 \cdot (x_k^*)^6 \ . \tag{4.36}$$

Indeed, it holds that $\operatorname{var}(y_k) \geq \mathcal{I}_{y_k}^{-1}(x_k)$ as is given by the general inequality (2.19). The relation between $\mathcal{I}_{y_k}$ from (4.34) and the Gaussian approximation is furthermore characterised by the relative accuracy $\Psi_{y_k} = \mathcal{I}_{y_k}(x_k) \operatorname{var}(y_k)$ which is shown in Fig. 4.1(c). It can be observed that this ratio remains very close to one with the highest deviations at higher distances and noise levels.

All in all it is thus expected that an extended Kalman filter, implicitly designed for the Gaussian measurement model (4.35), leads to similar performance in terms of estimator covariance as an optimal Bayes filter and reaches the Cramér-Rao bound.

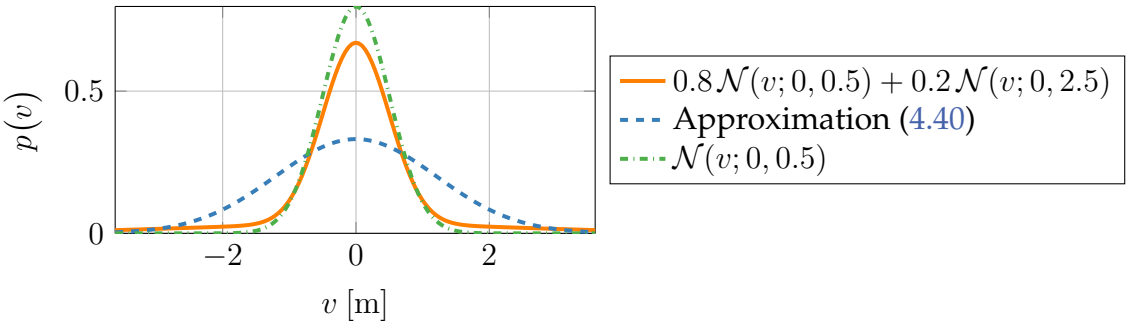
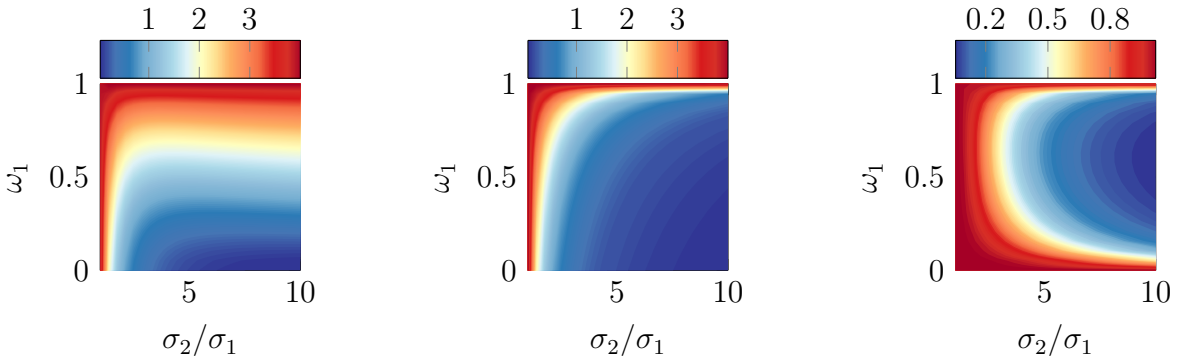


Figure 4.2 Probability density functions of a bi-Gaussian distribution compared to a Gaussian approximation and the density of the first mode.



(a) Intrinsic accuracy $\mathcal{I}_{y_k}(x_k)$ from (4.38).

(b) Inverse variance $\mathrm{var}(y_k)^{-1}$ from (4.40).

(c) Inverse relative accuracy $\Psi_{y_k}^{-1}$.

Figure 4.3 Comparison of a bi-Gaussian distribution to a Gaussian approximation.

Bi-Gaussian measurement errors: The second measurement model assumes that the errors follow a Gaussian mixture model with two components (bi-Gaussian distribution). Such a model has been proposed in [104] to describe range measurements from an automotive radar sensor. Besides Gaussian noise on the measured range values, a shift of the reflection point on the observed vehicle is included as a second cause for a measurement's fluctuation. The model therefore consists of a dominant Gaussian with variance $\sigma_{y,1}^2$ and weight ω_1 and a second mode with higher variance $\sigma_{y,2}^2$ and weight $1-\omega_1$:

$$\begin{aligned} y_k = x_k + v_k\,, \quad p(v_k) &= \omega_1 \mathcal{N}(v_k;0,\sigma_{y,1}^2) + (1-\omega_1)\,\mathcal{N}(v_k;0,\sigma_{y,2}^2) \\ \Longrightarrow p(y_k|x_k) &= \omega_1 \mathcal{N}(y_k;x_k,\sigma_{y,1}^2) + (1-\omega_1)\,\mathcal{N}(y_k;x_k,\sigma_{y,2}^2) \end{aligned}\,. \tag{4.37}$$

Fig. 4.2 depicts the probability density functions for an exemplary parameter choice with $\omega_1 = 0.8$, $\sigma_{y,1} = 0.5\,\mathrm{m}$ and $\sigma_{y,2} = 2.5\,\mathrm{m}$. Compared to solely the first mode, the bi-Gaussian density features heavier tails. An estimator which ignores these tails will be overconfident about measurements with a high deviation from the expected position and thus cause a drift towards these outliers.

Due to the flexibility of multimodal Gaussian distributions, these have been extensively studied for a range of different state estimation tasks in [74]. Some of these results will be reproduced in the following for the specific case of the bi-Gaussian model (4.37). Furthermore, the comparison of different filters in Sec. 4.1.5.2 will be extended to the ad hoc approach of a Kalman filter with gating as described in Sec. 4.1.3.5.

First, similar to (4.34), the information $\mathcal{I}_{y_k}(x_k)$ from a measurement y_k on x_k is derived. Since (4.37) is a linear model, this equals the intrinsic accuracy $\mathcal{I}_v$ of the additive noise and is independent of the state:

$$\begin{aligned}\mathcal{I}_{y_k}(x_k) = \mathcal{I}_v &= -\mathbb{E}_{y_k}\left[\Delta_{x_k}^{x_k} \log\left(\omega_1 \mathcal{N}\left(y_k; x_k, \sigma_{y,1}^2\right) + (1-\omega_1)\,\mathcal{N}\left(y_k; x_k, \sigma_{y,2}^2\right)\right)\Big|_{x_k^*}\right] \\ &= \int_{v=-\infty}^{\infty} v^2 \frac{\left(\frac{\omega_1}{\sigma_{y,1}^2}\mathcal{N}\left(v; 0, \sigma_{y,1}^2\right) + \frac{1-\omega_1}{\sigma_{y,2}^2}\mathcal{N}\left(v; 0, \sigma_{y,2}^2\right)\right)^2}{\omega_1 \mathcal{N}\left(v; 0, \sigma_{y,1}^2\right) + (1-\omega_1)\,\mathcal{N}\left(v; 0, \sigma_{y,2}^2\right)}\,\mathrm{d}v\,. \end{aligned} \tag{4.38}$$

In the special cases of either $\omega_1 = 1$ or $\sigma_{y,1} = \sigma_{y,2}$, the bi-Gaussian distribution reduces to a Gaussian and it can be seen from (4.38) that $\mathcal{I}_{y_k}(x_k) = \left(\sigma_{y,1}^2\right)^{-1}$. Moreover, if the second mode becomes entirely uninformative with $\sigma_{y,2} \to \infty$, one obtains the following inequality:

$$\mathcal{I}_{y_k}(x_k) \leq \frac{\omega_1}{\sigma_{y,1}^4} \int_{v=-\infty}^{\infty} v^2 \mathcal{N}\left(v; 0, \sigma_{y,1}^2\right)\,\mathrm{d}v = \frac{\omega_1}{\sigma_{y,1}^2}\,. \tag{4.39}$$

The integral (4.38) is numerically evaluated for different combinations of $\omega_1 \in [0.8; 1]$, $\sigma_{y,1} = 0.5\,\mathrm{m}$ and $\frac{\sigma_{y,2}}{\sigma_{y,1}} \in [1; 10]$ and the results are visualised in Fig. 4.3(a). Since the weight of the first mode remains relatively high, the information varies only slightly for the evaluated parameter values.

Second, a Gaussian approximation of the measurement noise in (4.37) is obtained as

$$p\left(y_k \middle| x_k\right) \approx \mathcal{N}\left(y_k; x_k, \omega_1 \sigma_{y,1}^2 + (1-\omega_1)\,\sigma_{y,2}^2\right)\,. \tag{4.40}$$

As is seen in the exemplary visualisation in Fig. 4.2, the approximate distribution is overly wide around the mean. This illustrates how information on the distribution's mean is lost due to the approximation.

The inverse of the variance $\mathrm{var}(y_k)$, that is the information carried by the Gaussian approximation, is displayed in Fig. 4.3(b). A noticeable decrease can be observed when increasing the variance or weight of the second mode. The comparison to $\mathcal{I}_{y_k}(x_k)$ in terms of the relative accuracy Ψ_{y_k} in Fig. 4.3(c) confirms that the deviation between the true and approximated distributions increases, the stronger the outlier mode becomes.

4.1.5.2 Time evolution of the Cramér-Rao bound

After the measurement models and approximations thereof have been investigated, these findings can be applied to a state estimation task. The simulated scenario consists of

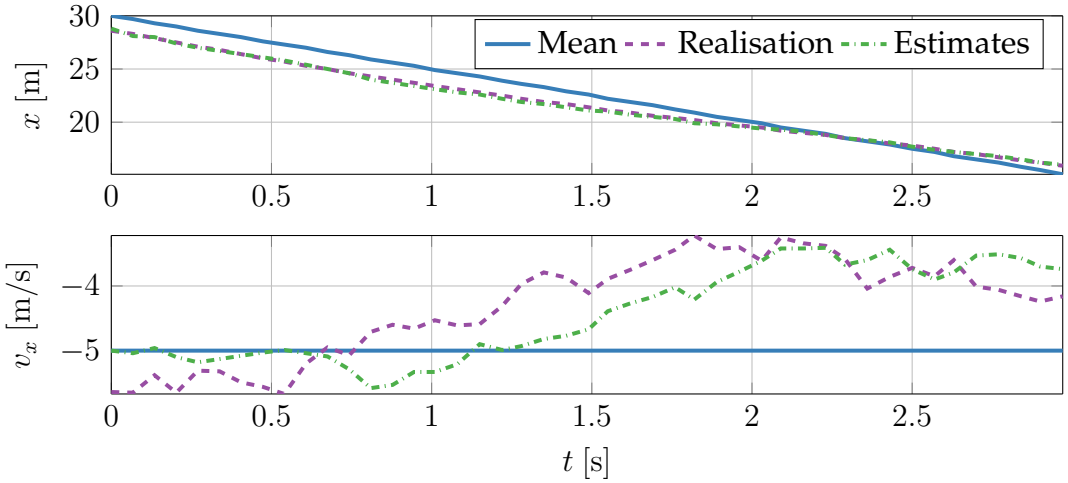


Figure 4.4 Trajectory of the state $\mathbf{x}_k$ (top: distance x_k, below: velocity $v_{x,k}$) in the simulated system (4.31) parametrised according to Tab. 4.1. The mean trajectory $\mathbb{E}[\mathbf{x}_k]$, one random realisation $\mathbf{x}_k^{(i)}$ and the corresponding filter estimates $\hat{\mathbf{x}}_k^{(i)}$ are shown.

an approaching vehicle with the distance and relative velocity shown in Fig. 4.4. Due to the stochastic process noise, each of the $i = 1, \ldots, N_{\text{sim}}$ realisations of the dynamic state $\mathbf{x}_k^{(i)}, k = 1, \ldots, N$ follows a random trajectory around the nominal mean $\mathbb{E}[\mathbf{x}_k]$. The parameter values used in the simulation are given in Tab. 4.1. For each trajectory, simulated distance measurements are generated according to the two models from Sec. 4.1.5.1. Depending on the model, different filters are employed to obtain state estimates $\hat{\mathbf{x}}_k^{(i)}, k = 1, \ldots, N$. These estimates are evaluated in terms of the mean squared error over all N_{sim} simulation runs:

$$\operatorname{cov}(\hat{\mathbf{x}}_k - \mathbf{x}_k) = \frac{1}{N_{\text{sim}} - 1} \sum_{i=1}^{N_{\text{sim}}} \left(\hat{\mathbf{x}}_k^{(i)} - \mathbf{x}_k^{(i)}\right) \left(\hat{\mathbf{x}}_k^{(i)} - \mathbf{x}_k^{(i)}\right)^{\mathrm{T}} . \tag{4.41}$$

The parametric and posterior bounds are calculated for the respective models and it will be reasoned whether the variance of the simulated filters can reach the lower bounds.

Distance measurements by stereo vision: An extended Kalman filter as outlined in Sec. 4.1.3.4 is applied to the non-linear measurement model (4.33). Due to the small difference in terms of the relative accuracy between this approximation and the true probability distribution which has been observed in Fig. 4.1, it is expected that the EKF comes close to the CRB.

Concerning the Cramér-Rao bounds, the parametric CRB is calculated from (4.34) where the linearisation is performed around the nominal mean trajectory. For the posterior bound on the other hand, the expressions (4.22) are numerically evaluated by employing a particle filter with $N_{\text{s}} = 1 \times 10^3$ samples.

The results in Fig. 4.5 show that the estimation error reaches the CRB and thereby confirms the hypothesised outcome. It can be observed how the estimation error improves

Table 4.1 Simulation parameters for analysing the time evolution of the CRB.

Variable	Value
Initial state mean	$\hat{\mathbf{x}}_0 = \begin{bmatrix} 30\,\mathrm{m} & -5\,\mathrm{m/s} \end{bmatrix}$
Initial state covariance	$\mathbf{\Sigma}_0 = \mathrm{diag}\left(1\,\mathrm{m}^2, 0.56\,\mathrm{m}^2/\mathrm{s}^2\right)$
Process noise	$\mathbf{Q} = 0.4\,\mathrm{m}^2/\mathrm{s}^4\mathrm{s}^{-1} \begin{bmatrix} \frac{1}{3}{T_\mathrm{S}}^3 & \frac{1}{2}{T_\mathrm{S}}^2 \\ \frac{1}{2}{T_\mathrm{S}}^2 & T_\mathrm{S} \end{bmatrix}$
Measurement model	Stereo vision: $\sigma_d = 0.1\,\mathrm{pel}$, $c_\mathrm{k} b_\mathrm{w} = 121\,\mathrm{m} \cdot \mathrm{pel}$ Bi-Gaussian noise: $\omega_1 = 0.8$, $\sigma_{y,1} = 0.5\,\mathrm{m}$, $\sigma_{y,2} = 2.5\,\mathrm{m}$
Sampling time	$T_\mathrm{S} = 0.0675\,\mathrm{s}$
Monte-Carlo iterations	$N_\mathrm{sim} = 1 \times 10^4$
Particle filter samples	$N_\mathrm{s} = 1 \times 10^3$

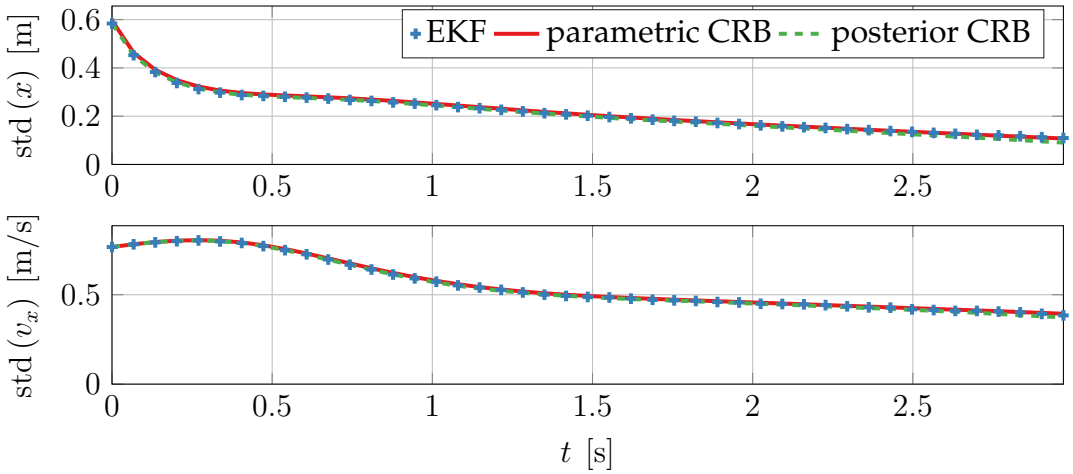


Figure 4.5 Standard deviation of the state estimates for the stereo vision measurement model (4.33) compared to the CRB. (Top: distance x_k, below: velocity $v_{x,k}$)

over time. Since this is a non-linear system, no asymptotic limit of the CRB is expected. Furthermore, the graphs reveal no observable difference between the parametric and posterior CRB. Therefore, the posterior density is sufficiently tight with respect to the gradient of the measurement model. Taking the expectation yields a similar result as an evaluation at the mean trajectory.

Gaussian measurements with outliers: Secondly, the bi-Gaussian measurement distribution (4.37) is analysed. This corresponds to a linear system with additive noise from Sec. 4.1.4.4. Three different strategies to cope with the non-Gaussian noise process are applied. First, a linear Kalman filter from Sec. 4.1.3.3 is parametrised with the variance of the Gaussian approximation (4.40) of the measurement noise. As has been seen in Fig. 4.2, this approximation overestimates a measurement's uncertainty around the mean

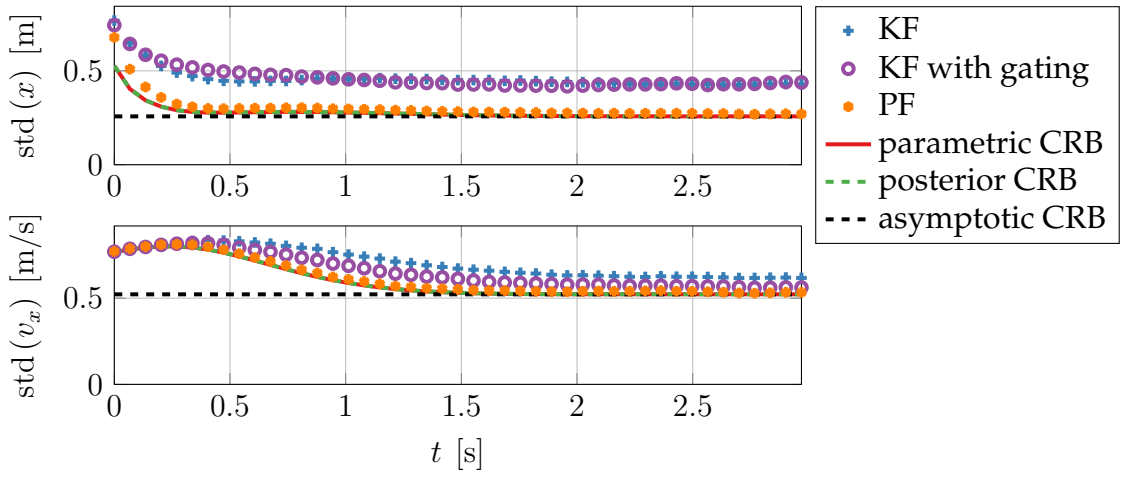


Figure 4.6 Standard deviation of the state estimates for the bi-Gaussian measurement model (4.37) compared to the CRB. (Top: distance x_k, below: velocity $v_{x,k}$)

and thus neglects valuable information.

A second strategy is to consider only the first mode $p(y_k|x_k) \approx \mathcal{N}(y_k, x_k, \sigma_{y,1}^2)$. The occurrence of measurements which are far from the current prediction have to be treated separately in the filter then. Due to the second mode of the true distribution, these outliers will occur more frequently than expected. To this end, a gating strategy as introduced in Sec. 4.1.3.5 is used. A χ^2 test with a $1-\alpha = 99\,\%$ acceptance interval is performed on the normalised squared innovation error $\boldsymbol{\epsilon}_{k+1}^{\mathrm{T}} \boldsymbol{\Sigma}_{\boldsymbol{\epsilon}_{k+1}}^{-1} \boldsymbol{\epsilon}_{k+1}$ from (4.12a). If a measurement falls out of the acceptance interval, the update step is skipped and the state prediction is used as an estimate. One advantage of this ad hoc approach over a completely probabilistic treatment is that it does not require a precise model of the outlier distribution. The method is therefore robust to different outlier distributions. However, neither of the two approaches takes the full noise distribution into account and it is thus not expected that the CRB will be reached.

Thirdly, a particle filter as described in Sec. 4.1.3.2 is implemented with $N_\mathrm{s} = 1 \times 10^3$ particles for the exact bi-Gaussian measurement model (4.37). The particles approximate the posterior density $p(\mathbf{x}_k|\mathbf{y}_{1:k})$ and the distribution's mean is used as a state estimate $\hat{\mathbf{x}}_k$. Since the full noise information is included, it is expected that the variance of the estimation error reaches the CRB and thus confirms the analytical bounds.

Due to the model's linearity, the parametric and posterior CRB are identical and given in closed form by (4.29). Here, the intrinsic accuracy $\mathcal{I}_v$ of the bi-Gaussian noise distribution from (4.38) has to be inserted. Moreover, an asymptotic value exists which is calculated by numerically solving the Riccati equation (4.30).

Fig. 4.6 depicts the standard deviations of the filter estimates and the CRB for comparison. The first Kalman filter that is based on the approximation (4.40) shows a lasting difference to the CRB. Improved estimates of v_x are observed for the second Kalman filter that is enhanced with the additional gating strategy. It has to be remarked that the quality of the estimates is sensitive to the chosen gating threshold. The acceptance

interval $1-\alpha$ has to be chosen in order to achieve a sensible compromise between outlier rejection and loss of measurement information.

As is expected, the particle filter estimates reach the CRB and confirm the bound. Therefore, knowing the CRB is of great value since one can quantify the potential for further improvement from a sub-optimal to a more sophisticated approach.

4.1.6 Summary

The goal of this section has been to model the uncertainty in a tracking filter as it can be used to estimate the velocity of a vehicle from range sensor measurements. To this end, the best attainable estimator variance given by the Cramér-Rao bound (CRB) has been studied. A numerical example, which is used to illustrate the bound, considers the task of tracking a vehicle. Nevertheless, state estimation as introduced in Sec. 4.1.3 plays an important role in other applications as well, for example localisation. Hence, by studying the CRB in Sec. 4.1.4, one is rewarded with a generic concept to draw conclusions for different driver assistance tasks. The derivation of the CRB has been exemplified for two different measurement models in Sec. 4.1.5 and compared to tracking filter estimates in simulations.

One question which can be answered with the CRB is whether the accuracy of a given sensor meets the necessary requirement to achieve a desired certainty of the state estimates. Moreover, the expected deviation of a sub-optimal filter algorithm in contrast to the optimal solution can be evaluated. In the context of embedded devices, slightly less accurate estimates might be acceptable in exchange for improved run time.

Two aspects determine the difficulty in calculating the CRB. These are whether the system model equations are linear or non-linear and whether the noise processes follow Gaussian or non-Gaussian distributions. Closed-form expressions can be obtained for linear systems with additive noise. Here, non-Gaussian instead of Gaussian noise can be seamlessly included by using the inverse intrinsic accuracy instead of the covariance, as developed in [74]. This powerful generalisation allows for straight-forward evaluation of the best attainable performance. The optimal estimator itself is unlikely to exist in closed form other than for special cases.

Compared to the theory of state estimation, work on the posterior CRB is a relatively new field. The major achievement in this regard was the derivation of the recursive formula (4.22) in [16, 188]. An open question is how the involved expectations can be efficiently evaluated for specific models to achieve interpretable analytic results. As it has been shown, even the extension from a Gaussian to a bi-Gaussian measurement distribution leads to complicated integrals. Conducting further in-depth analyses for the specific sensors and system models used in driver assistance applications could aid the development and objective evaluation of tracking algorithms for intelligent vehicles.

4.2 Vehicle motion models for short-term predictions

Situation analysis in traffic scenes requires making accurate predictions on the future motion of other traffic participants, for example vehicles. Likewise, state predictions are a crucial part of the Bayesian state estimation framework from the previous section. This section develops statistical models of uncertainties in these predictions.

A problem definition is first formulated in Sec. 4.2.1. Following an outline of related works in Sec. 4.2.2, the main findings are developed in six sections. These are organised as visualised in Fig. 4.7. First, commonly employed vehicle motion models are discussed in Sec. 4.2.3. Theoretical foundations of estimating the parameters of the statistical models are introduced in Sec. 4.2.4 and subsequently applied to three exemplary motion models in Sec. 4.2.5. Based on the parametrised models, an analytical propagation is derived for the uncertainty in the absolute motion of individual vehicles in Sec. 4.2.6 and the relative motion between them in Sec. 4.2.7.

The obtained models and analytical propagation are evaluated in Sec. 4.2.8. Especially the validity of the proposed noise model is critically studied by comparison to the distribution of prediction errors in a large-scale dataset of real-world trajectories. A summary in Sec. 4.2.9 concludes this section.

Major parts of the following content have been previously published in [218, 221].

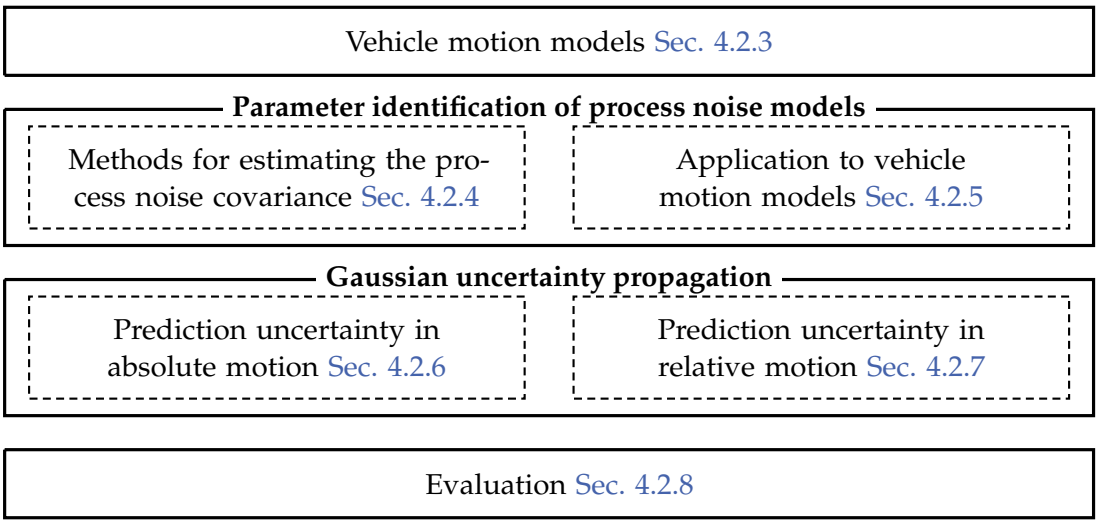


Figure 4.7 Outline of the main sections of Sec. 4.2.

4.2.1 Introduction

In the simplest form, trajectories are predicted individually in terms of kinematic motion quantities, for example position, velocities, and accelerations. These are written in the form of a state vector $\mathbf{x}(t) \in \mathbb{R}^n$. A prediction model that describes the time evolution of the state is defined by a system of differential equations:

$$\dot{\mathbf{x}}(t) = \mathbf{f}(\mathbf{x}(t)) \ , \qquad \mathbf{x}(t_k) = \mathbf{x}_k \ . \tag{4.42}$$

An initial value $\mathbf{x}_k$ of the dynamic state at an initial time t_k can be estimated from a history of measurements by means of a state estimator, as introduced in the previous Sec. 4.1. Then, the future trajectory is obtained by solving (4.42) with the initial value. In order to make a prediction solely based on the limited information of the initial state, the model includes certain assumptions on the future trajectories. Usually, some time derivatives of the state are zero, that is the corresponding kinematic quantity is assumed to remain at a constant value.

Due to an incomplete and uncertain perception of the environment, the predictions are in general affected by uncertainty. To assess the reliability of a prediction and facilitate its integration into a Bayesian framework, statistical models of the prediction uncertainty are required. Besides the obvious cause of noisy sensor measurements, the future behaviour of traffic participants is never certain. Thus, uncertainty is introduced by inaccuracies in the assumed model (4.42) of the time evolution. Two kinds of deviations can be differentiated [118]:

1. *Manoeuvre changes:* Abrupt changes in the driven manoeuvre, for example from straight driving to turning, can occur. These may be expressed as an unknown, deterministic input signal $\mathbf{u}(t)$ in $\dot{\mathbf{x}}(t) = \mathbf{f}(\mathbf{x}(t), \mathbf{u}(t))$.

   As these changes are not included in an initial motion state $\mathbf{x}(t_k)$ they can only be inferred on a higher level of abstraction than the kinematic quantities, for instance from context information or driver intention estimation.

2. *Stochastic disturbances:* Small perturbations occur during the same driven manoeuvre. For example, while driving along a straight road the vehicle's velocity can vary due to slopes. These deviations are modelled as a stochastic noise process $\mathbf{w}(t)$ which acts additively[6] on the system dynamics through an input matrix $\mathbf{L}$:

$$\dot{\mathbf{x}}(t) = \mathbf{f}(\mathbf{x}(t)) + \mathbf{L}\mathbf{w}(t) \ . \tag{4.43}$$

This section focusses on the second case of stochastic disturbances in kinematic motion models. A probabilistic model of these deviations will be employed and the model parameters will be estimated from recorded empirical data. Thereby, it is assumed here that no manoeuvre changes occur during the considered prediction horizon. Models with multiple modes that describe individual manoeuvres will be studied in Sec. 5.1.

Valuable insight can be obtained from models that allow closed-form solutions of the predicted state distribution $p\big(\mathbf{x}(t)\big)$. To this end, $\mathbf{w}(t)$ is assumed as a zero mean Gaussian white noise process, parametrised by a constant power spectral density $\mathbf{S}$:

$$\mathbb{E}\left[\mathbf{w}(t)\,\mathbf{w}^{\mathrm{T}}(\tau)\right] = \mathbf{S}\delta(t-\tau) \ . \tag{4.44}$$

[6] An additive influence of the process noise is not a principal requirement for the derivations in this section. Since this is usually the case for the physical motion models considered here and allows for easier notation though, this assumption is made.

The first goal, which will be approached in Sec. 4.2.4-4.2.5, is therefore to estimate $\mathbf{S}$ from recorded measurement data. Secondly, the distribution of the predicted state will be derived in Sec. 4.2.6-4.2.7.

4.2.2 Related work

Prediction models have two main applications in the driver assistance context, namely in object tracking and situation interpretation. Hence, related works on *parameter estimation* and *error analysis* for these models are reported in both areas. The difference is the time-span over which a prediction is made, that is either a fraction of a second for the system sampling time or multiple seconds for situation analysis.

For the purpose of object tracking, a number of motion models with white Gaussian process noise are known [124]. In order to make a sensible choice, the overall accuracy of the tracking filter results can be evaluated. This is done either in simulations [202] or based on real-world trajectories [175].

Concerning the parametrisation of the process noise model, two approaches can be differentiated. Firstly, the covariance can be chosen as an upper expected deviation between the model and a true trajectory. When modelling vehicle motion, bounds can be derived from the maximum acceleration capabilities [101, 103]. In a tracking algorithm, such a conservative choice minimises the risk of a track loss. For long-term predictions though, large and barely conclusive covariance predictions are a consequence because any physically possible trajectory is enclosed.

Secondly, the noise parameters can be estimated from recorded data. To this end, the *expectation maximisation (EM)* [45] will be employed in this work. This method has been applied for system identification in many fields, for example in the robotics domain [7, 126]. Additionally, it has been proposed to use the method for an on-line adaptation of a tracking filter in [20, 119]. However, a comprehensive account on estimating the process noise parameters of vehicle motion models from recorded trajectories has to the best of the author's knowledge not been provided before.

In the field of situation assessment, a plethora of works exist on the design, parameter inference and evaluation of models for long-term predictions [118]. For example, a sophisticated Markovian model, which incorporates multiple semantic aspects such as intentions and interactions, is proposed in [180]. Furthermore, a number of works, for example [129], deal with model identification of the car-following behaviour using naturalistic driving data. However, these models concern the interactions of a driver with a preceding vehicle and are used for instance in microscopic traffic simulations. For a driver assistance system, this information is usually unknown as vehicles in front of the observed one are occluded.

The focus of this work is on uncertainty propagation in driver assistance functions where simple, purely kinematic models are implicitly assumed. For example, algorithms for risk assessment will be studied in Sec. 5.2 and these are often based on the constant acceleration (CA) model [95]. The following section will provide an overview on commonly employed vehicle motion models.

4.2.3 Kinematic vehicle motion models

The scope of this section is predicting the two-dimensional motion of vehicles on the road. To make predictions for other road users, for example cyclists or pedestrians, other kinds of models are required which take into account the higher variability of their motion [189].

First, continuous-time vehicle motion models are described in Sec. 4.2.3.1. Second, the transformation to equivalent discrete-time models is explained in Sec. 4.2.3.2.

4.2.3.1 Continuous-time vehicle motion models

Following the classification in [118], the vehicle motion models considered here are physics-based. This stands for the most simple family of models where the future trajectory is assumed to be governed solely by physical laws. Higher levels of abstraction, for example the intentions and goals of a driver, are not included. Thus, the models' validity is limited to short prediction horizons of a few seconds, appropriate for example for collision avoidance functions.

Within this class, one can differentiate between dynamic and kinematic models [118]. Dynamic models consider the forces that act on a vehicle, for example the tire-road contact, and describe the resulting trajectory. While this view is viable for vehicle dynamics control, these models are usually too complicated to be useful when a vehicle is only observed with exteroceptive sensors. In this case, kinematic models are more suitable. They describe a vehicle's motion only in terms of kinematic quantities but without taking the internal forces and torques into account.

Kinematic models can be furthermore differentiated by their level of complexity:

- In terms of the order of time differentials which are comprised in the state vector, for instance velocity, acceleration, jerk etc. Considering higher derivatives leads to more unknown state variables. At the cost of increased state estimation effort, higher fidelity is possible.

- In terms of the dynamics: Models either assume purely translational motion or take an additional rotation into account (curvilinear models). The advantage of the former is their linearity, at the cost of realism. Hence, closed-form expressions for the exact estimation and prediction of Gaussian states can be employed.

Given the possible combinations of these aspects, a number of different models is commonly used in ADAS applications as seen in Fig. 4.8 [175]. The most general model considered in this work is the constant turn rate and acceleration model (CTRA). Other popular models are thereby included as special cases: If either the acceleration or the yaw rate is assumed as zero, the constant acceleration (CA) or constant turn rate (CT) models are obtained. In the case that both quantities are zero, one has the constant velocity (CV) model.

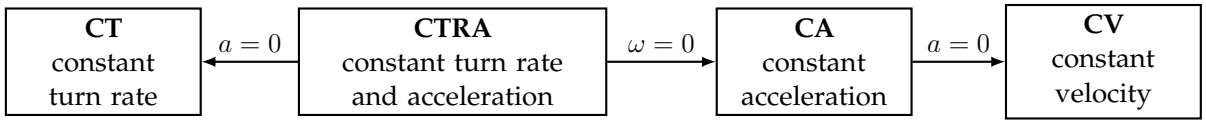


Figure 4.8 Relations between commonly used kinematic motion models.

The CTRA and CV models will be exemplarily studied throughout this chapter. Most of the derivations are however performed for general non-linear systems (4.43) and are thereby not limited to vehicle motion models.

4.2.3.2 Discrete-time vehicle motion models

The eventual goal is to estimate the power spectral density $\mathbf{S}$ of the process noise $\mathbf{w}(t)$ in the continuous-time models from measured data. However, this measurement data is obtained by sampling at discrete points in time. Thus, the discrete-time counterparts of the models have to be introduced.

We assume a constant sampling time T_S and denote the sampled time as $t_k = k \cdot T_\mathrm{S}$ and $\mathbf{x}_k := \mathbf{x}(t_k)$. The discretised model which corresponds to (4.43) is given as [63][7]

$$\mathbf{x}_{k+1} = \mathbf{f}(\mathbf{x}_k) + \mathbf{w}_k \ , \qquad \mathbf{w}_k \sim \mathcal{N}(\mathbf{0}, \mathbf{Q}_k) \ , \tag{4.45a}$$

where

$$\mathbf{Q}_k = \int_{t_k}^{t_{k+1}} \mathbf{\Phi}(t_{k+1}, \tau)\, \mathbf{L}\mathbf{S}\mathbf{L}^\mathrm{T} \mathbf{\Phi}^\mathrm{T}(t_{k+1}, \tau)\ \mathrm{d}\tau \ , \tag{4.45b}$$

$$\mathbf{\Phi}(t, t_0) = \exp\left(\int_{t_0}^{t} \mathbf{F}(\tau)\ \mathrm{d}\tau \right) \ , \qquad \mathbf{F}(t) = \nabla_{\mathbf{x}(t)} \mathbf{f}(\mathbf{x}(t)) \ . \tag{4.45c}$$

Here, $\mathbf{f}(\mathbf{x}_k)$ denotes the discrete time state transition function. It can be found by solving the differential equation (4.42) of the system in continuous time.

In general, the covariance of the process noise $\mathbf{Q}_k$ depends on the sampling time T_S, the spectral density $\mathbf{S}$ and the state $\mathbf{x}_k$ and is therefore time-variant. These dependencies will, however, complicate the estimation of $\mathbf{S}$ and thus, the use of alternative discretisation methods can be a viable option. Different approaches are discussed in [70]. One method is to assume a discrete-time noise input $\mathbf{w}_k \sim \mathcal{N}(\mathbf{0}, \mathbf{Q})$ with $\mathbf{Q} = {T_\mathrm{S}}^{-1}\mathbf{S}$ and modulate its influence on the system dynamics with a state-dependent input matrix $\mathbf{G}(\mathbf{x}_k)$:

$$\mathbf{x}_{k+1} = \mathbf{f}(\mathbf{x}_k) + \mathbf{G}(\mathbf{x}_k)\, \mathbf{w}_k \ , \qquad \mathbf{G}(\mathbf{x}_k) = \int_{t_k}^{t_{k+1}} \exp(\mathbf{F}(t_k)\, \tau)\, \mathbf{L}\ \mathrm{d}\tau \ . \tag{4.46a}$$

In the special case of linear dynamics $\dot{\mathbf{x}}(t) = \mathbf{F}\mathbf{x}(t)$ the transition matrix $\mathbf{\Phi}(t, t_0)$ only depends on the time difference $t - t_0$:

$$\mathbf{\Phi}(t, t_0) = \exp(\mathbf{F}(t - t_0)) \ . \tag{4.47}$$

[7] A rigorous analysis of the discretisation of continuous-time stochastic differential equations by means of Itô calculus is presented in [133]. However, to make the presentation easier accessible, this degree of formalism is not introduced in this work.

Thus, $\mathbf{Q}$ from (4.45b) is time-invariant and does not depend on the state $\mathbf{x}(t)$.

Note that $\mathbf{Q}$ is intentionally derived by starting from a continuous-time noise process. Alternatively, discrete-time noise inputs can be directly modelled under the assumption of a constant amplitude and covariance between two sampling time instants [70, 124, 161]. This is a viable approach if solely the discrete-time system is considered, for example for the design of a tracking filter. However, it is not applicable to our case since the link to the continuous-time model (4.43) is lost.

Example 4.1 (Constant turn rate and acceleration model)

A curvilinear trajectory is modelled by taking into account rotational and translational motion with constant acceleration a and yaw rate ω. In the state vector $\mathbf{x} = \begin{bmatrix} x & y & v & \theta & a & \omega \end{bmatrix}^{\mathrm{T}}$, velocity and acceleration are conveniently modelled in polar coordinates. Then, the differential equations in the form of (4.43) are given as:

$$\begin{bmatrix} \dot{x}(t) \\ \dot{y}(t) \\ \dot{v}(t) \\ \dot{\theta}(t) \\ \dot{a}(t) \\ \dot{\omega}(t) \end{bmatrix} = \begin{bmatrix} v(t)\cos(\theta(t)) \\ v(t)\sin(\theta(t)) \\ a(t) \\ \omega(t) \\ 0 \\ 0 \end{bmatrix} + \begin{bmatrix} 0 & 0 \\ 0 & 0 \\ 0 & 0 \\ 0 & 0 \\ 1 & 0 \\ 0 & 1 \end{bmatrix} \cdot \begin{bmatrix} w_a(t) \\ w_\omega(t) \end{bmatrix}, \qquad \mathbf{S} = \begin{bmatrix} S_a & 0 \\ 0 & S_\omega \end{bmatrix}. \tag{4.48}$$

Despite the model's non-linearity, closed-form expressions for the time evolution of the state and the process noise covariance matrix are available. The derivations and results are further detailed in appendix A.2.

Example 4.2 (Constant velocity model)

The model describes a purely translational, non-accelerated motion in Cartesian coordinates. This simplified view is useful when an object's motion is modelled without explicit knowledge of its orientation.

$$\begin{bmatrix} \dot{x}(t) \\ \dot{v}_x(t) \\ \dot{y}(t) \\ \dot{v}_y(t) \end{bmatrix} = \begin{bmatrix} 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 0 \end{bmatrix} \begin{bmatrix} x(t) \\ v_x(t) \\ y(t) \\ v_y(t) \end{bmatrix} + \begin{bmatrix} 0 & 0 \\ 1 & 0 \\ 0 & 0 \\ 0 & 1 \end{bmatrix} \cdot \begin{bmatrix} w_x(t) \\ w_y(t) \end{bmatrix}, \quad \mathbf{S} = \begin{bmatrix} S_x & 0 \\ 0 & S_y \end{bmatrix}. \tag{4.49}$$

Due to the model's linearity, a simple closed-form solution of the discrete-time state transition (4.45) can be obtained

$$\begin{bmatrix} x_{k+1} \\ v_{x,k+1} \\ y_{k+1} \\ v_{y,k+1} \end{bmatrix} = \begin{bmatrix} 1 & T_\mathrm{S} & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & T_\mathrm{S} \\ 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} x_k \\ v_{x,k} \\ y_k \\ v_{y,k} \end{bmatrix} + \mathbf{w}_k, \qquad \mathbf{w}_k \sim \mathcal{N}(\mathbf{0}, \mathbf{Q}) \tag{4.50a}$$

with the process noise covariance $\mathbf{Q}$ of the discretised model according to (4.45b):

$$\mathbf{Q} = \begin{bmatrix} \frac{1}{3}{T_\mathrm{S}}^3 S_x & \frac{1}{2}{T_\mathrm{S}}^2 S_x & 0 & 0 \\ \frac{1}{2}{T_\mathrm{S}}^2 S_x & T_\mathrm{S} S_x & 0 & 0 \\ 0 & 0 & \frac{1}{3}{T_\mathrm{S}}^3 S_y & \frac{1}{2}{T_\mathrm{S}}^2 S_y \\ 0 & 0 & \frac{1}{2}{T_\mathrm{S}}^2 S_y & T_\mathrm{S} S_y \end{bmatrix} . \tag{4.50b}$$

If the state vector of the CV model is extended with Cartesian accelerations, the constant acceleration (CA) model can be defined similarly to (4.49). Due to its similarity to the CV model, all further derivations can be performed in the same way and will not be discussed in detail. However, this model will be included in the evaluation in Sec. 4.2.8 in order to compare its accuracy with the simpler constant velocity one.

4.2.4 EM-based estimation of the process noise covariance

In this section, theoretical background on estimating the process noise parameters will be introduced. The estimator equations are derived for general non-linear systems with additive process noise in Sec. 4.2.4.1-4.2.4.3. A novel theoretical contribution is a closed-form adaptation of the algorithm to a process noise covariance matrix with a Kronecker product structure (Sec. 4.2.4.4). Such a factorisation can be found for example in the CV and CTRA motion models from examples 4.1-4.2.

4.2.4.1 Problem formulation and approach

The preliminary goal is to estimate the covariance $\mathbf{Q}$ which defines the process noise in the discretised system (4.45). Since the spectral density $\mathbf{S}$ of the noise process in the corresponding continuous-time system (4.43) is of main interest though, its relation to $\mathbf{Q}$ and the implication on the estimation are subsequently analysed.

Measurement data, which forms the basis for estimating $\mathbf{Q}$, usually comprise a partial and uncertain representation $\mathbf{y}_k \in \mathbb{R}^m$ of the state $\mathbf{x}_k$ only. This is expressed by a non-linear measurement function $\mathbf{h}(\cdot)$ and additive white Gaussian noise:

$$\mathbf{y}_k = \mathbf{h}(\mathbf{x}_k) + \mathbf{v}_k \,, \qquad \mathbf{v}_k \sim \mathcal{N}(\mathbf{0}, \mathbf{R}_k) \,. \tag{4.51}$$

It is assumed that $e = 1, \ldots, N_\mathrm{seq}$ independent sequences of measurements with $k = 1, \ldots, N$ samples each are available. These are denoted by $\mathcal{Y} \coloneqq \{\mathbf{y}_{1:N}^{(e)}\}_{e=1:N_\mathrm{seq}}$.[8]

One assumption is that one has full knowledge of the state transition function $\mathbf{f}(\cdot)$. For the kinematic motion models, this state transition is fully derived from physical considerations. However, when autoregressive processes with time-correlated states or

[8] For notational convenience, it is assumed that the measurement time-series are of equal length N. However, this is not a restriction and the method can be similarly derived for individual lengths N_e, $e = 1, \ldots, N_\mathrm{seq}$.

coloured process noise are to be modelled, additional degrees of freedom have to be included. In these cases, a parametric transition model for an augmented state vector which spans multiple time steps can be constructed. In addition to the process noise covariance, the parameters of the transition models have to be estimated. This can be performed with the EM principle as well [130].

The stochastic system (4.45) and the measurement model (4.51) allow us to formulate the probability density function $p\left(\mathcal{Y}\middle|\boldsymbol{\theta}\right)$ of a time series of measurements conditional on an unknown system parameter $\boldsymbol{\theta}$. In the case considered here, $\boldsymbol{\theta}$ defines the process noise covariance. For example, it may refer to the spectral density parameters S_x and S_y in the process noise covariance (4.50b) of the CV model.

According to the maximum likelihood principle, maximising the likelihood $p\left(\mathcal{Y}\middle|\boldsymbol{\theta}\right)$ yields the most likely estimate $\hat{\boldsymbol{\theta}}$ that explains the observations. For notational convenience, the log-likelihood $l_{\mathcal{Y}}\left(\boldsymbol{\theta}\right) \coloneqq \log p\left(\mathcal{Y}\middle|\boldsymbol{\theta}\right)$ is equivalently maximised:

$$\hat{\boldsymbol{\theta}} = \arg\max_{\boldsymbol{\theta}} l_{\mathcal{Y}}\left(\boldsymbol{\theta}\right) \ . \tag{4.52}$$

However, for a large number of data points, a maximisation in batch form becomes very difficult. An efficient, iterative approach is the EM-algorithm [45]. The central idea is to reformulate the log-likelihood $l_{\mathcal{Y}}\left(\boldsymbol{\theta}\right)$ in terms of the states $\mathcal{X} \coloneqq \left\{\mathbf{x}_{1:N}^{(e)}\right\}_{e=1:N_{\text{seq}}}$ as $l_{\mathcal{Y},\mathcal{X}}\left(\boldsymbol{\theta}\right) = \log p\left(\mathcal{Y},\mathcal{X}\middle|\boldsymbol{\theta}\right)$. Due to the assumption of Markovian processes, the conditional density will factorise and thus allow an efficient maximisation of the log-likelihood. However, the true states $\mathcal{X}$ are in general unknown and the complete log-likelihood is thus unavailable.

The key idea of the EM-principle is to estimate the distribution of $\mathcal{X}$ from the measurements $\mathcal{Y}$ and calculate the expected log-likelihood:

$$q_{\hat{\boldsymbol{\theta}}_l}\left(\boldsymbol{\theta}\right) = \mathbb{E}\left[\log p\left(\mathcal{Y},\mathcal{X}\middle|\boldsymbol{\theta}\right)\right] \ . \tag{4.53}$$

To find the distribution of $\mathcal{X}$, a fully parametrised system model and thus an estimate $\hat{\boldsymbol{\theta}}_l$ for $\boldsymbol{\theta}$ is required. A new estimate $\hat{\boldsymbol{\theta}}_{l+1}$ is then obtained by maximising the expected log-likelihood:

$$\hat{\boldsymbol{\theta}}_{l+1} = \arg\max_{\boldsymbol{\theta}} q_{\hat{\boldsymbol{\theta}}_l}\left(\boldsymbol{\theta}\right) \ . \tag{4.54}$$

Therefore, the EM-algorithm iteratively calculates the expectation in the E-Step (4.53) and subsequently maximises $q_{\hat{\boldsymbol{\theta}}_l}\left(\boldsymbol{\theta}\right)$ with respect to $\boldsymbol{\theta}$ in order to find a new estimate $\hat{\boldsymbol{\theta}}_{l+1}$ in the M-Step (4.54). It can be shown, using Jensen's inequality, that the sequence of estimates $\hat{\boldsymbol{\theta}}_l$ approaches the maximum likelihood estimate [45]. A possible convergence criterion for stopping the recursions after l^* steps is given by the difference in the expected log-likelihood. The values from two subsequent iterations are compared to a threshold value $\Delta q_{\min}$:

$$q_{\hat{\boldsymbol{\theta}}_l}\left(\hat{\boldsymbol{\theta}}_{l^*}\right) - q_{\hat{\boldsymbol{\theta}}_{l-1}}\left(\hat{\boldsymbol{\theta}}_l\right) \overset{!}{\leq} \Delta q_{\min} \ . \tag{4.55}$$

One remarkable property is that both steps of the algorithm can be efficiently performed for the (non-) linear Gaussian systems that are considered here. The algorithm's details will be laid out in the following, similar to the derivations found in [7, 64].

4.2.4.2 Expectation step

First, the joint log-likelihood $l_{\mathcal{Y},\mathcal{X}}(\boldsymbol{\theta})$ is simplified under the assumption that the $e = 1, \ldots, N_{\text{seq}}$ individual sequences of measurements are independent and that the noise sequences $\mathbf{w}_k$ and $\mathbf{v}_k$ are white and independent. Thus, the sequences $\mathbf{x}_{1:N}^{(e)}$ and $\mathbf{y}_{1:N}^{(e)}$ fulfil the one-step Markov property and one can rewrite the joint probability density using Bayes' rule (2.8) [7]:

$$\log p(\mathcal{Y}, \mathcal{X}) = \sum_{e=1}^{N_{\text{seq}}} \left[\log p\left(\mathbf{y}_1^{(e)}, \mathbf{x}_1^{(e)}\right) + \sum_{k=2}^{N} \left\{ \log p\left(\mathbf{y}_k^{(e)} \middle| \mathbf{x}_k^{(e)}\right) + \log p\left(\mathbf{x}_k^{(e)} \middle| \mathbf{x}_{k-1}^{(e)}\right) \right\} \right] . \tag{4.56}$$

Since process (4.45) and measurement (4.51) models with Gaussian noise are assumed, the conditional densities are Gaussians. Inserting them into (4.56) yields the expected log-likelihood:

$$q_{\hat{\mathbf{Q}}_l}(\mathbf{Q}) = c - \frac{1}{2} N_{\text{seq}} (N-1) \log\det(\mathbf{Q}) - \frac{1}{2} \operatorname{tr}\left(\mathbf{Q}^{-1}\mathbf{M}\right) \tag{4.57a}$$

where c is a constant that is independent of $\mathbf{Q}$ and $\mathbf{M}$ is an abbreviation for

$$\mathbf{M} := \sum_{e=1}^{N_{\text{seq}}} \sum_{k=2}^{N} \mathbf{M}_k^{(e)} , \qquad \mathbf{M}_k^{(e)} := \mathbb{E}\left[\left(\mathbf{x}_k^{(e)} - \mathbf{f}\left(\mathbf{x}_{k-1}^{(e)}\right) \right) \left(\mathbf{x}_k^{(e)} - \mathbf{f}\left(\mathbf{x}_{k-1}^{(e)}\right) \right)^{\mathrm{T}} \right] . \tag{4.57b}$$

The essential part of the E-step is now to separately calculate $\mathbf{M}_k^{(e)}$, $k = 2, \ldots, N$ for each sequence of observations $\mathbf{y}_{1:N}^{(e)}$ [7]. In a first step, it is thus necessary to obtain an estimate of the joint densities $p\left(\mathbf{x}_k^{(e)}, \mathbf{x}_{k-1}^{(e)}\right)$, $k = 2, \ldots, N$. Preserving Gaussianity, such an estimate is provided by the extended Kalman smoother (EKS) through a linearisation of the process and measurement models.

The smoother algorithm consists of the same prediction (4.14) and innovation step (4.15) as the extended Kalman filter from Sec. 4.1.3.4. At this stage, the estimate $\hat{\mathbf{Q}}_l$ of the process noise covariance is required to perform the prediction. In contrast to the filtering problem, measurements from all time steps $1, \ldots, k, \ldots, N$ can be employed to find an estimate of the state distribution at time k. These are included in an additional smoothing step. Based on the results from the prediction $\left(\hat{\mathbf{x}}_{k+1|k}^{(e)}, \boldsymbol{\Sigma}_{k+1|k}^{(e)}\right)$ and innovation step $\left(\hat{\mathbf{x}}_k^{(e)}, \boldsymbol{\Sigma}_k^{(e)}\right)$, the recursion of this backward pass reads for $k = N-1, \ldots, 1$ [166]:

$$\hat{\mathbf{x}}_{k|N}^{(e)} = \hat{\mathbf{x}}_k^{(e)} + \mathbf{J}_k^{(e)} \left(\hat{\mathbf{x}}_{k+1|N}^{(e)} - \hat{\mathbf{x}}_{k+1|k}^{(e)} \right) \tag{4.58a}$$

$$\boldsymbol{\Sigma}_{k|N}^{(e)} = \boldsymbol{\Sigma}_k^{(e)} + \mathbf{J}_k^{(e)} \left(\boldsymbol{\Sigma}_{k+1|N}^{(e)} - \boldsymbol{\Sigma}_{k+1|k}^{(e)} \right) \left(\mathbf{J}_k^{(e)} \right)^{\mathrm{T}} \tag{4.58b}$$

$$\boldsymbol{\Sigma}_{k+1,k|N}^{(e)} = \boldsymbol{\Sigma}_{k+1|N}^{(e)} \left(\mathbf{J}_k^{(e)}\right)^{\mathrm{T}} \tag{4.58c}$$

$$\text{with } \mathbf{J}_k^{(e)} = \boldsymbol{\Sigma}_k^{(e)} \left(\mathbf{A}_k^{(e)}\right)^{\mathrm{T}} \left(\boldsymbol{\Sigma}_{k+1|k}^{(e)}\right)^{-1}, \quad \mathbf{A}_k^{(e)} = \nabla_{\mathbf{x}_k} \mathbf{f}\left(\mathbf{x}_k\right)\big|_{\mathbf{x}_k = \hat{\mathbf{x}}_k^{(e)}} . \tag{4.58d}$$

The backward pass is initialised with the last estimate $\left(\hat{\mathbf{x}}_N^{(e)}, \boldsymbol{\Sigma}_N^{(e)}\right)$ of the forward pass.

Subsequently, the expectation in (4.57b) can be calculated with the obtained joint densities $p\left(\mathbf{x}_k^{(e)}, \mathbf{x}_{k-1}^{(e)}\right)$. To this end, the state transition function $\mathbf{f}\left(\mathbf{x}_k\right)$ is linearised around the smoothed state estimate [7]:[9]

$$\begin{aligned}
\mathbf{M}_k^{(e)} &\approx \mathbb{E}\left[\left(\mathbf{x}_{k-1}^{(e)} - \mathbf{f}\left(\hat{\mathbf{x}}_{k-1|N}^{(e)}\right) - \mathbf{A}_{k-1}^{(e)}\left(\mathbf{x}_{k-1}^{(e)} - \hat{\mathbf{x}}_{k-1|N}^{(e)}\right)\right)\left(\cdot\right)^{\mathrm{T}}\right] \\
&= \begin{bmatrix} -\mathbf{A}_{k-1}^{(e)} & \mathbf{I}_{n\times n} \end{bmatrix} \begin{bmatrix} \boldsymbol{\Sigma}_{k-1|N}^{(e)} & \boldsymbol{\Sigma}_{k,k-1|N}^{(e)\mathrm{T}} \\ \boldsymbol{\Sigma}_{k,k-1|N}^{(e)} & \boldsymbol{\Sigma}_{k|N}^{(e)} \end{bmatrix} \left(\cdot\right)^{\mathrm{T}} + \left(\hat{\mathbf{x}}_{k|N}^{(e)} - \mathbf{f}\left(\hat{\mathbf{x}}_{k-1|N}^{(e)}\right)\right)\left(\cdot\right)^{\mathrm{T}} \\
&\text{where} \quad \mathbf{A}_k^{(e)} = \nabla_{\mathbf{x}_k} \mathbf{f}\left(\mathbf{x}_k\right)\big|_{\mathbf{x}_k = \hat{\mathbf{x}}_{k|N}^{(e)}} .
\end{aligned} \tag{4.59}$$

Discretised systems according to (4.46), where the process noise is additionally modulated by an input matrix $\mathbf{G}\left(\mathbf{x}_k\right)$, are treated in [7]. The eventual result is an extension of (4.57b) that involves the Moore-Penrose inverse $\left(\cdot\right)^{\dagger}$:

$$\begin{aligned}
\mathbf{M} &= \sum_{e=1}^{N_{\mathrm{seq}}} \sum_{k=2}^{N} \left(\mathbf{G}\left(\hat{\mathbf{x}}_{k-1|N}^{(e)}\right)\right)^{\dagger} \mathbf{M}_k^{(e)} \left(\mathbf{G}^{\mathrm{T}}\left(\hat{\mathbf{x}}_{k-1|N}^{(e)}\right)\right)^{\dagger} \\
&\text{where} \quad \mathbf{G}^{\dagger} := \left(\mathbf{G}^{\mathrm{T}}\mathbf{G}\right)^{-1}\mathbf{G}^{\mathrm{T}} .
\end{aligned} \tag{4.60}$$

The subsequent steps of the algorithm do not rely on having an analytical expression for $\mathbf{M}$ and alternatively, numerical integration as in [172] can be used to calculate the expectation. However, since the calculation has to be repeated for each time step of each measurement sequence in each iteration, the computational burden quickly becomes very high.

4.2.4.3 Maximisation step

The objective is now to maximise the scalar expected log-likelihood $q_{\hat{\mathbf{Q}}_l}\left(\mathbf{Q}\right)$ from (4.57a) with respect to $\mathbf{Q}$. Employing the identities for matrix differentials from Sec. A.1, it follows that:

$$\frac{\partial}{\partial \mathbf{Q}} q_{\hat{\mathbf{Q}}_l}\left(\mathbf{Q}\right) = -\frac{1}{2} N_{\mathrm{seq}}\left(N-1\right)\mathbf{Q}^{-1} + \frac{1}{2}\mathbf{Q}^{-1}\mathbf{M}^{\mathrm{T}}\mathbf{Q}^{-1} . \tag{4.61}$$

Solving for $\mathbf{Q}$ so that all partial derivatives become zero, one obtains a closed-form result of the maximisation [7]:

$$\hat{\mathbf{Q}}_{l+1} = \frac{1}{N_{\mathrm{seq}}\left(N-1\right)}\mathbf{M}^{\mathrm{T}} . \tag{4.62}$$

[9] The shorthand notation $\mathbf{x}\mathbf{A}\left(\cdot\right)^{\mathrm{T}} := \mathbf{x}\mathbf{A}\mathbf{x}^{\mathrm{T}}$ is introduced for vectors $\mathbf{x}$ and matrices $\mathbf{A}$ of appropriate dimensions.

With this new covariance estimate, a further iteration of the expectation step can be started, that is the execution of the Kalman smoother and the subsequent expectation.

So far, the full covariance matrix $\mathbf{Q}$ has been estimated. A more general case is to assume a functional relationship $\mathbf{Q}(\boldsymbol{\theta})$ to an unknown, $n_{\boldsymbol{\theta}}$-dimensional parameter $\boldsymbol{\theta}$. While the expectation step remains identical, finding the estimate $\hat{\boldsymbol{\theta}}_{l+1}$ has a different solution. Instead of (4.61) the differentiation is now performed with respect to the elements of $\boldsymbol{\theta}$. Thereby, the chain rule can be applied [131]:

$$\frac{\partial}{\partial \theta_j} q_{\hat{\boldsymbol{\theta}}_l}(\boldsymbol{\theta}) = \operatorname{tr}\left(\left(\frac{\partial}{\partial \mathbf{Q}} q_{\hat{\mathbf{Q}}_l}(\mathbf{Q}) \right)^{\mathrm{T}} \frac{\partial \mathbf{Q}(\boldsymbol{\theta})}{\partial \theta_j} \right) , \qquad j = 1, \ldots, n_{\boldsymbol{\theta}} . \tag{4.63}$$

The previous result (4.61) appears within the $j = 1, \ldots, n_{\boldsymbol{\theta}}$ equations and it depends on the functional relation $\mathbf{Q}(\boldsymbol{\theta})$, how these can be solved for $\boldsymbol{\theta}$. One frequent case is that $\mathbf{Q}$ is defined by a Kronecker product between a known and unknown matrix. This will be further elaborated in the following.

4.2.4.4 Covariance matrix with Kronecker product structure

The process noise covariance of the motion models from Sec. 4.2.3 can be decomposed as a Kronecker product $\mathbf{Q}(\mathbf{S}) = \mathbf{S} \otimes \mathbf{Q}_1$. Here, $\mathbf{Q}_1 \in \mathbb{R}^{n_Q \times n_Q}$ is a known constant matrix and $\mathbf{S}$ is the unknown $n_S \times n_S$-dimensional spectral density. For example, the covariance of the discretised constant velocity model (4.50b) can be written as:

$$\mathbf{Q} = \begin{bmatrix} \frac{1}{3}T_{\mathrm{S}}{}^3 S_x & \frac{1}{2}T_{\mathrm{S}}{}^2 S_x & 0 & 0 \\ \frac{1}{2}T_{\mathrm{S}}{}^2 S_x & T_{\mathrm{S}} S_x & 0 & 0 \\ 0 & 0 & \frac{1}{3}T_{\mathrm{S}}{}^3 S_y & \frac{1}{2}T_{\mathrm{S}}{}^2 S_y \\ 0 & 0 & \frac{1}{2}T_{\mathrm{S}}{}^2 S_y & T_{\mathrm{S}} S_y \end{bmatrix} = \underbrace{\begin{bmatrix} S_x & 0 \\ 0 & S_y \end{bmatrix}}_{=\mathbf{S}} \otimes \underbrace{\begin{bmatrix} \frac{1}{3}T_{\mathrm{S}}{}^3 & \frac{1}{2}T_{\mathrm{S}}{}^2 \\ \frac{1}{2}T_{\mathrm{S}}{}^2 & T_{\mathrm{S}} \end{bmatrix}}_{=:\mathbf{Q}_1} . \tag{4.64}$$

Exploiting that $\mathbf{Q}$ is partly known in order to solely estimate the lower-dimensional $\mathbf{S}$ has to the best of the author's knowledge not been exemplified for the EM-algorithm. Related derivations exist though for covariance estimation of independent and identically distributed random samples, for example in [128, 198]. Independently distributed states are a special case of the Markovian processes which are studied here.

It will be shown in the following, how the decomposition (4.64) can be incorporated into the maximisation step from the previous Sec. 4.2.4.3. Instead of individually calculating the partial derivatives (4.63) and solving for the entries of $\mathbf{S}$, a simple closed-form solution similar to (4.62) will be derived with the help of matrix differentials.

First, (4.64) is inserted into (4.57a) and the expected log-likelihood $q_{\hat{\mathbf{S}}_l}(\mathbf{S})$ is obtained:

$$q_{\hat{\mathbf{S}}_l}(\mathbf{S}) = -\frac{1}{2} N_{\mathrm{seq}} (N-1) \log\det(\mathbf{S} \otimes \mathbf{Q}_1) - \frac{1}{2} \operatorname{tr}\left((\mathbf{S} \otimes \mathbf{Q}_1)^{-1} \mathbf{M} \right) . \tag{4.65}$$

In order to find the derivative with respect to $\mathbf{S}$, a reformulation is introduced. This is done using an abbreviation for the inverse of $\mathbf{Q}_1$, namely $\tilde{\mathbf{Q}} := \mathbf{Q}_1^{-1}$. Moreover, an $n_S \times n_S$ matrix $\tilde{\mathbf{M}}^{(j,i)}$ is introduced whose elements are taken from $\mathbf{M}$ at a regular pattern:

$$\tilde{\mathbf{M}}^{(j,i)} = \left[M_{j+(u-1)n_Q,\, i+(v-1)n_Q} \right]_{\substack{u=1,\ldots,n_S \\ v=1,\ldots,n_S}} . \tag{4.66}$$

As is shown in the appendix in Sec. A.3, it holds that

$$\operatorname{tr}\left(\left(\mathbf{S}\otimes\mathbf{Q}_1\right)^{-1}\mathbf{M}\right)=\sum_{i=1}^{n_Q}\sum_{j=1}^{n_Q}\tilde{Q}_{ij}\operatorname{tr}\left(\mathbf{S}^{-1}\tilde{\mathbf{M}}^{(j,i)}\right)\ . \tag{4.67}$$

Second, (4.67) is inserted into (4.65) and the derivative is obtained with the identities from Sec. A.1:

$$\frac{\partial}{\partial\mathbf{S}}q_{\hat{\mathbf{s}}_l}\left(\mathbf{S}\right)=-\frac{1}{2}N_{\text{seq}}\left(N-1\right)n_Q\mathbf{S}^{-1}-\frac{1}{2}\mathbf{S}^{-1}\left(\sum_{i=1}^{n_Q}\sum_{j=1}^{n_Q}\tilde{Q}_{ij}\tilde{\mathbf{M}}^{(j,i)}\right)^{\mathrm{T}}\mathbf{S}^{-1}\ . \tag{4.68}$$

This derivative has a similar form as (4.61) and thus, the desired estimate $\hat{\mathbf{S}}_{l+1}$ at which the derivatives vanish, is readily available:

$$\hat{\mathbf{S}}_{l+1}=\frac{1}{N_{\text{seq}}\left(N-1\right)n_Q}\left(\sum_{i=1}^{n_Q}\sum_{j=1}^{n_Q}\tilde{Q}_{ij}\tilde{\mathbf{M}}^{(j,i)}\right)^{\mathrm{T}}\ . \tag{4.69}$$

In summary, if the process noise covariance can be parametrised as a Kronecker product with one known factor, a simple closed-form solution is available. This result (4.69) has a similar form as the estimate of the unstructured covariance in (4.62). The central difference is that one has to employ the sum over the matrices $\tilde{\mathbf{M}}^{(j,i)}$ from (4.66), weighted with the known entries of $\mathbf{Q}_1^{-1}$, instead of $\mathbf{M}$.

4.2.5 Application to kinematic vehicle motion models

Having introduced the theoretical background, the estimation is carried out on vehicle trajectory data measured by a range sensing laser scanner. Applying the EM-algorithm to large datasets may become computationally and numerically challenging. These aspects are firstly discussed in Sec. 4.2.5.1 and advice for practitioners on commonly experienced issues is compiled. Details on the employed dataset and pre-processing steps are subsequently explained in Sec. 4.2.5.2. The estimation results are then summarised in Sec. 4.2.5.3.

4.2.5.1 Implementation of the EM-algorithm

Two aspects relevant for the implementation of the EM-algorithm will be briefly discussed. Firstly, each iteration of the E-Step includes a Kalman smoother run and hence computationally expensive matrix inversions. It is proposed in [130] to parallelise the E-Step and thus calculate $\mathbf{M}_k^{(e)}$ in (4.57b) independently for each sequence e. Another suggestion leverages that for linear, time-invariant systems, the gain and covariance matrices converge to stationary values. These can therefore be pre-computed by numerically solving an algebraic Riccati equation [130].

An alternative approach to increase the computational speed is to choose a different update rule in the M-Step with a better rate of convergence and thus reduce the number of iterations. For example, a Newton-type scheme can be applied where the Hessian of the expected log-likelihood is used to direct the next estimate [67].

Secondly, numerical robustness is concerned. Due to the iterative nature of the algorithm, round-off errors may accumulate. This can lead to negative definite estimates of a covariance matrix and divergence of the algorithm [64]. The numerical properties may be improved as follows:

- *E-Step:* Robust implementations of the Kalman smoother equations such as the square root form [64] can be used. Thus, it is ensured that the estimated $\boldsymbol{\Sigma}_{k|N}^{(e)}$ are always positive definite.
- *M-Step:* The estimate of $\mathbf{Q}$ in (4.62) is based on the matrix $\mathbf{M}$. For notational convenience, the calculation of the individual $\mathbf{M}_k^{(e)}$ in (4.59) is often explicitly written as additions and subtractions of matrices, for instance in [126, 130]. In order to avoid a potential loss of positivity due to round-off errors, the work [64] proposes an implementation based on Cholesky factorisations instead. Denote

$$\tilde{\boldsymbol{\Sigma}}\tilde{\boldsymbol{\Sigma}}^{\mathrm{T}} = \mathrm{chol}\left(\begin{bmatrix} \boldsymbol{\Sigma}_{k-1|N}^{(e)} & \boldsymbol{\Sigma}_{k,k-1|N}^{(e)\mathrm{T}} \\ \boldsymbol{\Sigma}_{k,k-1|N}^{(e)} & \boldsymbol{\Sigma}_{k|N}^{(e)} \end{bmatrix}\right), \qquad \boldsymbol{\Gamma} = \begin{bmatrix} -\mathbf{A}_{k-1}^{(e)} & \mathbf{I}_{n\times n} \end{bmatrix} \tilde{\boldsymbol{\Sigma}}^{\mathrm{T}} \tag{4.70}$$

so that

$$\mathbf{M}_k^{(e)} = \boldsymbol{\Gamma}\boldsymbol{\Gamma}^{\mathrm{T}} + \left(\hat{\mathbf{x}}_{k|N}^{(e)} - \mathbf{f}\left(\hat{\mathbf{x}}_{k-1|N}^{(e)}\right)\right)\left(\hat{\mathbf{x}}_{k|N}^{(e)} - \mathbf{f}\left(\hat{\mathbf{x}}_{k-1|N}^{(e)}\right)\right)^{\mathrm{T}} \tag{4.71}$$

is always positive definite.

4.2.5.2 Dataset and pre-processing

For the professed goal to determine the deviation between predictive models and recorded trajectories, two approaches to acquire the necessary data can be followed.

The first is to equip the ego-vehicle with recording devices that capture the required CAN-signals. Advantages of this method are the low efforts concerning measuring instruments and the generation of precise data. The drawback is a lack of diversity in vehicles and driving styles as only one car and driver is recorded at a time.

Therefore, a second approach was favoured. Both the CAN-data of the ego-vehicle and the positions of objects in the vehicle's environment are measured. Hence, variations caused by different driving styles and vehicles are included in the dataset, which comprises $\approx 170\,\mathrm{h}$ of raw recordings.[10] For the detection of the surrounding objects, the ego-vehicle was additionally provided with a laser scanner. The 2-D Cartesian position $\mathbf{y}_k$ of an object that is measured by this sensor can be described by a linear mapping

[10] The focus of later analyses in this work is on emergency brake systems based on a stereo vision sensor for accident prevention in urban scenarios. Therefore, only those parts of the raw recordings where the ego-vehicle maintains a speed of less than $70\,\mathrm{km/h}$ are considered in the following.

$\mathbf{h}(\mathbf{x}_k) = \mathbf{C}\mathbf{x}_k$ of the motion state $\mathbf{x}_k$. However, several preprocessing steps have to be performed on the exteroceptive measurements, which were developed in the supervised bachelor thesis [225] and are explained in the following paragraphs.

Filtering Detections which do not represent road users, like traffic signs or pedestrians, have to be filtered out of the measured objects. Employing the results of a classifier built-in the laser scanner, all recordings apart from cars, trucks and motorcycles are sorted out.

Transformation to ground-fixed coordinates Using the recorded CAN-data for velocity, acceleration and yaw rate, the trajectories of the ego-vehicle in a ground-fixed coordinate system can be calculated. Based on these, the trajectories of the remaining objects can be determined from the relative ranges between host and detected vehicles, which are measured by the laser scanner.

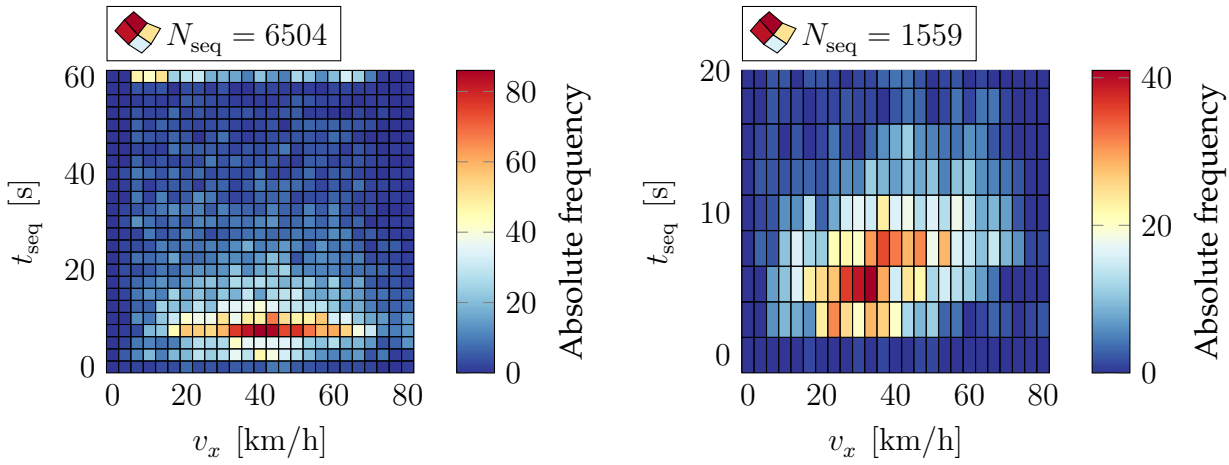
Classification of road user types In the next processing step, vehicles are categorised into those driving on the same lane, in the same direction and parked or oncoming vehicles. For the sake of data precision, only preceding vehicles on the same lane are considered in this work. The motivation is that the subsequent manoeuvre classification task can be based on the ego-vehicle CAN-data, because both the preceding vehicle and the following ego-vehicle drive along the same route but at different times.

Classification of manoeuvres For the purpose of evaluating the stochastic variations which occur during the same driven manoeuvre, the recorded trajectories are analysed for certain frequently occurring manoeuvres (straight driving, curves or turning). Afterwards, the recordings are split at the transitions between the manoeuvres. The reason for this is that manoeuvre changes severely violate the kinematic models' underlying assumptions. Hence, without further information about the driver's intention, any of the considered models will fail to correctly predict a future change which is not indicated in the current state.

Distribution into sequences The trajectories, which do now represent one manoeuvre type throughout, are grouped in a final processing step. Thus, three datasets which contain episodes of straight driving, curves and turns are obtained. Fig. 4.9 visualises the duration and velocity distribution of the straight driving and curved trajectory recordings.

4.2.5.3 Estimation results

At first, the two datasets for straight driving and curved trajectories are separated into a randomly chosen part (50%) for parameter estimation as well as the remaining 50% for evaluation purposes.



(a) Straight driving episodes. **(b)** Curves.

Figure 4.9 Visualisation of the mean absolute velocity and duration of detected manoeuvres in the datasets.

The trajectories in these disjoint sets have a non-uniform length, as visualised in Fig. 4.9. In a next step, they are cut in sequences of uniform length. The objective is to limit the influence of deterministic driver inputs, for example accelerations, to the short prediction time horizon that is to be captured by the probabilistic model. Thus, a sequence duration of 5 s is chosen for straight trajectories and 4 s for curves for the parameter estimation.[11]

Overall, $N_{\text{est,straight}} = 9576$ and $N_{\text{est,curves}} = 1205$ sequences are available for estimating the process noise parameters. This is performed for three different motion models, that are the CV and CA as well as the CTRA model from Sec. 4.2.3. For the first two, the covariance matrix has a Kronecker product form and the estimation is performed as proposed in Sec. 4.2.4.4. The discretisation of the non-linear CTRA model is detailed in Sec. A.2.

The EM algorithm is parametrised with randomly[12] chosen initial covariances $\hat{\mathbf{S}}_0$ and a threshold of $\Delta q_{\min} = 1 \times 10^{-7}$ on the likelihood difference between two iterations. A maximum number of 1×10^3 iterations is employed as an additional stop criterion.

In order to analyse the algorithm's convergence behaviour with respect to the initial values and how well the estimates generalise, the estimation is carried out not once but multiple times for different portions of the dataset. For each run, a portion of 20% is sampled from all sequences in the estimation dataset. This corresponds to approximately 17 h of raw driving data. The obtained estimates are visualised in Fig. 4.10 and Fig. 4.11 over the number of iterations. Moreover, the median values over all runs per model are given in Tab. 4.2(a).

[11] Given that a prediction horizon of up to 3 s is analysed, this would be a natural choice for the sequence length in the estimation as well. However, slightly higher values are chosen here in order to provide a sufficient amount of data to the estimator.

[12] Distributed uniformly within a range of $0 \ldots 200\%$ of a tentative estimate.

Firstly, despite the different datasets and initial values, the estimated process noise parameters converge to similar final values in most cases. This indicates that the chosen size of the data samples, that is 20% of the entire estimation dataset, is sufficiently high. Furthermore, the independence from the initial value choice suggests a robust convergence behaviour of the algorithm.

Secondly, different results for the straight and curved trajectory samples can be observed. Unsurprisingly, the process noise power S_y in lateral direction for the CV and CA models in curves takes on higher values than the longitudinal value S_x. Since these models only consider translational motion, the predictions are affected by high uncertainty when the vehicle is in fact performing a combined translational and rotational movement. The yaw rate component of the process noise S_ω for the CTRA model shows increased values compared to the case of straight trajectories as well. This suggests a higher variability of the steering input than in straight driving. A further potential cause is that the model assumption of a constant yaw rate is systematically violated for predictions that extend beyond a curve with finite length.

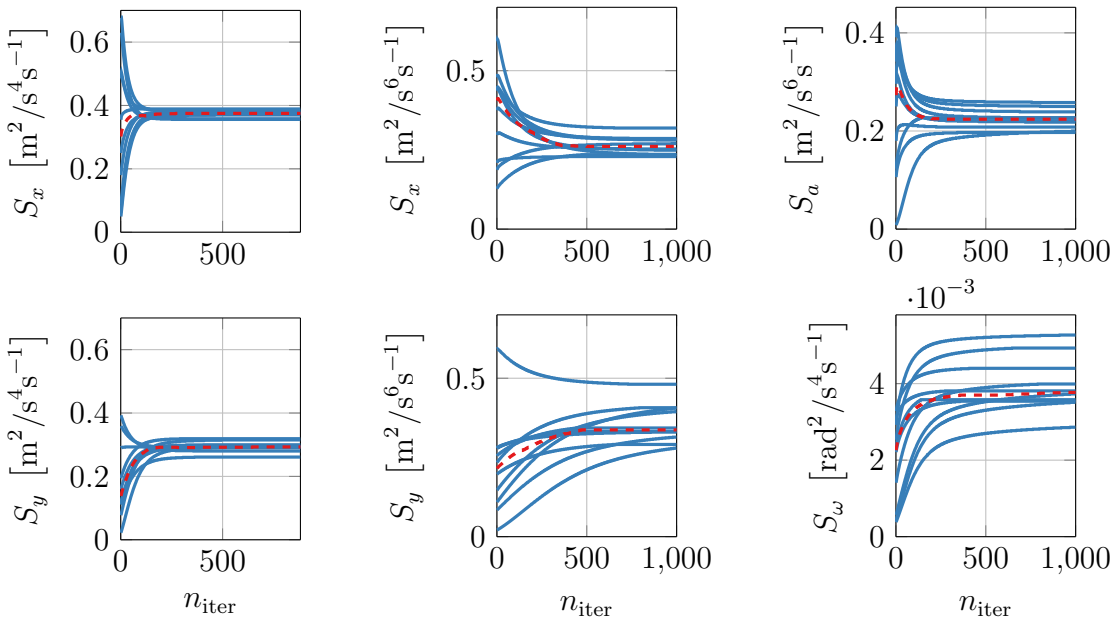
Thirdly, it can be noticed that for straight trajectories, both CA and CTRA model feature similar values for the longitudinal noise component S_x or S_a, respectively. This is expected since for a purely translational motion, the CTRA model reduces to the CA model. Thereby, the different variants of the EM algorithm which are employed for the two models lead to consistent results.

In conclusion, initial confidence in the estimation results is gained from their coherence and consistency to expectations. Nonetheless, a comprehensive evaluation on an independent validation dataset is necessary in order to draw meaningful conclusions. To this end, the second part of the dataset, which has not been used in the parameter estimation, is employed. Trajectory predictions are calculated with the parametrised models and compared to the true future course which will be detailed in Sec. 4.2.8.2. Previously though, theoretical foundations for obtaining the probabilistic trajectory predictions will be introduced in the following Sec. 4.2.6.

4.2.6 Gaussian prediction uncertainty in absolute motion

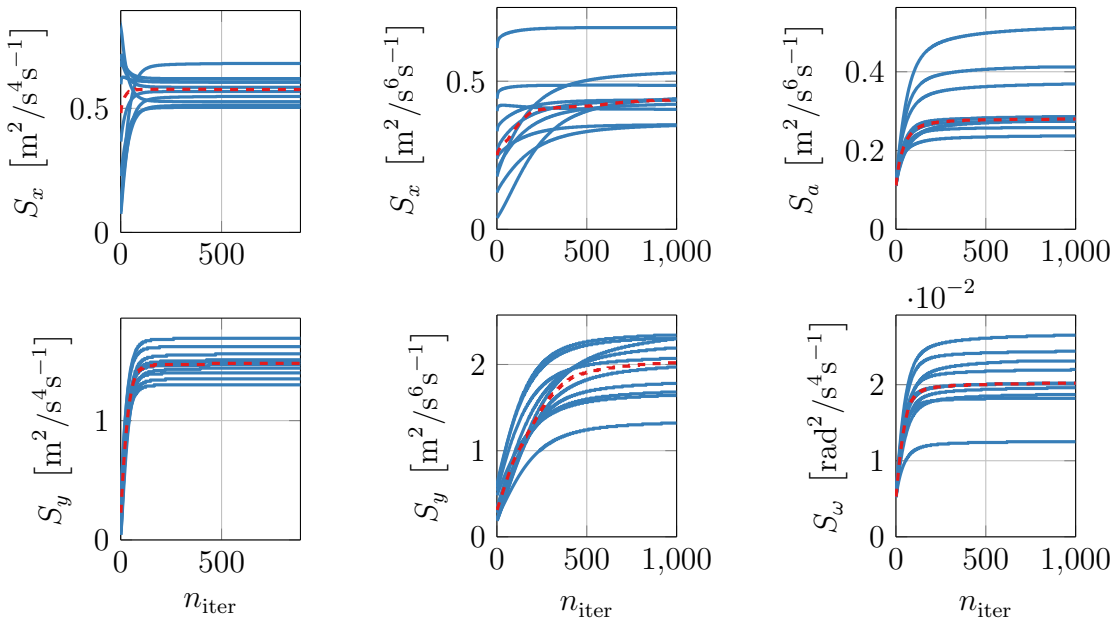
It is now assumed that a parametrised model of the form (4.43) with a known process noise power spectral density $\mathbf{S}$ is available. Given an initial state estimate $\mathbf{x}\left(t_k\right) \sim \mathcal{N}(\hat{\mathbf{x}}\left(t_k\right), \boldsymbol{\Sigma}_{\mathbf{x}}\left(t_k\right))$, the goal is to obtain a Gaussian model $\mathcal{N}(\hat{\mathbf{x}}\left(t_k+T\right), \boldsymbol{\Sigma}_{\mathbf{x}}\left(t_k+T\right))$ of the state prediction $\mathbf{x}\left(t_k+T\right)$ at a future time instance t_k+T. To this end, the foundations of uncertainty propagation in non-linear and linear dynamic systems are reviewed.

First of all, uncertainty propagation can be performed directly for the continuous-time model or iteratively for a discretised variant from Sec. 4.2.3.2 [90, 114]. The second method is identical to the prediction step (4.4) of a Bayes filter and its formalism possibly more familiar. However, this can introduce discretisation errors if the prediction horizon is not aligned with the chosen sample time. To circumvent such potential issues, this section concentrates on the treatment in continuous time.



(a) CV model **(b)** CA model **(c)** CTRA model

Figure 4.10 **Straight trajectories:** Estimation results of the process noise power spectral density for (a) the CV model, (b) the CA model and (c) the CTRA model. The graphs show the convergence of the iterative EM algorithm for 10 randomly drawn subsamples of the entire training data (solid) and the median estimates (dashed)



(a) CV model **(b)** CA model **(c)** CTRA model

Figure 4.11 **Curved trajectories:** Estimation results of the process noise power spectral density for (a) the CV model, (b) the CA model and (c) the CTRA model. The graphs show the convergence of the iterative EM algorithm for 10 randomly drawn subsamples of the entire training data (solid) and the median estimates (dashed).

Table 4.2 Estimated process noise power spectral density.

(a) Straight trajectories.

Model	Parameter estimates
CV	$S_x = 0.375\,\mathrm{m}^2/\mathrm{s}^4\mathrm{s}^{-1}$ $S_y = 0.293\,\mathrm{m}^2/\mathrm{s}^4\mathrm{s}^{-1}$
CA	$S_x = 0.261\,\mathrm{m}^2/\mathrm{s}^6\mathrm{s}^{-1}$ $S_y = 0.337\,\mathrm{m}^2/\mathrm{s}^6\mathrm{s}^{-1}$
CTRA	$S_a = 0.224\,\mathrm{m}^2/\mathrm{s}^6\mathrm{s}^{-1}$ $S_\omega = 0.0038\,\mathrm{rad}^2/\mathrm{s}^4\mathrm{s}^{-1}$

(b) Curves.

Model	Parameter estimates
CV	$S_x = 0.576\,\mathrm{m}^2/\mathrm{s}^4\mathrm{s}^{-1}$ $S_y = 1.476\,\mathrm{m}^2/\mathrm{s}^4\mathrm{s}^{-1}$
CA	$S_x = 0.437\,\mathrm{m}^2/\mathrm{s}^6\mathrm{s}^{-1}$ $S_y = 2.022\,\mathrm{m}^2/\mathrm{s}^3\mathrm{s}^{-1}$
CTRA	$S_a = 0.280\,\mathrm{m}^2/\mathrm{s}^6\mathrm{s}^{-1}$ $S_\omega = 0.020\,\mathrm{rad}^2/\mathrm{s}^4\mathrm{s}^{-1}$

For linear Gaussian models, an exact propagation in closed form is possible [63]. However, the state prediction density in motion models with non-linear dynamics, for example those including rotations, is non-Gaussian. As has been motivated in Sec. 2.2, a linearisation-based approximation [107] will be used due to consistency of notation. More accurate predictions can be achieved with non-Gaussian models, for example Gaussian mixture distributions [71].

The linearisation-based covariance propagation is given by [63]:

$$\mathbf{\Sigma}_\mathbf{x}\left(t_k + T\right) = \mathbf{\Phi}\left(t_k + T, t_k\right) \mathbf{\Sigma}_\mathbf{x}\left(t_k\right) \mathbf{\Phi}^\mathrm{T}\left(t_k + T, t_k\right) + \mathbf{Q}\left(t_k + T, t_k\right), \tag{4.72a}$$

where

$$\mathbf{Q}\left(t, t_0\right) = \int_{t_0}^{t} \mathbf{\Phi}\left(t, \tau\right) \mathbf{L}\mathbf{S}\mathbf{L}^\mathrm{T} \mathbf{\Phi}^\mathrm{T}\left(t, \tau\right) \,\mathrm{d}\tau, \tag{4.72b}$$

$$\mathbf{\Phi}\left(t, t_0\right) = \exp\left(\int_{t_0}^{t} \mathbf{F}\left(\tau\right) \,\mathrm{d}\tau\right), \quad \mathbf{F}\left(t\right) = \nabla_{\mathbf{x}(t)} \mathbf{f}\left(\mathbf{x}\left(t\right)\right). \tag{4.72c}$$

Furthermore, the mean state $\hat{\mathbf{x}}\left(t_k + T\right)$ has to be calculated by solving the non-linear differential equation (4.42) with $\hat{\mathbf{x}}\left(t_k\right)$ as the initial state.

It may be impossible to find closed-form expressions for the transition matrix $\mathbf{\Phi}\left(t, t_0\right)$ and the integral (4.72b) in general. Fortunately, the non-linear CTRA model makes a remarkable exception. One key insight is that while the matrix exponential in (4.72c) can always be approximated by breaking the series expansion at a certain order

$$\mathbf{\Phi}\left(t, t_0\right) = \mathbf{I}_{n \times n} + \int_{t_0}^{t} \mathbf{F}\left(\tau\right) \,\mathrm{d}\tau + \frac{1}{2}\left[\int_{t_0}^{t} \mathbf{F}\left(\tau\right) \,\mathrm{d}\tau\right]^2 + \ldots, \tag{4.73}$$

all terms of order n and beyond vanish for the case where $\mathbf{F}\left(\tau\right)$ is an $n \times n$ strict upper diagonal matrix [107].

Example 4.3 (Constant turn rate and acceleration model)
The Jacobian of the model from (4.48) is obtained as a strict upper diagonal matrix. Thus, a closed-form expression for $\mathbf{\Phi}(t, t_0)$ can be obtained according to (4.73). Subsequently, (4.72b) can be solved in closed form to obtain $\mathbf{Q}(t_k + T, T)$ which finally yields an analytical expression for the predicted state uncertainty $\mathbf{\Sigma}_{\mathbf{x}}(t_k + T)$. Further details on this result are provided in the appendix in Sec. A.2.

In the special case of linear dynamics, the propagation to a Gaussian is exact. Since the Jacobian of a linear model with $\mathbf{f}(\mathbf{x}(t)) = \mathbf{F}\mathbf{x}(t)$ is constant, the transition matrix (4.72c) only depends on the time difference $t - t_0$ as in (4.47). This will be illustrated for the constant velocity model in the following.

Example 4.4 (Constant velocity model)
This model has been defined in (4.49). Again, $\mathbf{F}$ is a strict upper diagonal matrix[13] and one obtains $\mathbf{\Phi}(t, t_0) = \mathbf{I}_{4\times4} + \mathbf{F}(t - t_0)$. It follows for the mean and covariance of the state prediction from (4.72):

$$\hat{\mathbf{x}}(t_k + T) = (\mathbf{I}_{4\times4} + \mathbf{F}T)\,\mathbf{x}(t_k) \tag{4.74a}$$

$$\begin{aligned}\mathbf{\Sigma}_{\mathbf{x}}(t_k + T) = {} & (\mathbf{I}_{4\times4} + \mathbf{F}T)\,\mathbf{\Sigma}_{\mathbf{x}}(t_k)\,(\mathbf{I}_{4\times4} + \mathbf{F}T)^{\mathrm{T}} + \mathbf{L}\mathbf{S}\mathbf{L}^{\mathrm{T}}T \\ & + \frac{1}{2}\left(\mathbf{F}\mathbf{L}\mathbf{S}\mathbf{L}^{\mathrm{T}} + \mathbf{L}\mathbf{S}\mathbf{L}^{\mathrm{T}}\mathbf{F}^{\mathrm{T}}\right)T^2 + \frac{1}{3}\mathbf{F}\mathbf{L}\mathbf{S}\mathbf{L}^{\mathrm{T}}\mathbf{F}^{\mathrm{T}}T^3\,.\end{aligned} \tag{4.74b}$$

4.2.7 Gaussian prediction uncertainty in relative motion

In the previous section, uncertainty propagation has been considered for motion models in a ground-fixed coordinate system. In the case of a moving ego-vehicle, object trajectories can also be described in a Cartesian coordinate system that is centred and aligned to the pose of the ego-vehicle. An exemplary visualisation of the two coordinate systems is shown in Fig. 4.12. In the following, a Gaussian model of the state prediction density in these relative coordinates will be derived.

Using relative coordinates is beneficial for tracking and situation interpretation. One reason is that position measurements from exteroceptive sensors are naturally obtained relative to the ego-vehicle. Thus, with a motion model defined in relative coordinates, further transformations of the measurements are avoided [3]. A second advantage is that relations between the ego-vehicle and an object can be represented in an algebraically concise form. For example, a collision with a geometrically extended object is defined as an overlap with the ego-vehicle's contour. In relative coordinates, this equals that the

[13] Here, $\mathbf{F}$ is in fact 4×4-dimensional, but since the x and y components of the state are not coupled, the dynamics could be written as two independent systems with a 2×2-dimensional system matrix each. This explains why already the second order term of the series expansion vanishes.

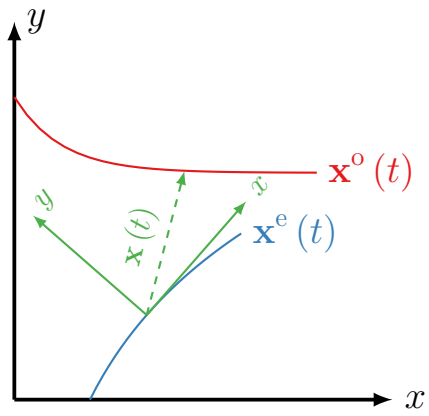


Figure 4.12 Illustration of ground-fixed coordinates (black) and the relative coordinate system of the moving ego-vehicle (green).

object position is located within an environment around the origin. Such an approach will be employed in Sec. 5.2 where algorithms for risk assessment are studied.

The notation used throughout this section is as follows: The dynamic states $\mathbf{x}^{\mathrm{e}}(t)$ of the ego-vehicle and the object $\mathbf{x}^{\mathrm{o}}(t)$ comprise Cartesian location and velocity components[14], that is $\mathbf{x}(t) = \begin{bmatrix} x(t) & y(t) & v_x(t) & v_y(t) \end{bmatrix}$. Moreover, it is assumed that the ego-vehicle's motion may contain rotations, where the heading angle is described by $\theta^{\mathrm{e}}(t)$ and yaw rate by $\omega^{\mathrm{e}}(t)$.

The transformation from ground-fixed Cartesian coordinates to a coordinate system relative to the ego-vehicle reads [3]:

$$\mathbf{x}(t) \coloneqq \mathbf{M}\left(\theta^{\mathrm{e}}(t), \omega^{\mathrm{e}}(t)\right)\left(\mathbf{x}^{\mathrm{o}}(t) - \mathbf{x}^{\mathrm{e}}(t)\right) \ , \tag{4.75a}$$

where

$$\mathbf{M}\left(\theta^{\mathrm{e}}(t), \omega^{\mathrm{e}}(t)\right) = \begin{bmatrix} \mathbf{R}\left(\theta^{\mathrm{e}}(t)\right) & \mathbf{0} \\ \dot{\mathbf{R}}\left(\theta^{\mathrm{e}}(t), \omega^{\mathrm{e}}(t)\right) & \mathbf{R}\left(\theta^{\mathrm{e}}(t)\right) \end{bmatrix} \tag{4.75b}$$

and $\mathbf{R}(\theta)$ defines a rotation matrix and $\dot{\mathbf{R}}(\theta, \omega)$ its time derivative:

$$\mathbf{R}(\theta) = \begin{bmatrix} \cos(\theta) & \sin(\theta) \\ -\sin(\theta) & \cos(\theta) \end{bmatrix} \ , \qquad \dot{\mathbf{R}}(\theta, \omega) = \begin{bmatrix} -\omega\sin(\theta) & \omega\cos(\theta) \\ -\omega\cos(\theta) & -\omega\sin(\theta) \end{bmatrix} \ . \tag{4.75c}$$

It will be outlined in the following that there are two principal ways to obtain a Gaussian representation of the state prediction density in relative coordinates: One, where the differential equations of the relative dynamics are derived first and the uncertainty is propagated for this model as described in Sec. 4.2.6. As it will be shown in Sec. 4.2.7.1, this approach can lead to very difficult expressions, depending on the assumed models for ego-vehicle and object motion. The second novel method, which is developed in Sec. 4.2.7.2, first propagates the uncertainty individually in absolute coordinates and

[14] The arrangement of the location and velocity components in the state vector, which is chosen here, leads to compact expressions since the rotation can be expressed with block matrices. All findings can be derived in a similar manner if higher moments of the motion, for example accelerations, are included.

then transforms the two distributions to relative coordinates. Since the motion models are treated independently, this approach is generic with respect to the employed models.

4.2.7.1 Uncertainty propagation via relative dynamics

The first approach is based on the differential equations of the relative motion state $\mathbf{x}(t)$. These follow from the dynamic equations of the two involved absolute motion models

$$\dot{\mathbf{x}}^{\mathrm{e}}(t) = \mathbf{f}^{\mathrm{e}}(\mathbf{x}^{\mathrm{e}}(t)) + \mathbf{L}^{\mathrm{e}}\mathbf{w}^{\mathrm{e}}(t) \ , \qquad \dot{\mathbf{x}}^{\mathrm{o}}(t) = \mathbf{f}^{\mathrm{o}}(\mathbf{x}^{\mathrm{o}}(t)) + \mathbf{L}^{\mathrm{o}}\mathbf{w}^{\mathrm{o}}(t) \tag{4.76}$$

and the transformation (4.75).

To derive the relative dynamics, the evolution of the ego-vehicle state $\mathbf{x}^{\mathrm{e}}(t)$ and $\theta^{\mathrm{e}}(t)$, $\omega^{\mathrm{e}}(t)$ has to be obtained first. Therefore, the transformation matrix only depends on the time $\mathbf{M}(t) \coloneqq \mathbf{M}(\theta^{\mathrm{e}}(t), \omega^{\mathrm{e}}(t))$. Then, the following time-variant differential equations describe the dynamics of $\mathbf{x}(t)$ [3]:

$$\begin{aligned}\dot{\mathbf{x}}(t) = \dot{\mathbf{M}}(t)\,\mathbf{M}^{-1}(t)\,\mathbf{x}(t) + \mathbf{M}(t) \\ \cdot \left[\mathbf{f}^{\mathrm{o}}\left(\mathbf{M}^{-1}(t)\,\mathbf{x}(t) + \mathbf{x}^{\mathrm{e}}(t)\right) + \mathbf{L}^{\mathrm{o}}\mathbf{w}^{\mathrm{o}}(t) - \mathbf{f}^{\mathrm{e}}(\mathbf{x}^{\mathrm{e}}(t)) - \mathbf{L}^{\mathrm{e}}\mathbf{w}^{\mathrm{e}}(t)\right] \ .\end{aligned} \tag{4.77}$$

Obviously, solving these equations to obtain the uncertainty propagation may become very difficult. In special cases though, an intuitive solution can be obtained.

Example 4.5 (Linear translational models)

We assume that both ego-vehicle and object move according to the same purely translational motion, for example the constant velocity model (4.49) from example 4.2. Instead of (4.76), one has

$$\dot{\mathbf{x}}^{\mathrm{e}}(t) = \mathbf{F}\mathbf{x}^{\mathrm{e}}(t) + \mathbf{L}\mathbf{w}^{\mathrm{e}}(t) \ , \qquad \dot{\mathbf{x}}^{\mathrm{o}}(t) = \mathbf{F}\mathbf{x}^{\mathrm{o}}(t) + \mathbf{L}\mathbf{w}^{\mathrm{o}}(t) \tag{4.78}$$

and $\theta^{\mathrm{e}}(t) = 0$, $\omega^{\mathrm{e}}(t) = 0$. This choice yields for the relative dynamics (4.77):

$$\begin{aligned}\dot{\mathbf{x}}(t) &= \mathbf{F}(\mathbf{x}(t) + \mathbf{x}^{\mathrm{e}}(t)) + \mathbf{L}\mathbf{w}^{\mathrm{o}}(t) - \mathbf{F}\mathbf{x}^{\mathrm{e}}(t) - \mathbf{L}\mathbf{w}^{\mathrm{e}}(t) \\ &= \mathbf{F}\mathbf{x}(t) + \mathbf{L}(\mathbf{w}^{\mathrm{o}}(t) - \mathbf{w}^{\mathrm{e}}(t)) \ .\end{aligned} \tag{4.79}$$

The relative state $\mathbf{x}$ adheres to the same differential equations as the absolute motion state but with the process noise added up. A closed-form solution for the Gaussian state prediction density $\mathcal{N}(\mathbf{x}(t_k + T); \hat{\mathbf{x}}(t_k + T), \mathbf{\Sigma}_{\mathbf{x}}(t_k + T))$ is therefore available.

4.2.7.2 Uncertainty propagation via absolute dynamics

To overcome the challenges that are caused by the relative motion dynamics (4.77), it is proposed to separately propagate uncertainties in absolute coordinates with the methods from Sec. 4.2.6 first and subsequently transform the resulting covariances $\mathbf{\Sigma}^{\mathrm{e}}_{\mathbf{x}}(t_k + T), \mathbf{\Sigma}^{\mathrm{o}}_{\mathbf{x}}(t_k + T)$ to $\mathbf{\Sigma}_{\mathbf{x}}(t_k + T)$ in the relative frame through (4.75).

If $\mathbf{M}\left(\theta^{\mathrm{e}}\left(t_k+T\right),\omega^{\mathrm{e}}\left(t_k+T\right)\right)$ is known exactly, the transformation reads:

$$\begin{aligned}\boldsymbol{\Sigma}_{\mathbf{x}}\left(t_k+T\right) = \mathbf{M}\left(\theta^{\mathrm{e}}\left(t_k+T\right),\omega^{\mathrm{e}}\left(t_k+T\right)\right)\cdot\left(\boldsymbol{\Sigma}_{\mathbf{x}}^{\mathrm{o}}\left(t_k+T\right)+\boldsymbol{\Sigma}_{\mathbf{x}}^{\mathrm{e}}\left(t_k+T\right)\right)\\ \cdot\mathbf{M}^{\mathrm{T}}\left(\theta^{\mathrm{e}}\left(t_k+T\right),\omega^{\mathrm{e}}\left(t_k+T\right)\right)\ .\end{aligned} \tag{4.80}$$

In practice, the heading angle and yaw rate of the ego-vehicle are subject to uncertainty as well. Therefore, the transformation matrix $\mathbf{M}\left(\theta^{\mathrm{e}}\left(t_k+T\right),\omega^{\mathrm{e}}\left(t_k+T\right)\right)$ has to be considered as a random variable. Since $\mathbf{x}\left(t_k+T\right)$ is thus given by a product of two Gaussians, its distribution is in general non-Gaussian. The idea of the proposed approach is to firstly write these products as quadratic forms of a Gaussian random variable. Secondly, closed-form results that are available for the moments of such quadratic forms yield a Gaussian approximation.

To this end, the ith row $\mathbf{m}_i^{\mathrm{T}}\left(\theta^{\mathrm{e}},\omega^{\mathrm{e}}\right)$ of $\mathbf{M}\left(\theta^{\mathrm{e}},\omega^{\mathrm{e}}\right)$ is firstly rewritten as a linear mapping $\mathbf{m}_i\left(\theta^{\mathrm{e}},\omega^{\mathrm{e}}\right)=\left(\nabla_{\mathbf{q}}\mathbf{m}_i\right)\cdot\mathbf{q}\left(\theta^{\mathrm{e}},\omega^{\mathrm{e}}\right)$, where

$$\mathbf{q}\left(\theta^{\mathrm{e}},\omega^{\mathrm{e}}\right) := \begin{bmatrix}\cos\left(\theta^{\mathrm{e}}\right) & \sin\left(\theta^{\mathrm{e}}\right) & \omega^{\mathrm{e}}\cos\left(\theta^{\mathrm{e}}\right) & \omega^{\mathrm{e}}\sin\left(\theta^{\mathrm{e}}\right)\end{bmatrix}^{\mathrm{T}} \tag{4.81}$$

are the trigonometric expressions which (4.75b) consists of.

Moreover, an abbreviation for the relative motion state prior to rotation into the ego-vehicle's coordinate system is introduced as $\boldsymbol{\xi} := \mathbf{x}^{\mathrm{o}}-\mathbf{x}^{\mathrm{e}}$ with $\boldsymbol{\xi}\sim\mathcal{N}\left(\hat{\boldsymbol{\xi}},\boldsymbol{\Sigma}_{\boldsymbol{\xi}}\right)$ and $\boldsymbol{\Sigma}_{\boldsymbol{\xi}} := \boldsymbol{\Sigma}_{\mathbf{x}}^{\mathrm{o}}+\boldsymbol{\Sigma}_{\mathbf{x}}^{\mathrm{e}}$. Hence, the ith component of $\mathbf{x}$ can be written equivalently to (4.75) as:

$$x_i = \frac{1}{2}\begin{bmatrix}\boldsymbol{\xi}^{\mathrm{T}} & \mathbf{q}^{\mathrm{T}}\left(\theta^{\mathrm{e}},\omega^{\mathrm{e}}\right)\end{bmatrix}\begin{bmatrix}\mathbf{0}_{4\times4} & \left(\nabla_{\mathbf{q}}\mathbf{m}_i\right)\\ \left(\nabla_{\mathbf{q}}\mathbf{m}_i\right)^{\mathrm{T}} & \mathbf{0}_{4\times4}\end{bmatrix}\begin{bmatrix}\boldsymbol{\xi}\\ \mathbf{q}\left(\theta^{\mathrm{e}},\omega^{\mathrm{e}}\right)\end{bmatrix}\ . \tag{4.82}$$

Now, a result on the moments of quadratic forms can be employed. That is, if two quadratic forms $\mathbf{y}^{\mathrm{T}}\mathbf{A}_i\mathbf{y}$ and $\mathbf{y}^{\mathrm{T}}\mathbf{A}_j\mathbf{y}$ are defined for a multivariate normal random variable $\mathbf{y}\sim\mathcal{N}\left(\boldsymbol{\mu}_y,\boldsymbol{\Sigma}_{\mathbf{y}}\right)$, it holds for the cross-covariance [134]:

$$\operatorname{cov}\left(\mathbf{y}^{\mathrm{T}}\mathbf{A}_i\mathbf{y},\mathbf{y}^{\mathrm{T}}\mathbf{A}_j\mathbf{y}\right) = 2\operatorname{tr}\left(\mathbf{A}_i\boldsymbol{\Sigma}_{\mathbf{y}}\mathbf{A}_j\boldsymbol{\Sigma}_{\mathbf{y}}\right)+4\boldsymbol{\mu}_y^{\mathrm{T}}\mathbf{A}_i\boldsymbol{\Sigma}_{\mathbf{y}}\mathbf{A}_j\boldsymbol{\mu}_y\ . \tag{4.83}$$

With this general result, the covariance matrix $\boldsymbol{\Sigma}_{\mathbf{x}} = \left[\operatorname{cov}\left(x_i,x_j\right)\right]_{i,j}$ can be obtained element-wise:

$$\begin{aligned}\operatorname{cov}\left(x_i,x_j\right) = \frac{1}{2}\operatorname{tr}\left(\begin{bmatrix}\left(\nabla_{\mathbf{q}}\mathbf{m}_i\right)\boldsymbol{\Sigma}_{\mathbf{q}}\left(\nabla_{\mathbf{q}}\mathbf{m}_j\right)^{\mathrm{T}}\boldsymbol{\Sigma}_{\boldsymbol{\xi}} & \mathbf{0}_{4\times4}\\ \mathbf{0}_{4\times4} & \left(\nabla_{\mathbf{q}}\mathbf{m}_i\right)^{\mathrm{T}}\boldsymbol{\Sigma}_{\boldsymbol{\xi}}\left(\nabla_{\mathbf{q}}\mathbf{m}_j\right)\boldsymbol{\Sigma}_{\mathbf{q}}\end{bmatrix}\right)\\ +\begin{bmatrix}\mathbb{E}\left[\mathbf{m}_i^{\mathrm{T}}\left(\theta^{\mathrm{e}},\omega^{\mathrm{e}}\right)\right] & \hat{\boldsymbol{\xi}}^{\mathrm{T}}\end{bmatrix}\begin{bmatrix}\boldsymbol{\Sigma}_{\boldsymbol{\xi}} & \mathbf{0}_{4\times4}\\ \mathbf{0}_{4\times4} & \left(\nabla_{\mathbf{q}}\mathbf{m}_i\right)\boldsymbol{\Sigma}_{\mathbf{q}}\left(\nabla_{\mathbf{q}}\mathbf{m}_j\right)^{\mathrm{T}}\end{bmatrix}\begin{bmatrix}\mathbb{E}\left[\mathbf{m}_i\left(\theta^{\mathrm{e}},\omega^{\mathrm{e}}\right)\right]\\ \hat{\boldsymbol{\xi}}\end{bmatrix}\ .\end{aligned} \tag{4.84}$$

Note that $\mathbf{q}\left(\theta^{\mathrm{e}},\omega^{\mathrm{e}}\right)$ is given by a non-linear transformation (4.81) of the Gaussian heading angle and yaw rate. We assume that $\boldsymbol{\theta} := \begin{bmatrix}\theta^{\mathrm{e}} & \omega^{\mathrm{e}}\end{bmatrix}$ is distributed according to $\boldsymbol{\theta}\sim\mathcal{N}\left(\hat{\boldsymbol{\theta}},\boldsymbol{\Sigma}_{\boldsymbol{\theta}}\right)$. Then, the covariance of $\mathbf{q}\left(\theta^{\mathrm{e}},\omega^{\mathrm{e}}\right)$ is given by $\boldsymbol{\Sigma}_{\mathbf{q}} = \left(\nabla_{\boldsymbol{\theta}}\mathbf{q}\right)\boldsymbol{\Sigma}_{\boldsymbol{\theta}}\left(\nabla_{\boldsymbol{\theta}}\mathbf{q}\right)^{\mathrm{T}}$. Furthermore, the expectation of $\mathbf{M}\left(\theta^{\mathrm{e}},\omega^{\mathrm{e}}\right)$ to first order is $\mathbb{E}\left[\mathbf{m}_i\left(\theta^{\mathrm{e}},\omega^{\mathrm{e}}\right)\right]=\mathbf{m}_i\left(\hat{\theta}^{\mathrm{e}},\hat{\omega}^{\mathrm{e}}\right)$.

Now, since $\nabla_{\mathbf{q}}\mathbf{m}_i\nabla_{\boldsymbol{\theta}}\mathbf{q}\left(\theta^{\mathrm{e}},\omega^{\mathrm{e}}\right) = \nabla_{\boldsymbol{\theta}}\mathbf{m}_i\left(\theta^{\mathrm{e}},\omega^{\mathrm{e}}\right)$, one obtains a simplification of the intermediate result (4.84) without the auxiliary variable $\mathbf{q}$:

$$\operatorname{cov}\left(x_i,x_j\right) = \mathbf{m}_i^{\mathrm{T}}\left(\hat{\theta}^{\mathrm{e}},\hat{\omega}^{\mathrm{e}}\right)\boldsymbol{\Sigma}_{\boldsymbol{\xi}}\mathbf{m}_j\left(\hat{\theta}^{\mathrm{e}},\hat{\omega}^{\mathrm{e}}\right) + \operatorname{tr}\left(\boldsymbol{\Sigma}_{\mathbf{m}_i,\mathbf{m}_j}\boldsymbol{\Sigma}_{\boldsymbol{\xi}}\right) + \hat{\boldsymbol{\xi}}^{\mathrm{T}}\boldsymbol{\Sigma}_{\mathbf{m}_i,\mathbf{m}_j}\hat{\boldsymbol{\xi}} \tag{4.85a}$$

$$\text{with } \boldsymbol{\Sigma}_{\mathbf{m}_i,\mathbf{m}_j} = \left(\nabla_{\boldsymbol{\theta}}\mathbf{m}_i\right)\boldsymbol{\Sigma}_{\boldsymbol{\theta}}\left(\nabla_{\boldsymbol{\theta}}\mathbf{m}_i\right)^{\mathrm{T}}\ . \tag{4.85b}$$

When comparing this result to (4.80), it can be recognised that the first term in (4.85a) equals the covariance in the case of certain $\boldsymbol{\theta}$. The additional term $\hat{\boldsymbol{\xi}}^{\mathrm{T}}\boldsymbol{\Sigma}_{\mathbf{m}_i,\mathbf{m}_j}\hat{\boldsymbol{\xi}}$ depends on the difference $\hat{\boldsymbol{\xi}}$ between the two motion states. Thereby, higher distances increase the uncertainty due to the uncertain heading angle. The effect of this additional uncertainty will be illustrated with a simulation example in the next section.

4.2.8 Evaluation

This section presents a twofold evaluation of the previously detailed uncertainty models of vehicle motion predictions. At first, the analytical propagation of the state uncertainty from Sec. 4.2.6 and the transformation to relative dynamics from Sec. 4.2.7 are addressed in simulations. All model parameters are pre-defined and solely the uncertainty transformation is analysed. Secondly, trajectory recordings from real traffic situations are used to evaluate the process noise models which have been estimated according to Sec. 4.2.4-4.2.5. Thus, both the parameter estimates as well as the applicability of the Gaussian white noise assumption are evaluated.

4.2.8.1 Simulations

All simulations are based on the configuration shown in Fig. 4.13. The ego-vehicle is moving in the x-direction according to the CTRA model (4.48), towards an object which is crossing into the driving corridor (CV model (4.49)). It is assumed that the driver has anticipated the situation and performs an evasive steering manoeuvre. Initial values and the process noise parameters are given in Tab. 4.3.

Firstly, $N_{\mathrm{sim}} = 1 \times 10^2$ trajectory predictions of the ego-vehicle and object are calculated. The analytical propagation from Sec. 4.2.6 is obtained using the respective models and visualised in the form of 90% credibility ellipses. Evaluated at six different time steps, the results in Fig. 4.14 show good correspondence.

Secondly, the resulting relative trajectories as seen from the ego-vehicle are visualised in Fig. 4.15. The propagation model from Sec. 4.2.7 is shown without (4.80) and with (4.85) taking the uncertainty in the rotation angle into account. Neglecting the heading errors causes too narrow credibility ellipses, most notable at far distances. In these cases, even small angle errors lead to high lateral deviations.

Besides this first qualitative evaluation, the accuracy of the analytical prediction is quantitatively assessed. To this end, $N_{\mathrm{sim}} = 1 \times 10^4$ trajectories $\mathbf{x}^{(i)}\left(t_k + T\right)$ are simulated. The relative frequency of trajectories that lie within a finite region $\mathbf{X}_\alpha$ of the state space at each time step is compared to the models' predictions of this probability.

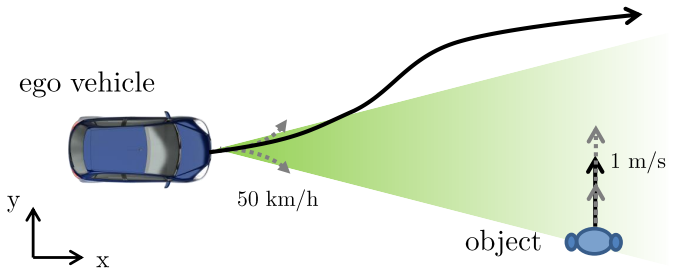
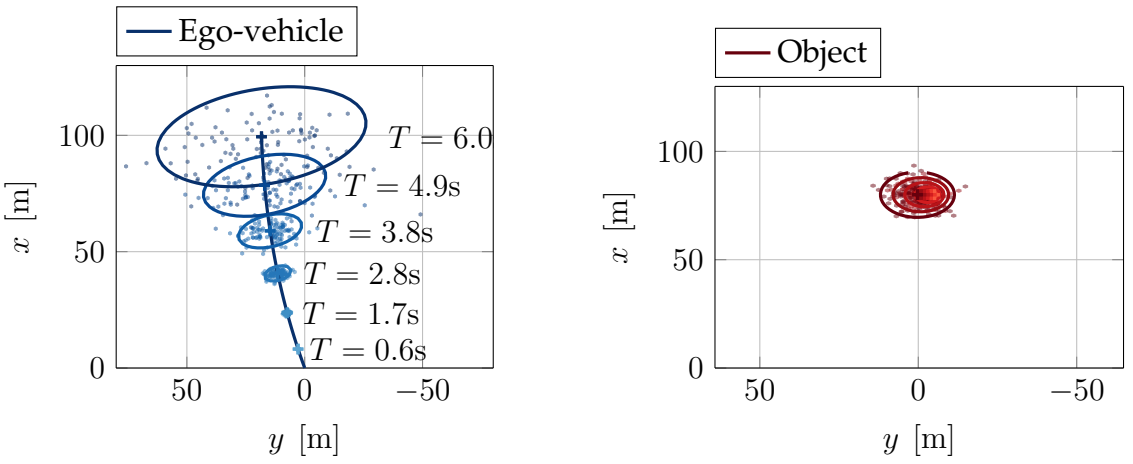


Figure 4.13 Illustration of the simulation setup.



(a) CTRA model (4.48). **(b)** CV model (4.49).

Figure 4.14 Absolute positions. $N_{\text{sim}} = 1 \times 10^2$ trajectories are generated with randomly sampled initial values and process noise realisations. The state distribution at specific time-steps is visualised by dots. For the analytical uncertainty propagation from Sec. 4.2.6, solid lines are used to indicate the mean trajectory and 90% quantiles.

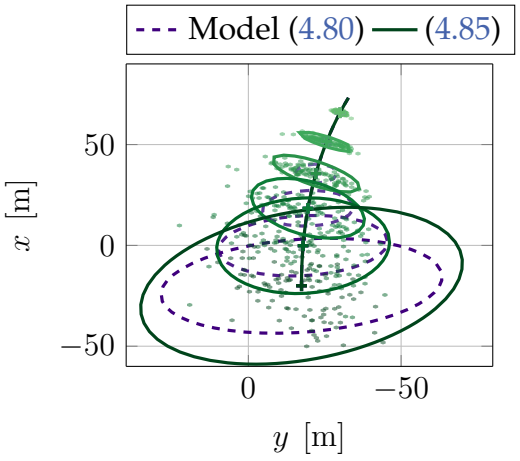


Figure 4.15 Relative positions. The analytical propagation from Sec. 4.2.7 is calculated with (solid) and without (dashed) considering uncertainty in heading θ^{e} and yaw rate ω^{e}.

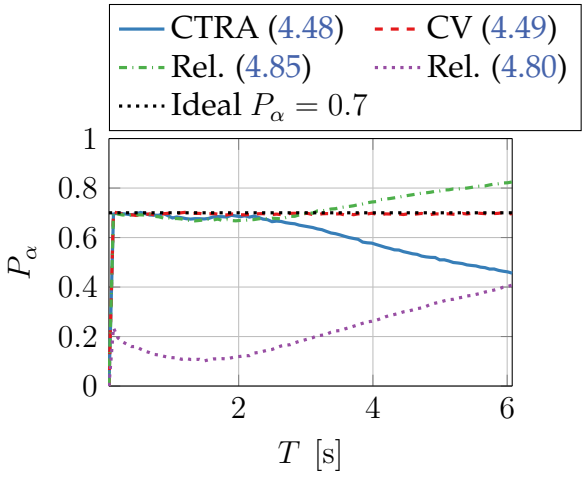


Figure 4.16 Coverage of models. Ideally, a proportion of $P_\alpha = 0.7$ of all simulated trajectories lies in a hyperellipse (4.86) that is defined by the model prediction.

Table 4.3 Parameter values for the simulation of state prediction uncertainty.

	Variable	Value
Ego-vehicle	Initial state mean	$\hat{\mathbf{x}}^{\mathrm{e}}(t_k) = \begin{bmatrix} 0\,\mathrm{m} & 0\,\mathrm{m} & 13.89\,\mathrm{m/s} & 20^\circ & 1\,\mathrm{m/s^2} & -0.0524\,\mathrm{rad/s} \end{bmatrix}^{\mathrm{T}}$
	Initial state covariance	$\boldsymbol{\Sigma}_{\mathbf{x}}^{\mathrm{e}}(t_k) = \mathrm{diag}\left(0.01\,\mathrm{m}^2, 0.01\,\mathrm{m}^2, 0.0025\,\mathrm{m}^2/\mathrm{s}^2, 0, 0.01\,\mathrm{m}^2/\mathrm{s}^4, 0\right)$
	Process noise	$S_a^{\mathrm{e}} = 0.224\,\mathrm{m}^2/\mathrm{s}^6\mathrm{s}^{-1}$, $S_\omega^{\mathrm{e}} = 0.0038\,\mathrm{rad}^2/\mathrm{s}^4\mathrm{s}^{-1}$
Object	Initial state mean	$\hat{\mathbf{x}}^{\mathrm{o}}(t_k) = \begin{bmatrix} 80\,\mathrm{m} & -5.75\,\mathrm{m} & 0\,\mathrm{m/s} & 1\,\mathrm{m/s} \end{bmatrix}^{\mathrm{T}}$
	Initial state covariance	$\boldsymbol{\Sigma}_{\mathbf{x}}^{\mathrm{o}}(t_k) = \mathrm{diag}\left(0.25\,\mathrm{m}^2, 0.25\,\mathrm{m}^2, 0.0625\,\mathrm{m}^2/\mathrm{s}^2, 0.0625\,\mathrm{m}^2/\mathrm{s}^2\right)$
	Process noise	$S_x^{\mathrm{o}} = 0.375\,\mathrm{m}^2/\mathrm{s}^4\mathrm{s}^{-1}$, $S_y^{\mathrm{o}} = 0.293\,\mathrm{m}^2/\mathrm{s}^4\mathrm{s}^{-1}$

The finite region in which the model's coverage $P_\alpha := \mathrm{P}\left(\mathbf{x} \in \mathbf{X}_\alpha\right)$ is evaluated is defined by a hyperellipse around the predicted mean $\hat{\mathbf{x}}$:

$$\mathbf{X}_\alpha := \left\{ \mathbf{x} \middle| \, (\mathbf{x} - \hat{\mathbf{x}}) \cdot \boldsymbol{\Sigma}_{\mathbf{x}}^{-1} \cdot (\mathbf{x} - \hat{\mathbf{x}})^{\mathrm{T}} \leq Q_{\chi_n^2}(\alpha) \right\} . \tag{4.86}$$

Here, $Q_{\chi_n^2}(\alpha)$ refers to the quantile of a χ^2 distribution with n degrees of freedom where n is the dimension of the state $\mathbf{x}$. Under the Gaussian model assumption, the hyperellipse (4.86) encompasses a probability of $P_\alpha = \alpha$. Since the model predictions $(\hat{\mathbf{x}}(t_k + T), \boldsymbol{\Sigma}_{\mathbf{x}}(t_k + T))$ depend on the prediction horizon T, these ellipses are functions of T.

For each T, the proportion of all simulated trajectories that fall into the respective hyperellipse $\mathbf{X}_\alpha(T)$ are calculated. This relative frequency P_α is shown in comparison to the ideal outcome in Fig. 4.16.

It can be recognised that only the linear CV model yields consistent results over the entire prediction horizon. This is not surprising since the Gaussian state propagation for the non-linear CTRA model is only an approximation. In this case, an acceptable accuracy until approximately $T = 2\,\mathrm{s}$ is achieved but the deviations increase for higher prediction times. Note that the baseline model in relative coordinates (4.85) leads to strong underestimations. This has been previously seen from the too narrow ellipses in Fig. 4.15.

4.2.8.2 Prediction error distribution in real traffic

Introduction In the previous section, the analytical propagation of an initial state distribution to a future predicted time is studied in simulations for a known parametrisation of the models. Now, the goal is to evaluate the estimated process noise models from Sec. 4.2.4 for the CV, CA and CTRA model using real-world vehicle trajectories.

Essentially, the approach is to predict both the mean trajectory and covariance and compare these to the true trajectory. In contrast to previous analyses, for example [175],

not only the deviations of the mean state but also the error distribution compared to the predicted covariance are evaluated. The rationale is that while prediction errors are inevitable with any model, knowing that these follow a given uncertainty model is of great value for example for further decision making. Eventually, this comparison tries to answer the questions whether

1. the assumed uncertainty model (4.43) with white Gaussian process noise is reasonable and

2. the estimated model parameters generalise to other trajectories than the ones used for estimation.

Method To evaluate the predictions, vehicle trajectories are taken from the dataset as described in Sec. 4.2.5.2. The length of an individual sequence is 6 s for straight trajectories where the first 3 s are used to estimate the initial state and the remaining 3 s to evaluate the actual prediction model. Similarly, curved trajectories with a total duration of 4 s are split evenly for predictions of up 2 s.

The data comprises 50% of the entire raw recordings or $N_{\text{eval,straight}} = 8189$ and $N_{\text{eval,curves}} = 1125$ sequences. Model estimation and validation are therefore conducted on two disjoint datasets. Approaches that divide the entire dataset in multiple rounds, for example k-fold cross-validation, are not applied due to the already vast amount of available data. Moreover, qualitative comparisons of the distributions can be performed without the need to aggregate the results from multiple rounds.

In order to calculate predictions with the three models, initial state values are required. Henceforth, an extended Kalman filter is applied to the first half of each trajectory and estimates $(\hat{\mathbf{x}}(t_k), \boldsymbol{\Sigma}_{\mathbf{x}}(t_k))$ are obtained. The actual predictions $\hat{\mathbf{x}}(t_k + T)$ and $\boldsymbol{\Sigma}_{\mathbf{x}}(t_k + T)$ are then calculated using (4.72) for the subsequent $T = 0 \ldots 3\,\text{s}$ for straight and $T = 0 \ldots 2\,\text{s}$ for curved trajectories.

Reference values for comparison are obtained by applying an extended Kalman smoother to the entire sequences. The prediction error is then given by the difference of the predicted state variables to this reference trajectory. For straight trajectories, the longitudinal motion is of main interest. Therefore, the differences in position and velocity are projected on the reference trajectory and the longitudinal components $\Delta x(T)$ and $\Delta v_x(T)$ are used as evaluation metrics. Curved trajectories on the other hand are evaluated in terms of longitudinal $\Delta x(T)$ and lateral $\Delta y(T)$ position errors.

By calculating the prediction errors for each trajectory at each time step of the prediction horizon, empirical error distributions are obtained. These are compared to the normal distributions defined by the predicted covariances[15] $\boldsymbol{\Sigma}_{\mathbf{x}}(t_k + T)$. The comparison is conducted qualitatively and quantitatively:

[15] In fact, slightly different variances are predicted for each sequence, depending on the initial Kalman filter estimate. Hence, the mean value is taken for the evaluation.

1. Histograms and quantile-quantile plots at three distinct prediction times $T \in \{1\,\mathrm{s}, 1.5\,\mathrm{s}, 2\,\mathrm{s}\}$ allow a visual comparison of the empirical error distribution to the model.

2. 70% percentile values of the absolute errors over T are compared to the corresponding value of the predicted normal distribution.

Thereby, the model assumptions as well as the estimated parameter values are evaluated. As a reference for comparison, a normal distribution is fitted to the empirical error distribution. This corresponds to the best possible Gaussian model as its parameters are estimated directly from the evaluation dataset.

Results The resulting error and model distributions are visualised in Fig. 4.17-4.18 and Fig. 4.19 for straight trajectories and curves, respectively.

As is expected, these results exemplify how the uncertainty on the future vehicle location and velocity increases with prediction time. Comparing the different models reveals, for example, that the CA model features slightly lower errors than the CV model, especially in terms of the highest absolute values. This concurs with the findings in [175].

Concerning the assumption of a Gaussian error model, two kinds of deviations can be observed for longitudinal motion in Fig. 4.17-4.18:

- On the one hand, heavy distribution tails are observed, that is high deviations occur more frequently than expected. This is visible from the non-linear shape in Fig. 4.17(b) and 4.18(b). The CA and CTRA model reveal a better behaviour in this regard than the simple CV model, since they take (constant) acceleration commands explicitly into account.

- On the other hand, the distribution peaks are slightly underestimated by the CA and CTRA model as seen in the histograms in Fig. 4.17(a) and Fig. 4.18(a). This leads to a too conservative estimation of the 70% percentile values in Fig. 4.17(c) and Fig. 4.18(c). The a posteriori reference fits better accommodate the distribution around the mean. However, as seen from the quantile-quantile plots Fig. 4.17(b) and Fig. 4.18(b), the approximation of the distribution tails is worse than for the predicted model. Therefore, describing the non-Gaussian error distribution with a Gaussian model leads to the compromise whether a good fit of the behaviour around the mean or the distribution tails is preferred.

Possible reasons for the observed non-Gaussianity are a violation of the assumption of white, uncorrelated driver inputs which is a strong simplification of reality. If acceleration commands (not considered by the CV model) are applied, these will typically last for time spans up to multiple seconds. A similar reasoning explains the overly high peaks of the acceleration models, where a driver keeps his inputs at a constant value (that is correlated) for some time.

In all cases, the deviation between model prediction and empirical values increases for higher prediction times, as it is revealed by the diverging percentile values over time

in Fig. 4.17(c) and Fig. 4.18(c). Overall, the reference fits are slightly better adapted to the distribution peaks. However, the tails are underestimated as well. This indicates that the estimation was correctly performed and yields sensible estimates within the limits of the underlying Gaussian assumptions.

In the case of curved trajectories, only the results of the CTRA model are depicted in Fig. 4.19. The assumption of a translational motion, as included in the CV and CA model, is obviously strongly violated in curves which causes high prediction errors. Moreover, these errors depend mainly on the road geometry and the curvature at the time of prediction. Such dependencies are not explicitly included in the process noise model which describes prediction errors in the form of stochastic driver inputs.

Overall, the deviations between true trajectories and the ones predicted by the CTRA model grow strongly with the prediction horizon. It can be seen that the 99% percentile values which are displayed in Fig. 4.19(b) are almost twice as high at $T = 2\,\mathrm{s}$ than at $T = 1\,\mathrm{s}$. The dependence on a curve's geometry might explain the high deviations in the lateral direction for increased prediction times. Once a curve is passed, the yaw rate drops to zero and the value which has been estimated during the curve is no longer a valid initial state for the prediction.

Conclusion In summary, the following conclusions can be drawn about the probabilistic vehicle motion predictions. Concerning the longitudinal motion in straight trajectories, predictions with zero mean errors and a reasonable description of the error distribution have been obtained. Slight inaccuracies in the predicted uncertainty arise due to a non-Gaussian shape with heavy tails.

The vehicle motion in curves is determined by the changing geometry of the road. This affects the predictions given by a CTRA model under the assumptions of constant acceleration and yaw rate. A strong increase and non-Gaussian distributions of the prediction errors in lateral direction have been noticed for a prediction horizon $T \geq 1\,\mathrm{s}$. This might be explained by the finite length of real curves.

4.2.9 Summary

This section has studied parameter identification and Gaussian uncertainty propagation for stochastic non-linear dynamic systems. Specifically, kinematic vehicle motion models have been analysed which are used in driver assistance functions to predict the future trajectories of other road users. The stochastic uncertainties in these predictions have been assumed to be driven by Gaussian white process noise. With this assumption, closed-form expressions for the distribution of the predicted states have been derived.

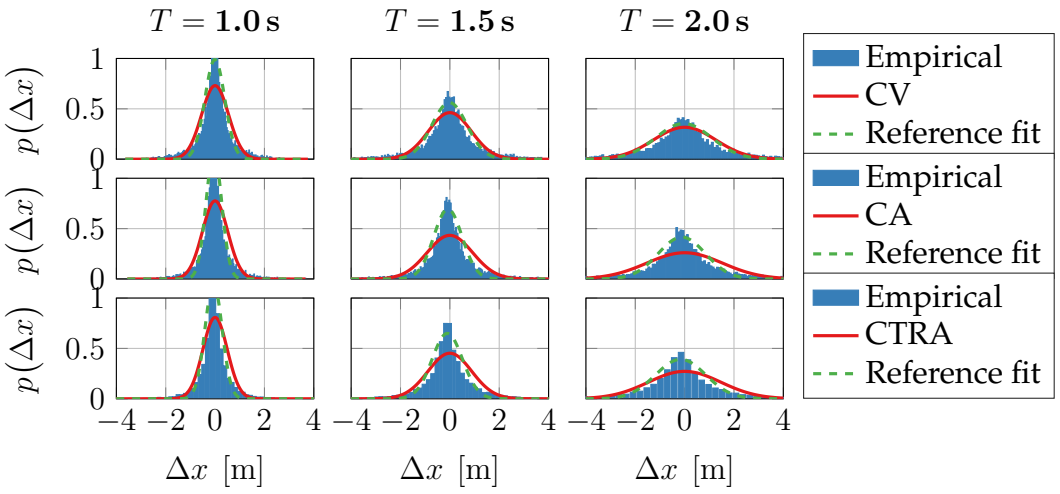
The process noise model parameters have been estimated with the iterative expectation maximisation algorithm. Theoretical foundations of the method have been first explained and then applied to the specific motion models. A generic closed-form result for the estimation of covariance matrices with a Kronecker product structure has been derived.

Estimating the parameters for three commonly used models has been then performed using an extensive dataset which comprises 170 h of driving data. Because the dataset is based on trajectories measured by surround environment perception, it encompasses a large variety of different vehicles and driving styles.

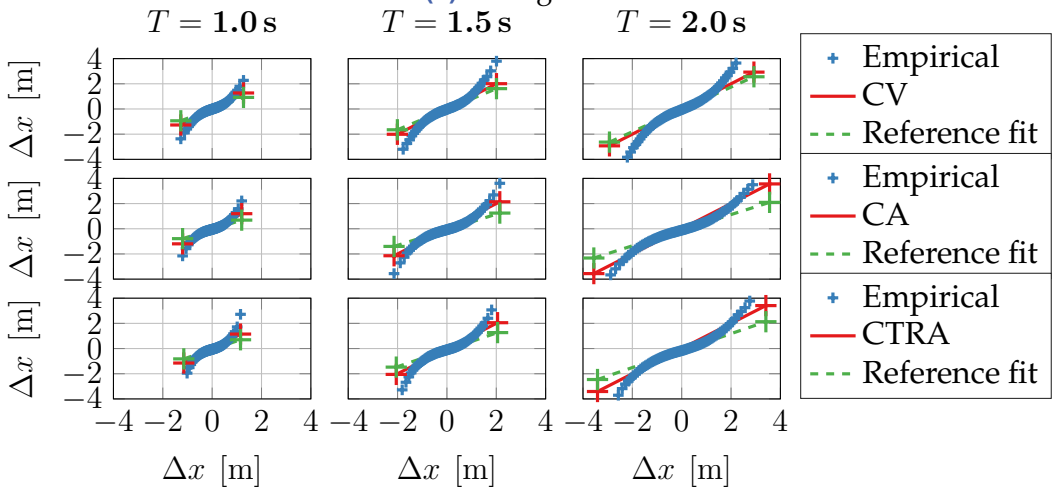
The appropriateness of a Gaussian process noise model with the estimated parametrisation has been evaluated on a second disjoint evaluation dataset. It has been found that prediction errors are often non-Gaussian and feature a heavy-tailed distribution. While a Gaussian noise model leads to satisfying results in describing the errors around the mean, rare cases of strong deviations are not predicted. The comparison of the three different models furthermore reveals that models which take (constant) accelerations into account are less affected by these outliers than a constant velocity model.

For driver assistance functions, not only the trajectories in a ground-fixed coordinate system but an object's motion relative to the moving ego-vehicle are relevant. In order to transform the uncertainty models into the moving frame of the ego-vehicle, a novel method has been developed. The proposed idea is to separately propagate the stochastic trajectories of ego-vehicle and object first and subsequently transform the distributions from absolute to relative motion. In contrast to an immediate propagation in relative coordinates, simple and more generic results are obtained.

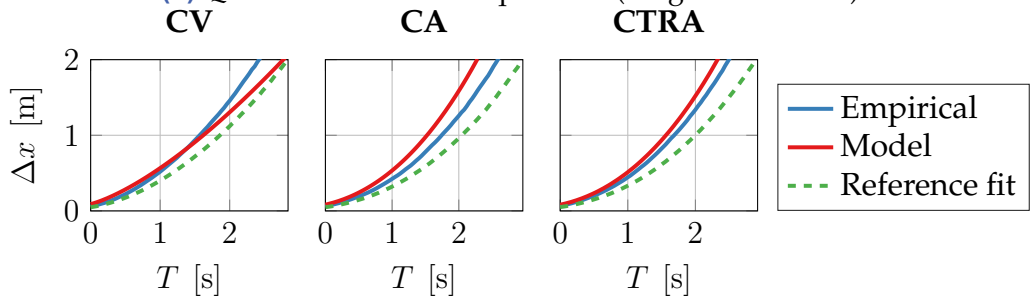
Essentially, trajectory predictions can be regarded as input quantities of a situation assessment algorithm. Similar to sensor measurement errors, the statistical models from this section will be propagated to such algorithms in Sec. 5.2.



(a) Histograms.

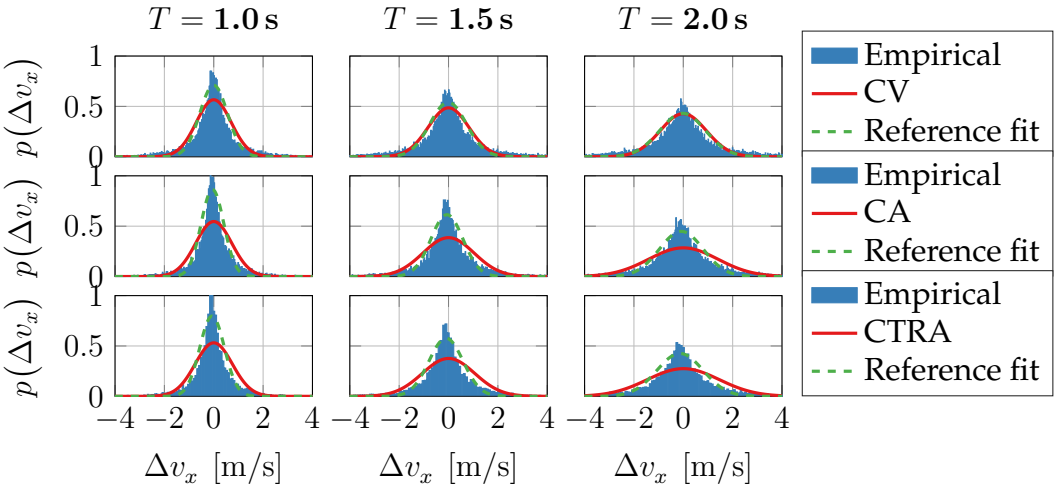


(b) Quantiles over model quantiles (range: $1\% \ldots 99\%$).

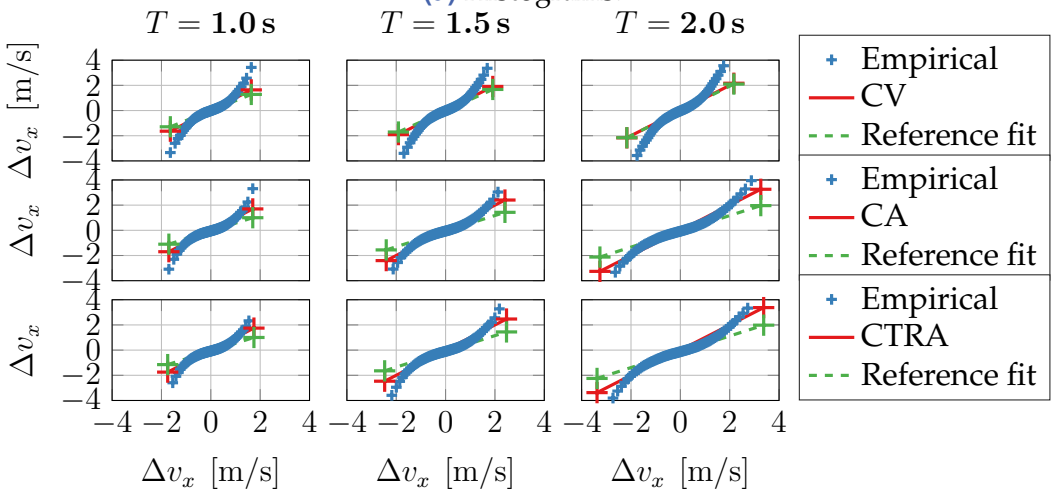


(c) 70% percentile values of absolute error.

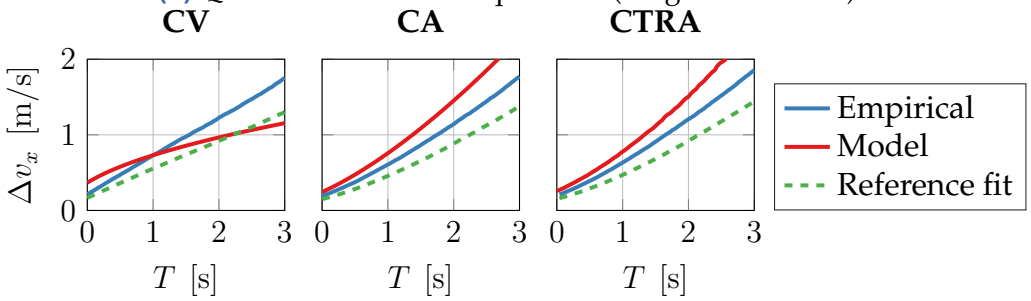
Figure 4.17 **Straight trajectories:** Longitudinal prediction errors are compared to the prediction model distributions and a-posteriori Gaussian fits for different prediction times $T \in \{1\,\mathrm{s}, 1.5\,\mathrm{s}, 2\,\mathrm{s}\}$. The non-Gaussianity of the error distribution becomes apparent in the non-linear shape of the quantile-quantile plots in (b). Still, the a-posteriori fits that are used to validate the model agree with the predicted distribution in most cases. Hence, the derived model provides a reasonable prediction within the limits of a Gaussian model. Increasing deviations are observed at higher prediction horizon. For the CA and CTRA model, the data shows an overly high peak around the mean in (a) which is not captured and hence the predicted percentile values in (c) are too conservative.



(a) Histograms.

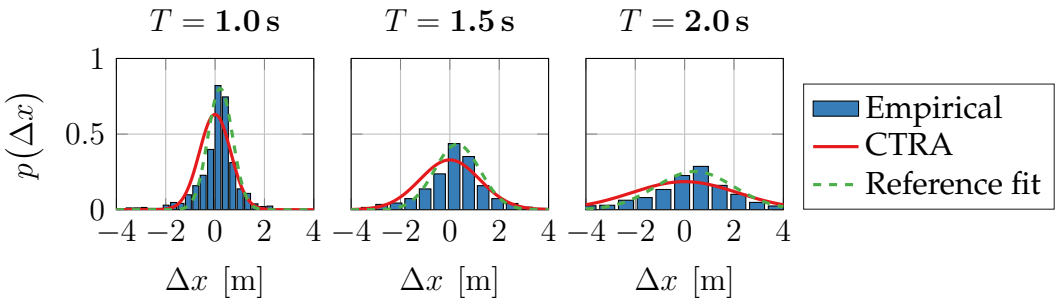


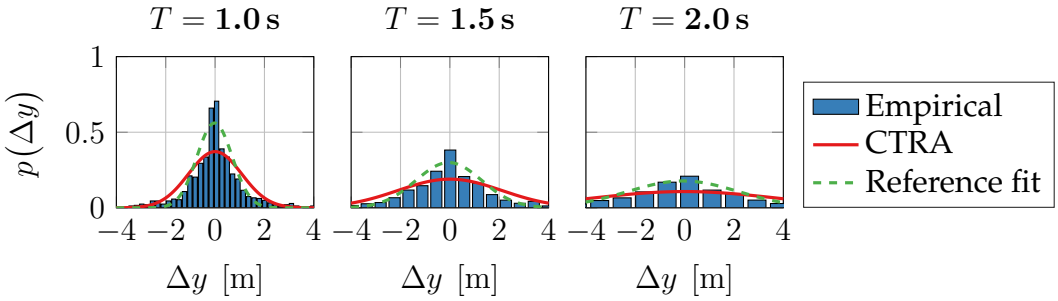
(b) Quantiles over model quantiles (range: 1% . . . 99%).



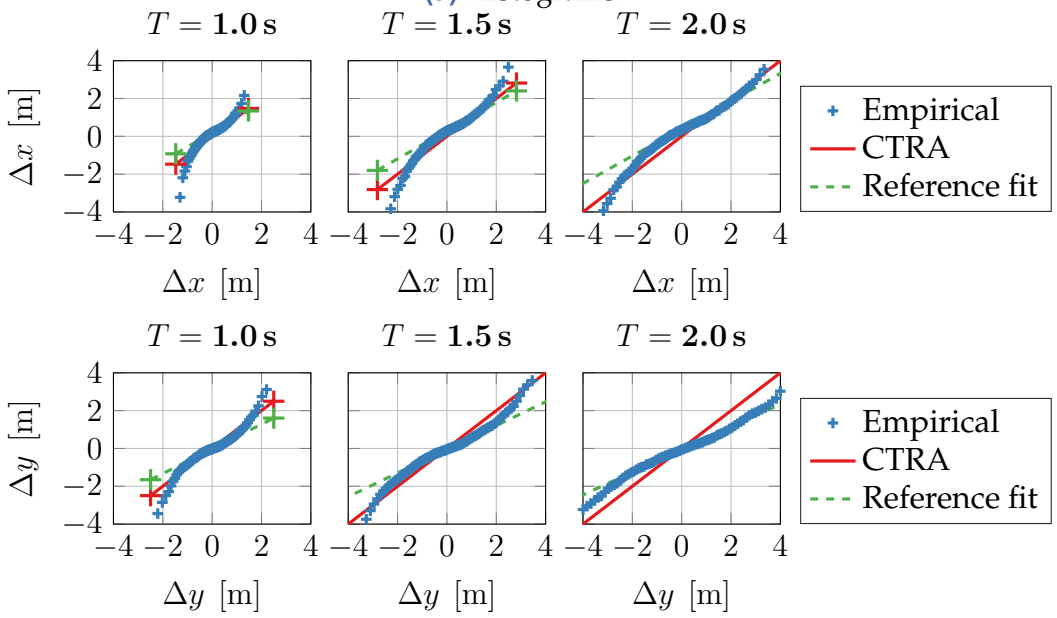
(c) 70% percentile values of absolute error.

Figure 4.18 **Straight trajectories:** Prediction errors in the longitudinal velocity v_x are compared to the prediction model distributions and a-posteriori Gaussian fits for different prediction times $T \in \{1\,\mathrm{s}, 1.5\,\mathrm{s}, 2\,\mathrm{s}\}$. A qualitative deviation from the theoretical evolution $\sigma_{\Delta v_x}(T) \propto \sqrt{T}$ can be observed for the CV model in (c).

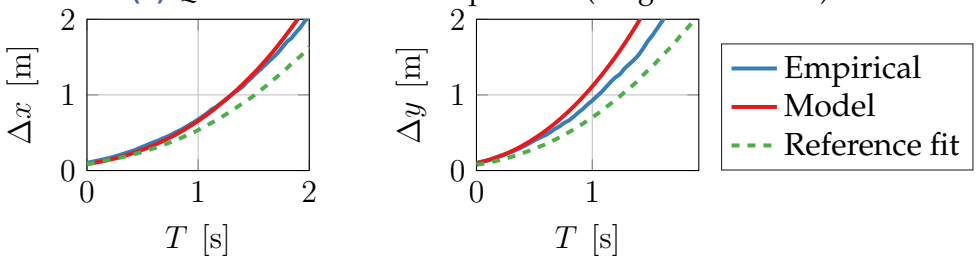




(a) Histograms.



(b) Quantiles over model quantiles (range: $1\% \ldots 99\%$).



(c) 70% percentile values of absolute error.

Figure 4.19 **Curved trajectories:** Prediction errors in longitudinal (x) and lateral (y) direction are compared to the CTRA prediction model distributions and a-posteriori Gaussian fits for different prediction times $T \in \{1\,\mathrm{s}, 1.5\,\mathrm{s}, 2\,\mathrm{s}\}$.

5 Modelling of algorithms for situation interpretation

5.1 Manoeuvre recognition for long-term predictions

Realising sophisticated driver assistance functions up to fully automated vehicles requires long-term trajectory predictions of other road users. How to obtain such predictions is an active research topic which is tackled from different directions. One commonly employed possibility is to extend the prediction models from Sec. 4.2 with an additional semantic layer at which high-level driver intentions are described. Thus, discrete driver intentions have to be inferred from measurement information. This section focusses on the statistical properties of this manoeuvre recognition problem.

After an introduction to the problem in Sec. 5.1.1, an overview of the diverse approaches for manoeuvre recognition is first given in Sec. 5.1.2. This leads to a specific formulation of the problem in Sec. 5.1.3 and thereafter the review of its optimal solution in Sec. 5.1.4. The statistical properties of this optimal solution constitute a performance bound for any algorithm which can be implemented in practice. As the main contribution, a novel recursive formulation of the bound is developed. The findings are illustrated with a numerical example in Sec. 5.1.5 and summarised in Sec. 5.1.6.

5.1.1 Introduction

In the previous Sec. 4.2, kinematic motion models for predicting a vehicle trajectory have been analysed. These models define a trajectory in the form of interpretable physical quantities. While a purely kinematic prediction is reasonable for short time spans, for example in collision avoidance systems, it generally lacks the ability to predict a sequence of changing driver inputs. For instance, a lane change manoeuvre consists of two consecutive driver actions, each of them possibly influenced by the presence of other vehicles. Considering only the kinematic quantities at one stage without knowledge of the future driver inputs will therefore lead to an erroneous prediction. It is thus necessary to employ knowledge on a driver's intentions in order to realise more sophisticated predictions.

Different approaches for driver intention recognition, for example pattern- or model-based algorithms, are reported in the literature. However, all methods have to rely on uncertain sensory measurements and model assumptions in order to infer the unobservable plans of a human driver. The goal of this section is to contribute to the understanding of how uncertainties affect the detection of an intended manoeuvre. As it has been highlighted in the context of the Bayes filter in Sec. 4.1, an optimal solution to statistical estimation problems is seldom available in closed-form. Nevertheless, theoretical upper performance bounds are a valuable means for drawing generic conclusions, for example for the evaluation of a sub-optimal solution. The focus of this section is to model the best attainable performance for the detection of discrete manoeuvres.

Major parts of the remainder of this chapter have been previously published in [219].

5.1.2 Related work

Analytical models or performance bounds for the task of manoeuvre recognition for intelligent vehicles have to the best of the author's knowledge not been discussed in the literature so far. The performance of an algorithm is often evaluated in simulations and thus, no analytical insight on how the individual components affect the results is obtained. One possible explanation for the lack of an established theory of performance bounds lies in the diversity of approaches. An overview of recent works will be given in the following. A differentiation is made between motion pattern based (Sec. 5.1.2.1) and intention-based (Sec. 5.1.2.2) approaches [118]. Sec. 5.1.2.3 contains a discussion of the different methods and defines the scope this work's analysis.

5.1.2.1 Motion pattern based predictions

The core principle of motion pattern based predictions is that measurements of a trajectory up to the current time step are compared to a set of prototype trajectories that have been stored in a database.

In a learning phase, vehicle trajectories are recorded and stored. To decrease the dimensionality of the data, a trajectory can be represented by the coefficients of an approximating Chebyshev polynomial [76, 200]. Trajectories can be stored individually or in aggregate form as clusters [192] or variational Gaussian mixture models [167, 200].

When the system is operating, a partially measured trajectory is compared to the database using a similarity metric. For example, the longest-common-subsequence method [148] or a likelihood value in the case of a probabilistic knowledge representations [98] can be employed.

One advantage of this method is that estimation and prediction are naturally connected since they only refer to different parts of the trajectory patterns, the observed and the future part. Once a trajectory pattern that matches the current measurements has been identified, the further trajectory evolution from the knowledge base can be used as a prediction.

5.1.2.2 Intention-based predictions

The basic idea of intention-based methods is to separate the prediction in high-level semantic aspects and low-level vehicle trajectories. A driver's intention to perform a specific manoeuvre is modelled at the semantic layer and has to be estimated first. Second, the corresponding trajectory can be calculated with a low-level kinematic model as introduced in Sec. 4.2. This distinguishes the approach from the matching of motion patterns where a data-based instead of a model-based trajectory representation is employed.

The main questions are how to define and parametrise a model of the semantic aspects and to establish the connection to the trajectory level. A differentiation of the driver behaviour into *goals*, *plans* and *behaviours* is suggested in [66]. Throughout this work, the term *manoeuvre* will be used to summarise the driver behaviour.

Diverse application goals and constraints lead to a variety of model structures which are reported in the literature. A model's properties affect the flexibility in describing different situations, how the model parameters can be identified and how the statistical properties can be derived.

In the following, several commonly employed forms of models are briefly summarised while a more exhaustive compilation is provided in [118]. The applications include predicting a lane change on a highway [42], recognising the intended direction at intersections [125, 199] or planning of safe trajectories in generic traffic situations with multiple road users [1, 66].

Bayesian networks (BN) represent conditional probabilities between random observables and latent variables in the form of a directed acyclic graph [56]. Given the known dependences between the different graph nodes, the probability density of unobservable variables can be inferred by measuring the realisations of other nodes in the graph.

For example, a BN can relate the occurrence of an *emergency braking situation* to the velocity with which another vehicle is approached. Given a measurement of this feature, one can infer the probability of an emergency braking situation from the network [171]. Further applications are reported for manoeuvre recognition in structured environments, like highways [42, 105, 174], and multimodal criticality assessment [173].

The advantage of a BN lies in its intuitive, graphical presentation. Moreover, it is shown in [171] how uncertainty in the feature measurements can be analytically propagated to the inference results. However, since the networks are static, they are not well suited to represent chains of events.

Dynamic Bayesian networks (DBN) extend static BNs to a time-dependent set of states. A DBN can be thought of as multiple BNs, one for each time step with an additional transition model [142]. This is a very generic concept and can therefore be used to describe complex traffic situations with interactions between multiple participants [1, 66, 117]. However, learning the model and estimating its state are challenging. Hidden Markov models and linear state space models are special cases which enable an efficient handling.

Hidden Markov models (HMM) describe the Markovian evolution of a *discrete* state variable in terms of a transition probability matrix. *Discrete* or *continuous* measurements are obtained according to conditional measurement distributions.

Multiple HMMs have to be defined in order to recognise different manoeuvres. Given a sequence of measurements, the most probable manoeuvre can be identified [56, 139]. Efficient algorithms exist for both learning and inference of HMMs. However, due to the discrete state space, continuous motion dynamics have to be approximated.

Linear state space models describe the time evolution of a *continuous* state by a linear stochastic difference equation as introduced in Sec. 4.1.3.3. A linear measurement model defines the probability distribution of a *continuous* observable conditional on the state.

Different manoeuvres are represented by different models and the goal is to infer each model's likelihood from observations. For example, a deterministic control input can be used to define a driver behaviour which is related to a specific intention [125, 180]. Specifically for highway scenarios, it is shown in [91] how the likelihood of a lane change manoeuvre can be estimated from the distance to the lane centres.

The advantages of this approach are that continuous motions are naturally modelled using physical relations. Furthermore, obtaining a trajectory prediction from the state transition function is straightforward once a manoeuvre hypothesis is known.

5.1.2.3 Discussion

Motion pattern based approaches are closely linked to a pre-defined database. The accuracy of a prediction depends on factors such as the data used for learning, the data representation and the similarity metric. This hinders to develop statistical models for generic conclusions. Thus, pattern-based methods are not considered in this work.

In the context of intention-based predictions, dynamic Bayesian networks with a mixed discrete and continuous state space are a very general class of models. However, their flexibility also increases the difficulty of deriving statistical properties. For instance, deriving bounds for the state estimation error in switching systems is still an active research topic [60].

Therefore, as a first step, linear state space models are used in this work to formulate different driving manoeuvres. Having a continuous state representation, these models can be easily linked to the previously introduced state estimation and trajectory prediction framework. However, continuous dynamics introduce limitations in the modelling of purely discrete clues. In these cases, HMMs, which rely on a discrete state space, would be better suited.[1] Going beyond the scope of this work, both types could be combined in a Bayesian approach.

5.1.3 Problem formulation

Based on the previous discussion, the linear state space model that is employed in this work will be detailed in Sec. 5.1.3.1. Subsequently, the intention recognition task is formalised as a hypothesis test in Sec. 5.1.3.2. Furthermore, an optimality criterion that is used for the derivation of performance bounds will be introduced.

[1] Nevertheless, linear state space models and HMMs are built on very similar assumptions, for example the Markov property. Hence, a transformation is possible with a suitable mapping between discrete and continuous states [209].

5.1.3.1 Model representation

Firstly, a linear transition model for the motion state $\mathbf{x}_k \in \mathbb{R}^n$ is assumed:

$$\mathbf{x}_{k+1} = \mathbf{A}\mathbf{x}_k + \mathbf{B}\mathbf{u}_k + \mathbf{w}_k \,, \qquad \mathbf{w}_k \sim p\left(\mathbf{w}_k\right) \,, \qquad \mathbb{E}\left[\mathbf{w}_k\right] = \mathbf{0} \,. \tag{5.1a}$$

In contrast to the model (4.26) as it is introduced for the state estimation task, $\mathbf{u}_k$ now stands for a deterministic but unknown control input signal which models the driver commands when a certain manoeuvre is executed. For example, the *Intelligent Driver Model* can be used to describe a driver's acceleration inputs when he intends to follow another vehicle or to stop at a certain position [125]. Particularly, a driver's intention to change the currently driven manoeuvre will result in changing control inputs.

In addition to the variability that is introduced by the possible realisations of the control input, stochastic variations are modelled with the process noise $\mathbf{w}_k$ in (5.1a). The case of a zero mean white non-Gaussian process which is characterised by the intrinsic accuracy $\boldsymbol{\mathcal{I}}_{\mathbf{w}}$ is assumed.

Detecting a deterministically changing input signal is a classical problem in supervision and fault detection[2] for technical processes [13, 74, 106, 214]. The central idea in this section is therefore to formalise a manoeuvre recognition task as a change detection problem. Thus, general results on the best attainable performance of a detection algorithm can be applied.

Even for the same manoeuvre, the corresponding sequence of driver inputs $\mathbf{u}_1, \ldots, \mathbf{u}_k$ is likely to vary among different driving styles and may depend on unobservable variables, for example the distance to a preceding vehicle. It is not important at this stage how such variations and dependencies are modelled in a specific application context. Instead, a generic formulation as proposed in [74] is assumed. It consists of known time-dependent basis functions $\boldsymbol{\Psi}_k$, for instance Chebyshev polynomials, and an unknown constant coefficient vector $\boldsymbol{\theta} \in \mathbb{R}^{n_\theta}$:

$$\mathbf{u}_k = \boldsymbol{\Psi}_k \cdot \boldsymbol{\theta} \,. \tag{5.1b}$$

Therefore, different manoeuvres lead to different sequences $\mathbf{u}_1^{(i)}, \ldots, \mathbf{u}_k^{(i)}$ of the driver input and are described by different coefficient vectors $\boldsymbol{\theta}_i$. The objective is thus to differentiate between the possible realisations of $\boldsymbol{\theta}$. For clarity, it will be assumed in the following that there are only two different manoeuvres, that is $i \in \{\, 0, 1 \,\}$.[3]

Having specified the state transition model, it remains to detail how measurements $\mathbf{y}_k \in \mathbb{R}^m$ are obtained. These form the basis of any estimator. A linear measurement model with additive zero mean white noise $\mathbf{v}_k$ with intrinsic accuracy $\boldsymbol{\mathcal{I}}_{\mathbf{v}}$ is assumed:

$$\mathbf{y}_k = \mathbf{C}\mathbf{x}_k + \mathbf{D}\mathbf{u}_k + \mathbf{v}_k \,, \qquad \mathbf{v}_k \sim p\left(\mathbf{v}_k\right) \,, \qquad \mathbb{E}\left[\mathbf{v}_k\right] = \mathbf{0} \,. \tag{5.1c}$$

[2] In the context of fault detection, the *unknown* input is usually denoted by $\mathbf{f}_k$, to differentiate it from an additional *known* control input $\mathbf{u}_k$. Here, $\mathbf{u}_k$ will be used exclusively to be aligned with the notation in other parts of the thesis.

[3] Extending models of a binary hypothesis test problem to M hypotheses is discussed in [106].

5.1.3.2 Formal problem definition

Given a sequence of L noisy measurements $\mathbf{y}_{k-L+1}, \dots, \mathbf{y}_k$ and an initial estimate of the state $\hat{\mathbf{x}}_{k-L+1}$ at time $k-L$, the objective of a manoeuvre detection algorithm is to decide if a change in the input signal $\mathbf{u}_k$ has occurred.

Formally, this means to decide between two hypotheses $\mathcal{H}_i$, $i \in \{0, 1\}$ where in the following, $\mathcal{H}_0$ refers to the current and $\mathcal{H}_1$ to the alternative manoeuvre. These hypotheses correspond to different probability distributions from which the measurements originate. The complexity of the hypothesis test problem depends on the available knowledge about these distributions [106]:

1. In a *simple hypothesis* test problem, the probability densities are fully specified:

$$\begin{cases} \mathcal{H}_0: & \boldsymbol{\theta} = \boldsymbol{\theta}_0 \\ \mathcal{H}_1: & \boldsymbol{\theta} = \boldsymbol{\theta}_1 \end{cases}. \tag{5.2}$$

   Under these conditions, an optimal decision rule is provided by the *Likelihood ratio test* which will be discussed in Sec. 5.1.4.2.

2. A *composite hypothesis* test problem on the other hand refers to the case where unknown parameters are part of the hypotheses. Specifically the case where $\boldsymbol{\theta}$ is unknown under $\mathcal{H}_1$ will be considered:

$$\begin{cases} \mathcal{H}_0: & \boldsymbol{\theta} = \boldsymbol{\theta}_0 \\ \mathcal{H}_1: & \boldsymbol{\theta} \neq \boldsymbol{\theta}_0 \end{cases}. \tag{5.3}$$

   For example, $\mathcal{H}_1$ may represent a braking manoeuvre in contrast to keeping the current velocity under $\mathcal{H}_0$. However, different drivers and situations will lead to varying characteristics of the vehicle's deceleration. Therefore, the value of $\boldsymbol{\theta}_1$ is not known beforehand.

   Unfortunately, no generally optimal detector exists for composite hypothesis problems [106]. An approach with asymptotically favourable performance, known as the *Generalised likelihood ratio test*, will be studied in Sec. 5.1.4.3.

In order to derive a performance bound for this task, the metrics and definition of optimality have to be clarified first. In the case of a binary hypothesis test, where an estimate $\hat{\mathcal{H}}$ with only two possible outcomes is obtained, the probability of making an incorrect decision can be used. This is expressed by the true positive detection probability $P_\mathrm{D} \coloneqq \mathrm{P}\left(\hat{\mathcal{H}} = \mathcal{H}_1 \middle| \mathcal{H}_1\right)$ and the false alarm probability $P_\mathrm{FA} \coloneqq \mathrm{P}\left(\hat{\mathcal{H}} = \mathcal{H}_1 \middle| \mathcal{H}_0\right)$.

In practice, these two probabilities are coupled and it is therefore not possible to improve the true positive rate without making the sacrifice of an increased chance of false alarms. This relationship is usually visualised in the form of a diagram of P_D over P_FA which is known as the receiver operating characteristic (ROC) of a detector.

One property, which is considered as an optimality criterion here, denotes a hypothesis test as being *uniformly most powerful* if it features the highest P_D at a specific level P_FA

among all possible tests. A performance bound for the described hypothesis test problem should therefore define a curve $P_{\mathrm{D}}\left(P_{\mathrm{FA}}\right)$ which encloses the ROC curves of any detection algorithm [106].

5.1.4 Statistical modelling of optimal hypothesis tests

After the manoeuvre recognition task has been formulated in the framework of a hypothesis test problem, this section develops statistical models of the optimal detector. Following the approach in [74], a batch-matrix formulation is first introduced in Sec. 5.1.4.1. Known statistical models for the case of two simple hypotheses are stated in Sec. 5.1.4.2 and for the combination of a simple and a composite hypothesis in Sec. 5.1.4.3. As the central contribution of this section, a novel recursive derivation of the statistical models is developed in Sec. 5.1.4.4.

5.1.4.1 Transformation to a general linear model

The system (5.1) has been defined recursively. Therefore, the probability densities which relate a series $\mathbf{y}_{k-L+1:k}$ of L measurements to the unknown manoeuvre parameter $\boldsymbol{\theta}$ are only known per time step k through the probabilistic measurement model $p\left(\mathbf{y}_k\middle|\mathbf{x}_k\right)$. The unobservable state $\mathbf{x}_k$ evolves according to the system state transition density $p\left(\mathbf{x}_{k+1}\middle|\mathbf{x}_k,\boldsymbol{\theta}\right)$ from (5.1a). In order to apply fundamental results on general hypothesis test problems, it is beneficial to concatenate the individual time steps and rewrite the problem in the form of a general linear model [106]:

$$\mathbf{R}_L = \boldsymbol{\Phi}_L\boldsymbol{\theta} + \mathbf{E}_L \ . \tag{5.4}$$

Here, $\mathbf{R}_L$ denotes a vector of the measurement data, $\mathbf{E}_L$ a random vector of zero mean noise and $\boldsymbol{\Phi}_L$ an observation matrix with full rank of appropriate dimensions.

To achieve a model in the form of (5.4) the following notations are proposed in [74]. First, the relevant signals from the time steps $k-L+1,\ldots,k$ are written as stacked vectors:

$$\mathbf{X}_L \coloneqq \begin{bmatrix}\mathbf{x}_{k-L+1}\\ \vdots \\ \mathbf{x}_k\end{bmatrix}, \quad \mathbf{W}_L \coloneqq \begin{bmatrix}\mathbf{w}_{k-L+1}\\ \vdots \\ \mathbf{w}_k\end{bmatrix}, \quad \mathbf{Y}_L \coloneqq \begin{bmatrix}\mathbf{y}_{k-L+1}\\ \vdots \\ \mathbf{y}_k\end{bmatrix}, \quad \mathbf{V}_L \coloneqq \begin{bmatrix}\mathbf{v}_{k-L+1}\\ \vdots \\ \mathbf{v}_k\end{bmatrix}. \tag{5.5}$$

Moreover, the following matrices are defined from the system matrices[4] in (5.1):

$$\boldsymbol{\mathcal{O}}_L \coloneqq \begin{bmatrix}\mathbf{C}^{\mathrm{T}} & \left(\mathbf{CA}\right)^{\mathrm{T}} & \ldots & \left(\mathbf{CA}^{L-1}\right)^{\mathrm{T}}\end{bmatrix}^{\mathrm{T}}, \tag{5.6a}$$

[4] In order to simplify the notation, time-invariant system matrices are assumed in (5.1). Nevertheless, the case of a time-dependent system can be treated accordingly and further details are given in [74].

$$\mathbf{H}_L \coloneqq \begin{bmatrix} \mathbf{C} & \mathbf{0} & \dots & \mathbf{0} \\ \mathbf{0} & \mathbf{C} & \ddots & \vdots \\ \vdots & \ddots & \ddots & \mathbf{0} \\ \mathbf{0} & \mathbf{0} & \dots & \mathbf{C} \end{bmatrix}, \qquad \mathbf{H}_L^{\mathbf{w}} \coloneqq \begin{bmatrix} \mathbf{0} & \mathbf{0} & \dots & \mathbf{0} & \mathbf{0} \\ \mathbf{C} & \mathbf{0} & \ddots & \vdots & \vdots \\ \vdots & \ddots & \ddots & \mathbf{0} & \mathbf{0} \\ \mathbf{C}\mathbf{A}^{L-2} & \mathbf{C}\mathbf{A}^{L-3} & \dots & \mathbf{C} & \mathbf{0} \end{bmatrix}, \tag{5.6b}$$

$$\mathbf{\Phi}_L \coloneqq \begin{bmatrix} \boldsymbol{\varphi}_1 & \dots & \boldsymbol{\varphi}_L \end{bmatrix}^{\mathrm{T}}, \qquad \boldsymbol{\varphi}_l^{\mathrm{T}} \coloneqq \mathbf{D}\mathbf{\Psi}_{k-L+l} + \sum\nolimits_{j=1}^{l-1} \mathbf{C}\mathbf{A}^{l-j-1}\mathbf{B}\mathbf{\Psi}_{k-L+j}\,. \tag{5.6c}$$

$\boldsymbol{\mathcal{O}}_L$ is the observability matrix of the system (5.1), $\mathbf{H}_L$ relates the stacked measurements $\mathbf{Y}_L$ to the states $\mathbf{X}_L$ and the Toeplitz matrix $\mathbf{H}_L^{\mathbf{w}}$ describes the impact of the process noise $\mathbf{W}_L$ on $\mathbf{Y}_L$.

Lastly, we suppose that the initial state estimate $\hat{\mathbf{x}}_{k-L+1}$ is affected by a zero mean estimation error $\Delta\mathbf{x}_{k-L+1}$. This error is assumed as independent of all other noise processes, for example because it has been estimated using data from outside of the now considered window $\{\, k-L+1, \dots, k \,\}$. Its intrinsic accuracy is denoted by $\boldsymbol{\mathcal{I}}_{\mathbf{x}_{k-L+1}}$.

These definitions allow (5.1) to be recast in the shape of (5.4):

$$\underbrace{\mathbf{Y}_L - \boldsymbol{\mathcal{O}}_L\hat{\mathbf{x}}_{k-L+1}}_{=:\mathbf{R}_L} = \mathbf{\Phi}_L\boldsymbol{\theta} + \underbrace{\boldsymbol{\mathcal{O}}_L\Delta\mathbf{x}_{k-L+1} + \mathbf{H}_L^{\mathbf{w}}\mathbf{W}_L + \mathbf{V}_L}_{=:\mathbf{E}_L}\,. \tag{5.7}$$

All noise terms have been collected on the right-hand side of the equation. Thus, the probability density $p\big(\mathbf{R}_L\big|\boldsymbol{\theta}\big)$ of the measured residuals $\mathbf{R}_L$ conditional on the unknown manoeuvre parameter $\boldsymbol{\theta}$ can be obtained from the individual noise densities. This density is crucial for the definition of a sensible rule to distinguish between $\mathcal{H}_0$ and $\mathcal{H}_1$ based on a sample of measurements $\mathbf{R}_L$.

5.1.4.2 Simple hypotheses: Likelihood ratio test

At first, we consider the case where the unknown parameter can take on two known values $\boldsymbol{\theta} \in \{\, \boldsymbol{\theta}_0, \boldsymbol{\theta}_1 \,\}$. According to the Neyman-Pearson theorem, the decision rule of the most powerful test is then based on the likelihood ratio $L\left(\mathbf{R}_L\right)$ [106]:

$$L\left(\mathbf{R}_L\right) \coloneqq \frac{p\big(\mathbf{R}_L\big|\boldsymbol{\theta}_1\big)}{p\big(\mathbf{R}_L\big|\boldsymbol{\theta}_0\big)} \underset{\mathcal{H}_0}{\overset{\mathcal{H}_1}{\gtrless}} \gamma\,. \tag{5.8}$$

Here, γ denotes a user-defined threshold value. The true and false detection probabilities of this test are:

$$P_{\mathrm{D}}\left(\gamma\right) = \mathrm{P}\big(L\left(\mathbf{R}_L\right) > \gamma\big|\boldsymbol{\theta}_1\big)\,, \qquad P_{\mathrm{FA}}\left(\gamma\right) = \mathrm{P}\big(L\left(\mathbf{R}_L\right) > \gamma\big|\boldsymbol{\theta}_0\big)\,. \tag{5.9}$$

It depends on the involved densities, how the likelihood ratio (LR) and the properties of this optimal test can be calculated. For the example of Gaussian noise, this will be further detailed in the following.

Example 5.1 (Likelihood ratio test for linear model with Gaussian noise)

We assume that the noise distributions are zero mean Gaussians, that is $\mathbf{w}_k \sim \mathcal{N}(\mathbf{0}, \mathbf{Q})$, $\mathbf{v}_k \sim \mathcal{N}(\mathbf{0}, \mathbf{R})$ and $\Delta\mathbf{x}_{k-L+1} \sim \mathcal{N}\left(\mathbf{0}, \mathbf{\Sigma}_{\mathbf{x}_{k-L+1}}\right)$. Since the noise is assumed independent between different time steps, the covariances $\mathbf{\Sigma}_{\mathbf{W}_L}$ and $\mathbf{\Sigma}_{\mathbf{V}_L}$ of the stacked noise vectors from (5.5) have a block diagonal form. Then, $\mathbf{E}_L$ in (5.7) is a linear combination of zero mean Gaussian random variables and thus a zero mean Gaussian as well with the covariance

$$\mathbf{\Sigma}_{\mathbf{E}_L} = \boldsymbol{\mathcal{O}}_L \mathbf{\Sigma}_{\mathbf{x}_{k-L+1}} \boldsymbol{\mathcal{O}}_L^{\mathrm{T}} + \mathbf{H}_L^{\mathbf{w}} \mathbf{\Sigma}_{\mathbf{W}_L} \mathbf{H}_L^{\mathbf{w}\mathrm{T}} + \mathbf{\Sigma}_{\mathbf{V}_L} \,. \tag{5.10}$$

According to (5.8), the optimal decision rule can be established. For simplified expressions, the log-likelihood ratio $L'(\mathbf{R}_L) := 2 \log L(\mathbf{R}_L)$ is considered instead of $L(\mathbf{R}_L)$ [106]:

$$\begin{aligned} L'(\mathbf{R}_L) &= 2 \log \frac{\mathcal{N}\left(\mathbf{R}_L; \mathbf{\Phi}_L \boldsymbol{\theta}_1, \mathbf{\Sigma}_{\mathbf{E}_L}\right)}{\mathcal{N}\left(\mathbf{R}_L; \mathbf{\Phi}_L \boldsymbol{\theta}_0, \mathbf{\Sigma}_{\mathbf{E}_L}\right)} \\ &= \left(\mathbf{\Phi}_L \boldsymbol{\theta}_0\right)^{\mathrm{T}} \mathbf{\Sigma}_{\mathbf{E}_L}^{-1} \left(\mathbf{\Phi}_L \boldsymbol{\theta}_0\right) - \left(\mathbf{\Phi}_L \boldsymbol{\theta}_1\right)^{\mathrm{T}} \mathbf{\Sigma}_{\mathbf{E}_L}^{-1} \left(\mathbf{\Phi}_L \boldsymbol{\theta}_1\right) \\ &\quad + 2 \left(\boldsymbol{\theta}_1 - \boldsymbol{\theta}_0\right)^{\mathrm{T}} \mathbf{\Phi}_L^{\mathrm{T}} \mathbf{\Sigma}_{\mathbf{E}_L}^{-1} \cdot \mathbf{R}_L \end{aligned} \,. \tag{5.11}$$

The first two terms are constant whereas the actual dependence on the measured residuals $\mathbf{R}_L$ is given by the third term.

The statistical properties of this decision rule are defined by the probability distributions of $L'(\mathbf{R}_L)$ under the two possible realisations of $\boldsymbol{\theta}$. It follows from (5.7) that $\mathbf{R}_L \sim \mathcal{N}\left(\mathbf{\Phi}_L \boldsymbol{\theta}_1, \mathbf{\Sigma}_{\mathbf{E}_L}\right)$ under $\mathcal{H}_1$ and otherwise $\mathbf{R}_L \sim \mathcal{N}\left(\mathbf{\Phi}_L \boldsymbol{\theta}_0, \mathbf{\Sigma}_{\mathbf{E}_L}\right)$. Since (5.11) is a linear transformation of the Gaussian random variable $\mathbf{R}_L$, the distribution of $L'(\mathbf{R}_L)$ remains Gaussian:

$$L'(\mathbf{R}_L) \sim \begin{cases} \mathcal{N}(\lambda_L, 4\lambda_L) & \text{under } \mathcal{H}_1 \\ \mathcal{N}(-\lambda_L, 4\lambda_L) & \text{under } \mathcal{H}_0 \end{cases} \tag{5.12a}$$

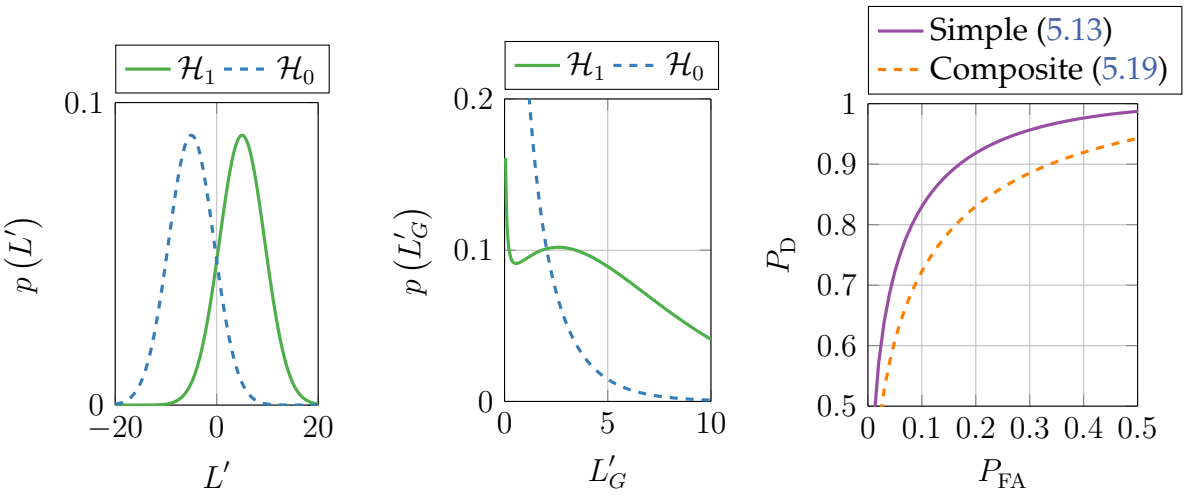
$$\text{where } \lambda_L = \left(\boldsymbol{\theta}_1 - \boldsymbol{\theta}_0\right)^{\mathrm{T}} \mathbf{\Phi}_L^{\mathrm{T}} \mathbf{\Sigma}_{\mathbf{E}_L}^{-1} \mathbf{\Phi}_L \left(\boldsymbol{\theta}_1 - \boldsymbol{\theta}_0\right) \,. \tag{5.12b}$$

The two distributions are symmetric around $L'(\mathbf{R}_L) = 0$ and further apart for higher values of λ, which facilitates detecting the correct hypothesis. For an exemplary case, the probability density functions are illustrated in Fig. 5.1(a).

Furthermore, it is possible to obtain a closed-form expression for the ROC $P_{\mathrm{D}}(P_{\mathrm{FA}})$. To this end, the cumulative probability function of the normal distribution is denoted by $P_{\mathcal{N}\left(\mu,\sigma^2\right)}(\cdot)$ and the quantile function, that is the inverse cumulative probability function, as $Q_{\mathcal{N}\left(\mu,\sigma^2\right)}(\cdot)$. Inserting (5.12) into (5.9) yields:

$$P_{\mathrm{D}}(P_{\mathrm{FA}}) = 1 - P_{\mathcal{N}(\lambda_L, 4\lambda_L)}\left(Q_{\mathcal{N}(-\lambda_L, 4\lambda_L)}\left(1 - P_{\mathrm{FA}}\right)\right) \,. \tag{5.13}$$

An exemplary case is visualised in Fig. 5.1(c).



(a) Simple hypotheses. **(b)** Composite hypotheses. **(c)** ROC

Figure 5.1 Illustration of the test statistics for two simple hypotheses with Gaussian noise (5.12) and the asymptotic distribution (5.18) of the composite hypothesis test. An exemplary parametrisation with $\lambda_L = 5$ and $n_\theta = 1$ is chosen. For a specific value of the detector threshold γ, the resulting probabilities of true positive detections and false alarms can be calculated by integrating the probability densities in (a)-(b) according to (5.9). This leads to the ROC curves in (c). For an identical parametrisation, deciding between two simple hypotheses allows better detection results due to the reduced degrees of freedoms.

When deciding between two simple hypotheses, (5.8) provides a very generic recipe on how to find an optimal decision rule. In the more complex case of composite hypotheses from (5.3), a similar result does not exist and optimality is in general only attained asymptotically. This will be further elaborated in the following section.

5.1.4.3 Composite hypotheses: Generalised likelihood ratio test

While the probability density $p(\mathbf{R}_L|\boldsymbol{\theta})$ is completely specified for simple hypotheses, additional unknown parameters are included in composite hypothesis test problem. In general, the optimality of a decision rule depends on the value of these parameters. A test whose optimality is invariant against the parameter is therefore *uniformly* most powerful. Unfortunately, such tests only exist under special conditions. However, the *Generalised likelihood ratio* (GLR) test delivers this favourable property in an asymptotic sense (for an infinite number of measurements) and for different families of probability distributions (for example Gaussian distributions, as it will be shown in example 5.2) [106].

The GLR test generalises the test statistic (5.8) of the LR test by replacing the unknown parameters $\boldsymbol{\theta}_0$, $\boldsymbol{\theta}_1$ with their maximum likelihood estimates $\hat{\boldsymbol{\theta}}_0$ and $\hat{\boldsymbol{\theta}}_1$ [106]:

$$L_{\mathrm{G}}(\mathbf{R}_L) := \frac{p(\mathbf{R}_L|\hat{\boldsymbol{\theta}}_1)}{p(\mathbf{R}_L|\hat{\boldsymbol{\theta}}_0)} \underset{\mathcal{H}_0}{\overset{\mathcal{H}_1}{\gtrless}} \gamma \,. \tag{5.14}$$

In the following, the special case where $\mathcal{H}_0 : \boldsymbol{\theta} = \boldsymbol{\theta}_0$ is fully specified according to (5.3) will be analysed. In the more general case where $\boldsymbol{\theta}_0$ is unknown as well, the GLR test (5.14) is still applicable but no general result on the statistical properties of the decision rule is available [106].

In practice, implementing the GLR test can be facilitated by alternative decision rules known as the *Wald test* or *Rao test*. These tests provide the same asymptotic performance as (5.14) [106]. The Wald test will be detailed in the following.

The basic idea of the Wald test is to calculate a maximum likelihood estimate of the parameter $\boldsymbol{\theta}$ under $\mathcal{H}_1$ from the measured residuals $\mathbf{R}_L$:

$$\hat{\boldsymbol{\theta}} = \arg\max_{\boldsymbol{\theta}} p\left(\mathbf{R}_L \middle| \boldsymbol{\theta}\right) , \qquad \boldsymbol{\Sigma}_{\hat{\boldsymbol{\theta}}} = \left(\boldsymbol{\Phi}_L \boldsymbol{\mathcal{I}}_{\mathbf{E}_L} \boldsymbol{\Phi}_L^{\mathrm{T}}\right)^{-1} . \tag{5.15}$$

Here, $\boldsymbol{\Sigma}_{\hat{\boldsymbol{\theta}}}$ denotes the asymptotic covariance of the maximum likelihood estimate and $\boldsymbol{\mathcal{I}}_{\mathbf{E}_L}$ is the intrinsic accuracy of the noise vector $\mathbf{E}_L$ [74][5]:

$$\boldsymbol{\mathcal{I}}_{\mathbf{E}_L} = \left(\boldsymbol{\mathcal{O}}_L \boldsymbol{\mathcal{I}}_{\mathbf{x}_{k-L+1}}^{-1} \boldsymbol{\mathcal{O}}_L^{\mathrm{T}} + \mathbf{H}_L^{\mathbf{w}} \boldsymbol{\mathcal{I}}_{\mathbf{W}_L}^{-1} \mathbf{H}_L^{\mathbf{w}\mathrm{T}} + \boldsymbol{\mathcal{I}}_{\mathbf{V}_L}^{-1}\right)^{-1} . \tag{5.16}$$

The decision rule is then to compare the Mahalanobis norm of $\hat{\boldsymbol{\theta}} - \boldsymbol{\theta}_0$ against a threshold $\gamma > 0$. Ideally, the norm becomes zero under $\mathcal{H}_0$ where $\boldsymbol{\theta} = \mathbf{0}$:

$$L_{\mathrm{W}}\left(\mathbf{R}_L\right) = \left(\hat{\boldsymbol{\theta}} - \boldsymbol{\theta}_0\right)^{\mathrm{T}} \boldsymbol{\Sigma}_{\hat{\boldsymbol{\theta}}}^{-1} \left(\hat{\boldsymbol{\theta}} - \boldsymbol{\theta}_0\right) \underset{\mathcal{H}_0}{\overset{\mathcal{H}_1}{\gtrless}} \gamma . \tag{5.17}$$

Quadratic forms of Gaussian random variables follow the χ^2-distribution. Therefore, depending on the true hypothesis, $L_{\mathrm{W}}\left(\mathbf{R}_L\right)$ is distributed according to a central or non-central χ^2-distribution [106]:

$$L'_{\mathrm{G}}\left(\mathbf{R}_L\right) \coloneqq 2 \log L_{\mathrm{G}}\left(\mathbf{R}_L\right) = L_{\mathrm{W}}\left(\mathbf{R}_L\right) \overset{\text{asymp.}}{\sim} \begin{cases} \chi^2_{n_\theta}\left(\lambda_L\right) & \text{under } \mathcal{H}_1 \\ \chi^2_{n_\theta}\left(0\right) & \text{under } \mathcal{H}_0 \end{cases} \tag{5.18a}$$

$$\text{where } \lambda_L = \left(\boldsymbol{\theta}_1 - \boldsymbol{\theta}_0\right)^{\mathrm{T}} \boldsymbol{\Phi}_L^{\mathrm{T}} \boldsymbol{\mathcal{I}}_{\mathbf{E}_L} \boldsymbol{\Phi}_L \left(\boldsymbol{\theta}_1 - \boldsymbol{\theta}_0\right) . \tag{5.18b}$$

The non-centrality parameter λ_L, which is in a fact a generalisation of (5.12b), can be interpreted as a measure of how different the distributions of $L'_{\mathrm{G}}\left(\mathbf{R}_L\right)$ under the two hypotheses are. Finding the correct distinction becomes easier for higher λ_L because this separates the non-central $\chi^2_{n_\theta}\left(\lambda_L\right)$-distribution further from $\chi^2_{n_\theta}\left(0\right)$. An exemplary case is shown in Fig. 5.1(b).

Interestingly, the probability of a false alarm P_{FA}, that is the detection threshold γ is exceeded under $\mathcal{H}_0$, remains independent of any system parameter. Therefore, a system improvement only affects the true positive detection probability P_{D}. The false alarm probability is thus only indirectly reduced by choosing a higher threshold value γ which achieves the same P_{D} but fewer false alarms.

[5] Note that assuming equality in (5.16) is a best case approximation and thus leads to an upper performance estimate. Equality holds for example in the case of Gaussian noise [193].

With the respective cumulative distribution and quantile functions, an expression for the ROC is obtained as

$$P_{\mathrm{D}}\left(P_{\mathrm{FA}}\right) = 1 - P_{\chi^2_{n_\theta}(\lambda_L)}\left(Q_{\chi^2_{n_\theta}}\left(1 - P_{\mathrm{FA}}\right)\right) . \tag{5.19}$$

A comparison of this expression to the previous result (5.13) in Fig. 5.1(c) illustrates that the composite hypothesis test problem is in fact more difficult to solve than the case of two simple hypotheses.

Example 5.2 (Generalised likelihood ratio test for linear model with Gaussian noise)
In the special case of Gaussian noise $\mathbf{E}_L \sim \mathcal{N}\left(\mathbf{0}, \mathbf{\Sigma}_{\mathbf{E}_L}\right)$ in (5.4), the maximum likelihood estimate (5.15) is given by the Generalised least squares (GLS) estimator [108]:

$$\hat{\boldsymbol{\theta}} = \left(\mathbf{\Phi}_L^{\mathrm{T}} \mathbf{\Sigma}_{\mathbf{E}_L}^{-1} \mathbf{\Phi}_L\right)^{-1} \mathbf{\Phi}_L^{\mathrm{T}} \mathbf{\Sigma}_{\mathbf{E}_L}^{-1} \mathbf{R}_L , \qquad \mathbf{\Sigma}_{\hat{\boldsymbol{\theta}}} = \left(\mathbf{\Phi}_L^{\mathrm{T}} \mathbf{\Sigma}_{\mathbf{E}_L}^{-1} \mathbf{\Phi}_L\right)^{-1} . \tag{5.20}$$

Alternatively, a recursive algorithm based on a Kalman filter is proposed by WILLSKY and JONES [201].

Inserting (5.20) into (5.17) yields for the Wald test decision rule:

$$L_{\mathrm{W}}\left(\mathbf{R}_L\right) = \mathrm{l}_{\mathrm{W}}^{\mathrm{T}} \cdot \left(\mathbf{\Phi}_L^{\mathrm{T}} \mathbf{\Sigma}_{\mathbf{E}_L}^{-1} \mathbf{\Phi}_L\right)^{-1} \cdot \mathrm{l}_{\mathrm{W}} , \quad \mathrm{l}_{\mathrm{W}} \coloneqq \mathbf{\Phi}_L^{\mathrm{T}} \mathbf{\Sigma}_{\mathbf{E}_L}^{-1} \mathbf{R}_L - \mathbf{\Phi}_L^{\mathrm{T}} \mathbf{\Sigma}_{\mathbf{E}_L}^{-1} \mathbf{\Phi}_L \boldsymbol{\theta}_0 . \tag{5.21}$$

Since $\mathbf{E}_L$ is Gaussian, the distribution of l_{W} is:

$$\mathrm{l}_{\mathrm{W}} \sim \begin{cases} \mathcal{N}\left(\mathbf{\Phi}_L^{\mathrm{T}} \mathbf{\Sigma}_{\mathbf{E}_L}^{-1} \mathbf{\Phi}_L \left(\boldsymbol{\theta}_1 - \boldsymbol{\theta}_0\right), \mathbf{\Phi}_L^{\mathrm{T}} \mathbf{\Sigma}_{\mathbf{E}_L}^{-1} \mathbf{\Phi}_L\right) & \text{under } \mathcal{H}_1 \\ \mathcal{N}\left(\mathbf{0}, \mathbf{\Phi}_L^{\mathrm{T}} \mathbf{\Sigma}_{\mathbf{E}_L}^{-1} \mathbf{\Phi}_L\right) & \text{under } \mathcal{H}_0 \end{cases} . \tag{5.22}$$

Thus, $L_{\mathrm{W}}\left(\mathbf{R}_L\right)$ in (5.21) is the Mahalanobis norm of the Gaussian random vector l_{W}. The distribution thus follows as [106]:

$$L_{\mathrm{W}} \sim \begin{cases} \chi^2_{n_\theta}\left(\lambda_L\right) & \text{under } \mathcal{H}_1 \\ \chi^2_{n_\theta}\left(0\right) & \text{under } \mathcal{H}_0 \end{cases} , \quad \lambda_L = \left(\boldsymbol{\theta}_1 - \boldsymbol{\theta}_0\right)^{\mathrm{T}} \mathbf{\Phi}_L^{\mathrm{T}} \mathbf{\Sigma}_{\mathbf{E}_L}^{-1} \mathbf{\Phi}_L \left(\boldsymbol{\theta}_1 - \boldsymbol{\theta}_0\right) . \tag{5.23}$$

Note that while this result is included in the general case given in (5.18), the derivation does not require the assumption of asymptotic Gaussianity of an MLE. Therefore, (5.23) holds for arbitrary window length L.

Although all relevant information on the asymptotic properties of the test is contained in the scalar λ_L from (5.18b), one has to obtain the intrinsic accuracy $\boldsymbol{\mathcal{I}}_{\mathbf{E}_L}$ first. According to (5.16), this requires inverting a matrix of dimensions $L \cdot m \times L \cdot m$. It will be analysed in the following section, how $\boldsymbol{\mathcal{I}}_{\mathbf{E}_{L+1}}$ for $L+1$ measurements relates to $\boldsymbol{\mathcal{I}}_{\mathbf{E}_L}$. This leads to a recursive expression for λ_{L+1} where the increment $\Delta\lambda_{L+1}$ added by the $(L+1)$st measurement is identified.

5.1.4.4 Deriving a recursive form of the GLR test distribution

Decision making in a driver assistance context is often very time critical. Thus, improving the certainty of an estimate by additional measurements comes at the price of a decreased remaining reaction time. Being based on a batch matrix notation as in [74], the results from the previous section do not provide an intuitive insight into how the number of measurements L influences the detector performance. It is therefore desirable to obtain an equivalent recursive model, similar to the Cramér-Rao bound for state estimation problems presented in Sec. 4.1.4.

The objective of this section is thus to derive a recursive expression for the non-centrality parameter λ_L from (5.18b). Especially the matrix inversion necessary to find the information matrix $\boldsymbol{\mathcal{I}}_{\mathbf{E}_L}$ in (5.16) is to be avoided. Due to the assumption of white noise processes it follows that $\boldsymbol{\mathcal{I}}_{\mathbf{W}_L}^{-1}$ and $\boldsymbol{\mathcal{I}}_{\mathbf{V}_L}^{-1}$ are block diagonal matrices with entries $\boldsymbol{\mathcal{I}}_{\mathbf{w}}^{-1}$ and $\boldsymbol{\mathcal{I}}_{\mathbf{v}}^{-1}$, respectively [74]. Nevertheless, the sum in (5.16) results in a dense matrix and finding the inverse is not straightforward.

Partly, this problem is similar to the derivation of the posterior CRB whose recursion is given in Sec. 4.1.4.2. There, it is strived for a recursive expression for the information matrix $\boldsymbol{\mathcal{I}}_{\mathbf{x}_k}$ of the state estimate which is part of the larger information matrix for the complete state sequence $\boldsymbol{\mathcal{I}}_{\mathbf{Y}_{1:k}}(\mathbf{X}_{1:k})$. To this end, it is necessary to find a recursion for the lower right block matrix of the inverse $\boldsymbol{\mathcal{I}}_{\mathbf{Y}_{1:k}}^{-1}(\mathbf{X}_{1:k})$. As has been shown in [188], this block matrix can be efficiently calculated without obtaining the entire inverse. The increased difficulty of the problem at hand comes from the outer multiplications in (5.18b). Thus, not only the lower right block but the full matrix $\boldsymbol{\mathcal{I}}_{\mathbf{E}_L}^{-1}$ is needed for the result.

The different terms in the linear regression problem (5.7) will be analysed first, since this is where the combined error term $\mathbf{E}_L$ comes from. On the left-hand side, the state estimate $\hat{\mathbf{x}}_{k-L+1}$ from the initial time step $k-L+1$ is multiplied by the extended observability matrix $\boldsymbol{\mathcal{O}}_L$ from (5.6a). Thereby, predictions of the state $\hat{\mathbf{x}}_{k-L+1}, \ldots, \hat{\mathbf{x}}_k$ for the subsequent time steps are implicitly calculated. These predictions, which are denoted by $\hat{\mathbf{X}}_L$ in stacked vector form, are then transformed as $\mathbf{H}_L\hat{\mathbf{X}}_L$ according to the measurement model (5.1c). Thus, the residual $\mathbf{R}_L$ is the difference between these predicted measurements and the actual observed ones, that is $\mathbf{Y}_L$.

The right-hand side of (5.7) contains the reasons for such a difference to occur. On the one hand, a manoeuvre change is reflected in the deterministic term $\boldsymbol{\Phi}_L\boldsymbol{\theta}$. All remaining terms on the other hand denote stochastic uncertainties:

1. Measurement errors are given by $\mathbf{V}_L$.

2. Uncertainty in the state predictions $\hat{\mathbf{X}}_L$ arises from errors $\Delta\mathrm{x}_{k-L+1}$ in the initial value and stochastic process noise $\mathbf{W}_L$. The overall state prediction error is denoted by $\Delta\mathbf{X}_L := \hat{\mathbf{X}}_L - \mathbf{X}_L$. Thus, this error projected on the measurements is $\mathbf{H}_L\Delta\mathbf{X}_L = \boldsymbol{\mathcal{O}}_L\Delta\mathrm{x}_{k-L+1} + \mathbf{H}_L^{\mathrm{w}}\mathbf{W}_L$ which constitutes the second part of $\mathbf{E}_L$ in (5.7).

Therefore, (5.16) can be equivalently written in terms of the intrinsic accuracy of the state prediction $\boldsymbol{\mathcal{I}}_{\mathbf{X}_L}$:

$$\boldsymbol{\mathcal{I}}_{\mathbf{E}_L} = \left(\boldsymbol{\mathcal{O}}_L \boldsymbol{\mathcal{I}}_{\mathbf{x}_{k-L+1}}^{-1} \boldsymbol{\mathcal{O}}_L^{\mathrm{T}} + \mathbf{H}_L^{\mathrm{w}} \boldsymbol{\mathcal{I}}_{\mathbf{W}_L}^{-1} \mathbf{H}_L^{\mathrm{w}\mathrm{T}}\right)^{-1} = \left(\mathbf{H}_L \boldsymbol{\mathcal{I}}_{\mathbf{X}_L}^{-1} \mathbf{H}_L^{\mathrm{T}} + \boldsymbol{\mathcal{I}}_{\mathbf{V}_L}^{-1}\right)^{-1} . \tag{5.24}$$

As a first step, the matrix inversion lemma (A.9) is used to reformulate (5.24):

$$\boldsymbol{\mathcal{I}}_{\mathbf{E}_L} = \boldsymbol{\mathcal{I}}_{\mathbf{V}_L} - \boldsymbol{\mathcal{I}}_{\mathbf{V}_L} \mathbf{H}_L \underbrace{\left(\boldsymbol{\mathcal{I}}_{\mathbf{X}_L} + \mathbf{H}_L^{\mathrm{T}} \boldsymbol{\mathcal{I}}_{\mathbf{V}_L} \mathbf{H}_L\right)^{-1}}_{=:\mathbf{M}_L} \mathbf{H}_L^{\mathrm{T}} \boldsymbol{\mathcal{I}}_{\mathbf{V}_L} . \tag{5.25}$$

The majority of the upcoming derivations focusses on the inverse that is abbreviated by $\mathbf{M}_L$. Note that this expression comprises $\boldsymbol{\mathcal{I}}_{\mathbf{X}_L}$ instead of its inverse. The key to finding a recursive expression for $\boldsymbol{\mathcal{I}}_{\mathbf{E}_L}$ will be to leverage the structure of $\boldsymbol{\mathcal{I}}_{\mathbf{X}_L}$. Due to the Markov assumption, this matrix has a block diagonal form [188]:[6]

$$\boldsymbol{\mathcal{I}}_{\mathbf{X}_L} = \begin{bmatrix} \boldsymbol{\mathcal{A}} & \boldsymbol{\mathcal{B}} \\ \boldsymbol{\mathcal{B}}^{\mathrm{T}} & \boldsymbol{\mathcal{C}} \end{bmatrix} , \qquad \boldsymbol{\mathcal{I}}_{\mathbf{X}_{L+1}} = \begin{bmatrix} \boldsymbol{\mathcal{A}} & \boldsymbol{\mathcal{B}} & \mathbf{0} \\ \boldsymbol{\mathcal{B}}^{\mathrm{T}} & \boldsymbol{\mathcal{C}} + \mathbf{D}_{11} & \mathbf{D}_{12} \\ \mathbf{0} & \mathbf{D}_{21} & \mathbf{D}'_{22} \end{bmatrix} . \tag{5.26}$$

The expressions for $\mathbf{D}_{11}$, $\mathbf{D}_{21}$ and $\mathbf{D}'_{22}$ for a linear system with additive non-Gaussian noise (5.1) are:

$$\mathbf{D}_{11} = \mathbf{A}^{\mathrm{T}} \boldsymbol{\mathcal{I}}_{\mathbf{w}} \mathbf{A} , \qquad \mathbf{D}_{12} = -\mathbf{A}^{\mathrm{T}} \boldsymbol{\mathcal{I}}_{\mathbf{w}} , \qquad \mathbf{D}'_{22} = \boldsymbol{\mathcal{I}}_{\mathbf{w}} . \tag{5.27}$$

The first two of these are identical to the ones introduced for the tracking bound in Sec. 4.1.4.4. The third term $\mathbf{D}'_{22}$ though does not contain a term related to the measurement information $\boldsymbol{\mathcal{I}}_{\mathbf{v}}$ since it refers to a *predicted* state.

It is then strived for an expression for $\mathbf{M}_{L+1}, L > 1$ which avoids the inversion of an $(L+1) \cdot n \times (L+1) \cdot n$-dimensional matrix. Firstly, $\mathbf{M}_L$ is written in a partitioned form using $\boldsymbol{\mathcal{I}}_{\mathbf{X}_L}$ from (5.26):

$$\mathbf{M}_L = \overset{\begin{matrix}(L-1)\cdot n & n\end{matrix}}{\begin{bmatrix} \mathbf{M}_{L,11} & \mathbf{M}_{L,12} \\ \mathbf{M}_{L,21} & \mathbf{M}_{L,22} \end{bmatrix}} = \left(\begin{bmatrix} \boldsymbol{\mathcal{A}} & \boldsymbol{\mathcal{B}} \\ \boldsymbol{\mathcal{B}}^{\mathrm{T}} & \boldsymbol{\mathcal{C}} \end{bmatrix} + \begin{bmatrix} \mathbf{H}_{L-1}^{\mathrm{T}} \boldsymbol{\mathcal{I}}_{\mathbf{V}_{L-1}} \mathbf{H}_{L-1} & \mathbf{0} \\ \mathbf{0} & \mathbf{C}^{\mathrm{T}} \boldsymbol{\mathcal{I}}_{\mathbf{v}} \mathbf{C} \end{bmatrix}\right)^{-1} . \tag{5.28}$$

Due to the white noise assumption on the measurement noise, $\boldsymbol{\mathcal{I}}_{\mathbf{V}_L}$ is a block diagonal matrix. The inversion in (5.28) can be calculated block-wise according to (A.12):

$$\mathbf{M}_{L,11} \overset{(A.12b)}{=} \left(\boldsymbol{\mathcal{A}} + \mathbf{H}_{L-1}^{\mathrm{T}} \boldsymbol{\mathcal{I}}_{\mathbf{V}_{L-1}} \mathbf{H}_{L-1} - \boldsymbol{\mathcal{B}} \left(\boldsymbol{\mathcal{C}} + \mathbf{C}^{\mathrm{T}} \boldsymbol{\mathcal{I}}_{\mathbf{v}} \mathbf{C}\right)^{-1} \boldsymbol{\mathcal{B}}^{\mathrm{T}}\right)^{-1} \tag{5.29a}$$

$$\mathbf{M}_{L,22} \overset{(A.12c)}{=} \left(\boldsymbol{\mathcal{C}} + \mathbf{C}^{\mathrm{T}} \boldsymbol{\mathcal{I}}_{\mathbf{v}} \mathbf{C} - \boldsymbol{\mathcal{B}}^{\mathrm{T}} \left(\boldsymbol{\mathcal{A}} + \mathbf{H}_{L-1}^{\mathrm{T}} \boldsymbol{\mathcal{I}}_{\mathbf{V}_{L-1}} \mathbf{H}_{L-1}\right)^{-1} \boldsymbol{\mathcal{B}}\right)^{-1} \tag{5.29b}$$

[6] The notation of the entries of $\boldsymbol{\mathcal{I}}_{\mathbf{X}_L}$ is chosen in accordance with [188]. Though, in order to avoid confusion with the system matrices from (5.1), a calligraphic font ($\boldsymbol{\mathcal{A}}$) is used here.

$$\mathbf{M}_{L,21} \overset{(A.12e)}{=} \mathbf{M}_{L,12}^{\mathrm{T}} = -\mathbf{M}_{L,22}\boldsymbol{\mathcal{B}}^{\mathrm{T}} \left(\boldsymbol{\mathcal{A}} + \mathbf{H}_{L-1}^{\mathrm{T}} \boldsymbol{\mathcal{I}}_{\mathbf{V}_{L-1}} \mathbf{H}_{L-1}\right)^{-1} . \tag{5.29c}$$

Secondly, a similar partitioning is introduced for $\mathbf{M}_{L+1}$, using (5.26):

$$\mathbf{M}_{L+1} = \left(\begin{bmatrix} \boldsymbol{\mathcal{A}} & \boldsymbol{\mathcal{B}} & \mathbf{0} \\ \boldsymbol{\mathcal{B}}^{\mathrm{T}} & \boldsymbol{\mathcal{C}} + \mathbf{D}_{11} & \mathbf{D}_{12} \\ \mathbf{0} & \mathbf{D}_{21} & \mathbf{D}'_{22} \end{bmatrix} + \begin{bmatrix} \mathbf{H}_{L-1}^{\mathrm{T}} \boldsymbol{\mathcal{I}}_{\mathbf{V}_{L-1}} \mathbf{H}_{L-1} & \mathbf{0} & \mathbf{0} \\ \mathbf{0} & \mathbf{C}^{\mathrm{T}} \boldsymbol{\mathcal{I}}_{\mathbf{v}} \mathbf{C} & \mathbf{0} \\ \mathbf{0} & \mathbf{0} & \mathbf{C}^{\mathrm{T}} \boldsymbol{\mathcal{I}}_{\mathbf{v}} \mathbf{C} \end{bmatrix} \right)^{-1} . \tag{5.30}$$

Again, the inversion in (5.30) is calculated block-wise and expressions which are identical to the ones in $\mathbf{M}_L$ are replaced by the respective matrices from (5.29). Firstly, the $L{\cdot}n \times L{\cdot}n$-dimensional upper left block of $\mathbf{M}_{L+1}$ is denoted by $\mathbf{M}_{L+1,11}$:

$$\begin{aligned} \mathbf{M}_{L+1,11} &\overset{(A.12b)}{=} \left(\begin{bmatrix} \boldsymbol{\mathcal{A}} + \mathbf{H}_{L-1}^{\mathrm{T}} \boldsymbol{\mathcal{I}}_{\mathbf{V}_{L-1}} \mathbf{H}_{L-1} & \boldsymbol{\mathcal{B}} \\ \boldsymbol{\mathcal{B}}^{\mathrm{T}} & \boldsymbol{\mathcal{C}} + \mathbf{D}_{11} + \mathbf{C}^{\mathrm{T}} \boldsymbol{\mathcal{I}}_{\mathbf{v}} \mathbf{C} \end{bmatrix} \right. \\ &\qquad \left. - \begin{bmatrix} \mathbf{0} \\ \mathbf{D}_{12} \end{bmatrix} \left(\mathbf{D}'_{22} + \mathbf{C}^{\mathrm{T}} \boldsymbol{\mathcal{I}}_{\mathbf{v}} \mathbf{C} \right)^{-1} \begin{bmatrix} \mathbf{0} & \mathbf{D}_{21} \end{bmatrix} \right)^{-1} \\ &\overset{(5.28)}{=} \left(\mathbf{M}_L^{-1} + \begin{bmatrix} \mathbf{0} \\ \mathbf{I}_{n\times n} \end{bmatrix} \boldsymbol{\mathcal{S}} \begin{bmatrix} \mathbf{0} & \mathbf{I}_{n\times n} \end{bmatrix} \right)^{-1} \\ &\overset{(A.11)}{=} \mathbf{M}_L - \begin{bmatrix} \mathbf{M}_{L,12} \\ \mathbf{M}_{L,22} \end{bmatrix} \boldsymbol{\mathcal{S}} \left(\mathbf{M}_{L,22}^{-1} + \boldsymbol{\mathcal{S}} \right)^{-1} \mathbf{M}_{L,22}^{-1} \begin{bmatrix} \mathbf{M}_{L,21} & \mathbf{M}_{L,22} \end{bmatrix} . \end{aligned} \tag{5.31}$$

Here, the following abbreviation has been introduced:

$$\boldsymbol{\mathcal{S}} := \mathbf{D}_{11} - \mathbf{D}_{12} \underbrace{\left(\mathbf{D}'_{22} + \mathbf{C}^{\mathrm{T}} \boldsymbol{\mathcal{I}}_{\mathbf{v}} \mathbf{C} \right)}_{\overset{(4.28)}{=:} \mathbf{D}_{22}}{}^{-1} \mathbf{D}_{21} = \boldsymbol{\mathcal{S}}^{\mathrm{T}} . \tag{5.32}$$

$\boldsymbol{\mathcal{S}}$ is the Schur complement of the matrix $\mathbf{D}$ with respect to its part $\mathbf{D}_{22}$.

Secondly, $\mathbf{M}_{L+1,22}$ denotes the $n \times n$-dimensional lower right block:

$$\begin{aligned} &\mathbf{M}_{L+1,22} \\ &\overset{(A.12c)}{=} \left(\mathbf{D}_{22} - \begin{bmatrix} \mathbf{0} & \mathbf{D}_{21} \end{bmatrix} \begin{bmatrix} \boldsymbol{\mathcal{A}} + \mathbf{H}_{L-1}^{\mathrm{T}} \boldsymbol{\mathcal{I}}_{\mathbf{V}_{L-1}} \mathbf{H}_{L-1} & \boldsymbol{\mathcal{B}} \\ \boldsymbol{\mathcal{B}}^{\mathrm{T}} & \boldsymbol{\mathcal{C}} + \mathbf{D}_{11} + \mathbf{C}^{\mathrm{T}} \boldsymbol{\mathcal{I}}_{\mathbf{v}} \mathbf{C} \end{bmatrix}^{-1} \begin{bmatrix} \mathbf{0} \\ \mathbf{D}_{12} \end{bmatrix} \right)^{-1} \\ &\overset{(A.12c)}{=} \left(\mathbf{D}_{22} - \mathbf{D}_{21} \left(\boldsymbol{\mathcal{C}} + \mathbf{D}_{11} + \mathbf{C}^{\mathrm{T}} \boldsymbol{\mathcal{I}}_{\mathbf{v}} \mathbf{C} - \boldsymbol{\mathcal{B}}^{\mathrm{T}} \left(\boldsymbol{\mathcal{A}} + \mathbf{H}_{L-1}^{\mathrm{T}} \boldsymbol{\mathcal{I}}_{\mathbf{V}_{L-1}} \mathbf{H}_{L-1} \right)^{-1} \boldsymbol{\mathcal{B}} \right)^{-1} \mathbf{D}_{12} \right)^{-1} \\ &\overset{(5.29b)}{=} \left(\mathbf{D}_{22} - \mathbf{D}_{21} \left(\mathbf{D}_{11} + \mathbf{M}_{L,22}^{-1} \right)^{-1} \mathbf{D}_{12} \right)^{-1} . \end{aligned} \tag{5.33}$$

Besides the different initial value $\mathbf{M}_{1,22} = \left(\boldsymbol{\mathcal{I}}_{\mathbf{x}_{k-L+1}} + \mathbf{C}^{\mathrm{T}} \boldsymbol{\mathcal{I}}_{\mathbf{v}} \mathbf{C} \right)^{-1}$ this is the same recursion as for the inverse information matrix of a *filtered* state from (4.29).

Finally, the lower left block, which is $n \times L \cdot n$-dimensional, is obtained as:

$$\begin{aligned}
&\mathbf{M}_{L+1,21} \\
&\overset{(A.12e)}{=} -\mathbf{M}_{L+1,22} \cdot \begin{bmatrix} \mathbf{0} & \mathbf{D}_{21} \end{bmatrix} \begin{bmatrix} \boldsymbol{\mathcal{A}} + \mathbf{H}_{L-1}^{\mathrm{T}} \boldsymbol{\mathcal{I}}_{\mathbf{V}_{L-1}} \mathbf{H}_{L-1} & \boldsymbol{\mathcal{B}} \\ \boldsymbol{\mathcal{B}}^{\mathrm{T}} & \boldsymbol{\mathcal{C}} + \mathbf{D}_{11} + \mathbf{C}^{\mathrm{T}} \boldsymbol{\mathcal{I}}_{\mathbf{v}} \mathbf{C} \end{bmatrix}^{-1} \\
&\overset{(5.29b)}{=} -\mathbf{M}_{L+1,22} \mathbf{D}_{21} \left(\mathbf{D}_{11} + \mathbf{M}_{L,22}^{-1} \right)^{-1} \left[-\boldsymbol{\mathcal{B}}^{\mathrm{T}} \left(\boldsymbol{\mathcal{A}} + \mathbf{H}_{L-1}^{\mathrm{T}} \boldsymbol{\mathcal{I}}_{\mathbf{V}_{L-1}} \mathbf{H}_{L-1} \right)^{-1} \quad \mathbf{I}_{n \times n} \right] \\
&\overset{(5.29c)}{=} \underbrace{-\mathbf{M}_{L+1,22} \mathbf{D}_{21} \left(\mathbf{D}_{11} + \mathbf{M}_{L,22}^{-1} \right)^{-1} \mathbf{M}_{L,22}^{-1}}_{=: \boldsymbol{\Gamma}_{L+1}} \begin{bmatrix} \mathbf{M}_{L,21} & \mathbf{M}_{L,22} \end{bmatrix} .
\end{aligned} \tag{5.34}$$

Therefore, $\mathbf{M}_{L+1,21}$ consists of L sub-matrices $\mathbf{M}_{L+1,21}^{(l)}$ of size $n \times n$:

$$\mathbf{M}_{L+1,21} = \begin{bmatrix} \mathbf{M}_{L+1,21}^{(1)} & \dots & \mathbf{M}_{L+1,21}^{(L)} \end{bmatrix} \qquad \text{with} \qquad \mathbf{M}_{L+1,21}^{(l)} = \prod_{j=l}^{L} \boldsymbol{\Gamma}_{j+1} \mathbf{M}_{l,22} . \tag{5.35}$$

In a similar manner and due to symmetry it follows that $\mathbf{M}_{L+1,12} = \mathbf{M}_{L+1,21}^{\mathrm{T}}$.

To summarise the intermediate result, a recursive expression for $\mathbf{M}_L$ has been derived which can be used to obtain $\boldsymbol{\mathcal{I}}_{\mathbf{E}_L}$ according to (5.25). In contrast to the batch calculation in (5.16), only inverses of matrices with a size of $n \times n$ have to be calculated.

It is especially noteworthy that $\mathbf{M}_{L+1,11}$ in (5.31) consists of the result from the previous iteration, that is $\mathbf{M}_L$, and a second additive term. Since all subsequent calculations that lead to λ_L only involve further matrix multiplications and additions, this incremental relationship will propagate to the final result.

In the next step, $\mathbf{M}_{L+1}$ is inserted into (5.25) to obtain $\boldsymbol{\mathcal{I}}_{\mathbf{E}_{L+1}}$:

$$\boldsymbol{\mathcal{I}}_{\mathbf{E}_{L+1}} = \begin{bmatrix} \boldsymbol{\mathcal{I}}_{\mathbf{V}_L} & \mathbf{0} \\ \mathbf{0} & \boldsymbol{\mathcal{I}}_{\mathbf{v}} \end{bmatrix} - \begin{bmatrix} \boldsymbol{\mathcal{I}}_{\mathbf{V}_L} \mathbf{H}_L \mathbf{M}_{L+1,11} \mathbf{H}_L^{\mathrm{T}} \boldsymbol{\mathcal{I}}_{\mathbf{V}_L} & \boldsymbol{\mathcal{I}}_{\mathbf{V}_L} \mathbf{H}_L \mathbf{M}_{L+1,12} \mathbf{C}^{\mathrm{T}} \boldsymbol{\mathcal{I}}_{\mathbf{v}} \\ \boldsymbol{\mathcal{I}}_{\mathbf{v}} \mathbf{C} \mathbf{M}_{L+1,21} \mathbf{H}_L^{\mathrm{T}} \boldsymbol{\mathcal{I}}_{\mathbf{V}_L} & \boldsymbol{\mathcal{I}}_{\mathbf{v}} \mathbf{C} \mathbf{M}_{L+1,22} \mathbf{C}^{\mathrm{T}} \boldsymbol{\mathcal{I}}_{\mathbf{v}} \end{bmatrix} . \tag{5.36}$$

According to the definition (5.6c), $\boldsymbol{\Phi}_{L+1}$ can be written as $\boldsymbol{\Phi}_{L+1} = \begin{bmatrix} \boldsymbol{\Phi}_L^{\mathrm{T}} & \boldsymbol{\varphi}_{L+1} \end{bmatrix}^{\mathrm{T}}$. Then, inserting into (5.18b) yields:

$$\begin{aligned}
\lambda_{L+1} = &\left(\boldsymbol{\Phi}_L \left(\boldsymbol{\theta}_1 - \boldsymbol{\theta}_0 \right) \right)^{\mathrm{T}} \left(\boldsymbol{\mathcal{I}}_{\mathbf{V}_L} - \boldsymbol{\mathcal{I}}_{\mathbf{V}_L} \mathbf{H}_L \mathbf{M}_{L+1,11} \mathbf{H}_L^{\mathrm{T}} \boldsymbol{\mathcal{I}}_{\mathbf{V}_L} \right) \left(\boldsymbol{\Phi}_L \left(\boldsymbol{\theta}_1 - \boldsymbol{\theta}_0 \right) \right) \\
&- 2 \left(\boldsymbol{\Phi}_L \left(\boldsymbol{\theta}_1 - \boldsymbol{\theta}_0 \right) \right)^{\mathrm{T}} \boldsymbol{\mathcal{I}}_{\mathbf{V}_L} \mathbf{H}_L \mathbf{M}_{L+1,21}^{\mathrm{T}} \mathbf{C}^{\mathrm{T}} \boldsymbol{\mathcal{I}}_{\mathbf{v}} \left(\boldsymbol{\varphi}_{L+1}^{\mathrm{T}} \left(\boldsymbol{\theta}_1 - \boldsymbol{\theta}_0 \right) \right) \\
&+ \left(\boldsymbol{\varphi}_{L+1}^{\mathrm{T}} \left(\boldsymbol{\theta}_1 - \boldsymbol{\theta}_0 \right) \right)^{\mathrm{T}} \left(\boldsymbol{\mathcal{I}}_{\mathbf{v}} - \boldsymbol{\mathcal{I}}_{\mathbf{v}} \mathbf{C} \mathbf{M}_{L+1,22} \mathbf{C}^{\mathrm{T}} \boldsymbol{\mathcal{I}}_{\mathbf{v}} \right) \left(\boldsymbol{\varphi}_{L+1}^{\mathrm{T}} \left(\boldsymbol{\theta}_1 - \boldsymbol{\theta}_0 \right) \right) .
\end{aligned} \tag{5.37}$$

The expression (5.37) can be further simplified as detailed in Sec. A.4 of the appendix:

$$\lambda_{L+1} = \lambda_L + \underbrace{\mathbf{a}_{L+1}^{\mathrm{T}} \cdot \left(\boldsymbol{\mathcal{I}}_{\mathbf{v}}^{-1} - \mathbf{C} \mathbf{M}_{L+1,22} \mathbf{C}^{\mathrm{T}} \right) \cdot \mathbf{a}_{L+1}}_{=: \Delta\lambda_{L+1}} , \tag{5.38a}$$

where

$$\mathbf{a}_{L+1} = \boldsymbol{\mathcal{I}}_{\mathbf{v}} \boldsymbol{\varphi}_{L+1}^{\mathrm{T}} \left(\boldsymbol{\theta}_1 - \boldsymbol{\theta}_0 \right) - \left(\boldsymbol{\mathcal{I}}_{\mathbf{v}}^{-1} - \mathbf{C} \mathbf{M}_{L+1,22} \mathbf{C}^{\mathrm{T}} \right)^{-1} \mathbf{C} \boldsymbol{\Lambda}_{L+1}^{\mathrm{T}} , \tag{5.38b}$$

$$\begin{aligned}\boldsymbol{\Lambda}_{L+1} &= -\left(\boldsymbol{\Lambda}_L \mathbf{M}_{L,22}^{-1} + \left(\boldsymbol{\varphi}_L^{\mathrm{T}}\left(\boldsymbol{\theta}_1 - \boldsymbol{\theta}_0\right)\right)^{\mathrm{T}} \boldsymbol{\mathcal{I}}_{\mathrm{v}} \mathbf{C}\right)\left(\boldsymbol{\mathcal{S}} + \mathbf{M}_{L,22}^{-1}\right)^{-1} \mathbf{D}_{12}\mathbf{D}_{22}^{-1}\,, \\ \boldsymbol{\Lambda}_1 &= \mathbf{0}_{n_\theta \times n}\,.\end{aligned} \tag{5.38c}$$

This is the final result for the desired recursive form of (5.18b). In order to calculate λ_{L+1} it is thus necessary to propagate the recursions for two matrices,[7] that are $\mathbf{M}_{L+1,22}$ from (5.33) and $\boldsymbol{\Lambda}_{L+1}$ from (5.38c). On the one hand, $\mathbf{M}_{L+1,22}$ contains the information matrix of the predicted state which is used to calculate the residual and closely linked to the CRB for the state estimation problem from Sec. 4.1.4.4. $\boldsymbol{\Lambda}_{L+1}$ on the other hand defines a projection of this matrix on the space of the manoeuvre change and thereby integrates its time-dependence.

With an expression for the increment $\Delta\lambda_{L+1}$, the effect of an additional measurement on the detection probability from (5.19) can be investigated. To this end, $P_{\mathrm{D}}\left(P_{\mathrm{FA}}\right)$ is linearised around λ_L:

$$P_{\mathrm{D}}\left(P_{\mathrm{FA}}\right) \approx P_{\mathrm{D}}\left(P_{\mathrm{FA}}\right)|_{\lambda_L} + \underbrace{\frac{\mathrm{d}}{\mathrm{d}\lambda} P_{\mathrm{D}}\left(P_{\mathrm{FA}}\right)\Big|_{\lambda_L} \Delta\lambda_{L+1}}_{:=\Delta P_{\mathrm{D}}}\,. \tag{5.39}$$

However, there is no closed-form expression for the cumulative distribution function $P_{\chi^2_{n_\theta}(\lambda)}(\cdot)$ of the non-central χ^2-distribution, which is required here. Instead, an analytical approximation is employed and the derivative thereof calculated. Out of the numerous proposed approximations, a compact variant by PATNAIK [165] is used here:

$$P_{\chi^2_{n_\theta}(\lambda)}(z) = \int_0^z \chi^2_\nu\left(\zeta;\lambda\right)\,\mathrm{d}\zeta \approx \int_{-\infty}^{z'(z,\lambda)} \mathcal{N}\left(\zeta;\mu_\zeta\left(\lambda\right),1\right)\,\mathrm{d}\zeta \tag{5.40a}$$

$$\text{with} \qquad z'\left(z,\lambda\right) = \sqrt{2z\frac{\nu+\lambda}{\nu+2\lambda}}\,, \qquad \mu_\zeta\left(\lambda\right) = \sqrt{2\frac{\left(\nu+\lambda\right)^2}{\nu+2\lambda} - 1}\,. \tag{5.40b}$$

Thus, one obtains an approximate derivative with respect to λ in closed-form

$$\begin{aligned}\frac{\mathrm{d}}{\mathrm{d}\lambda} P_{\chi^2_\nu(\lambda)}(z) &\approx \frac{\mathrm{d}}{\mathrm{d}\lambda}\int_{-\infty}^{z'(z,\lambda)} \frac{1}{\sqrt{2\pi}}\exp\left(-\frac{1}{2}\left(\zeta-\mu_\zeta\left(\lambda\right)\right)^2\right)\,\mathrm{d}\zeta \\ &= \frac{1}{\sqrt{2\pi}}\exp\left(-\frac{1}{2}\left(z'\left(z,\lambda\right)-\mu_\zeta\left(\lambda\right)\right)^2\right)\cdot\frac{\mathrm{d}}{\mathrm{d}\lambda}\left(z'\left(z,\lambda\right)-\mu_\zeta\left(\lambda\right)\right)\,.\end{aligned} \tag{5.41}$$

5.1.5 Numerical example

In order to exemplify the theoretical results, a possible application for a manoeuvre detection algorithm is analysed in simulations. While the example is not explicitly related to a specific driver assistance function, the ambiguity of the situation makes it a typical candidate for driver intention recognition.

[7] The recursions for the sub-matrices $\mathbf{M}_{L+1,11}$ and $\mathbf{M}_{L+1,21}$ are not necessary to calculate since these are implicitly contained in the final result.

The simulated scene is visualised in Fig. 5.2 and consists of the ego-vehicle and a preceding target vehicle. Both vehicles approach a signalled intersection at equal velocity of $50\,\mathrm{km/h}$. Since the traffic light has just switched from green to yellow, the driver of the first vehicle may conduct two possible manoeuvres:

1. He may keep his velocity or even accelerate slightly to pass the intersection.[8]

2. He may decelerate in order to stop in front of the intersection.

These two possibilities lead to qualitatively different trajectories of the target vehicle. In order to plan the future course of the ego-vehicle, an early differentiation between the manoeuvres is crucial.

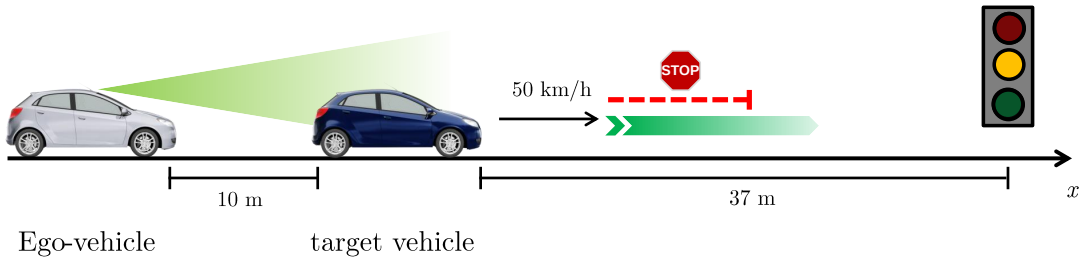


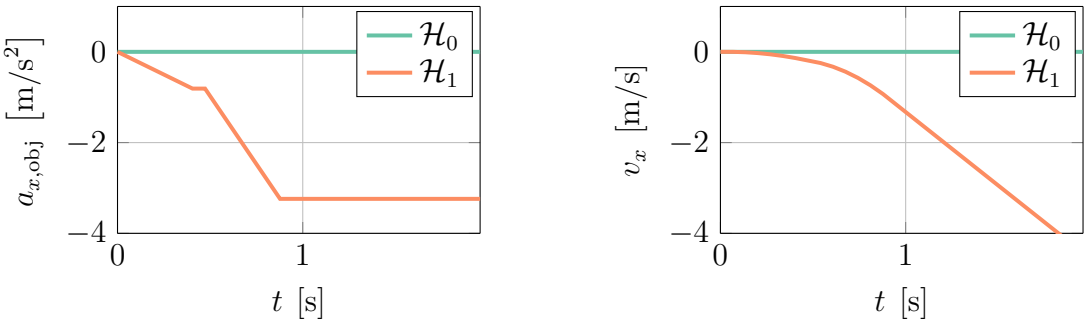
Figure 5.2 Illustration of the simulation scenario. The ego-vehicle (equipped with a front facing environment sensor) and a preceding target vehicle approach a traffic light at equal initial velocity. Given the remaining distance to the traffic light when the signal switches from green to yellow, it is ambiguous whether the driver of the target vehicle intends to stop or pass. For driver assistance functions, early detection of this driver intention is a crucial task.

Firstly, the problem will be formulated in the chosen framework from Sec. 5.1.3. Secondly, the optimal decision rule for a hypothesis test as well as the asymptotic distribution can be set up according to Sec. 5.1.4. A Monte-Carlo simulation with $N_{\mathrm{sim}} = 1 \times 10^5$ independent runs is then used to obtain a numerical estimate of the true and false positive detection probabilities which can be compared to the analytical performance bound. Different lengths $L \in \{1, \ldots, 20\}$ of the sliding window are used. The initial time-step corresponds to the beginning of the manoeuvre.[9]

Since the two vehicles are driving with equal velocity at first, the CV model is used to describe their relative motion. The state vector comprises distance x and relative

[8] An initial distance of $37\,\mathrm{m}$ to the intersection and a duration of the yellow phase of $3\,\mathrm{s}$, according to German legislation for inner-city traffic lights, are assumed. Therefore, the intersection can be reached after $2.67\,\mathrm{s}$ at a speed of $50\,\mathrm{km/h}$.

[9] In practice, a sliding window will encompass time-steps both before and after occurrence of the manoeuvre. One could treat the time of occurrence as an additional unknown parameter which is to be estimated [201]. By considering only those measurements which contribute information to the detection problem, we thus analyse the optimal performance.



(a) Acceleration profile. **(b)** Relative velocity.

Figure 5.3 Deceleration of the target vehicle over time (a) and the resulting relative velocity between ego and target vehicle (b). The brake characteristic is modelled with two ramp functions according to (5.42b) with the parameter values as given in Tab. 5.1.

longitudinal velocity v_x. A braking manoeuvre is described by a deceleration command as input signal u_k. Therefore, the state transition in the form of (5.1a) reads:

$$\begin{bmatrix} x_{k+1} \\ v_{x,k+1} \end{bmatrix} = \begin{bmatrix} 1 & T \\ 0 & 1 \end{bmatrix} \begin{bmatrix} x_k \\ v_{x,k} \end{bmatrix} + \begin{bmatrix} \frac{1}{2}T^2 \\ T \end{bmatrix} u_k + \mathbf{w}_k \ , \qquad \mathbf{w}_k \sim \mathcal{N}(\mathbf{0}, \mathbf{Q}) \ . \tag{5.42a}$$

The time evolution of u_k describes the applied deceleration profile. There are multiple possible characteristics, from a simple ramp function to more sophisticated models which take a controller-like adaptation into account [137]. The framework (5.1b) permits to approximate such general non-linear characteristics by means of a linear parametrisation. In the simulation, an ideally known model where the parameters enter linearly is chosen. The brake deceleration is assumed to follow two subsequent ramp functions with known durations $\left(t_{\mathrm{ramp},1}, t_{\mathrm{ramp},2}\right)$ but unknown slope parameters $\left(r_{\mathrm{brake},1}, r_{\mathrm{brake},2}\right)$. This can be interpreted as a driver who applies an initial force to the brake pedal but realises that the resulting deceleration requires further adjustment in order to stop at a desired point:

$$u_k = \begin{cases} r_{\mathrm{brake},1} t_k & 0 \leq t_k \leq t_{\mathrm{ramp},1} \\ r_{\mathrm{brake},1} t_{\mathrm{ramp},1} & t_{\mathrm{ramp},1} < t_k \leq t_{\mathrm{start},2} \\ r_{\mathrm{brake},1} t_{\mathrm{ramp},1} + r_{\mathrm{brake},2} \left(t_k - t_{\mathrm{start},2}\right) & t_{\mathrm{start},2} < t_k \leq t_{\mathrm{start},2} + t_{\mathrm{ramp},2} \end{cases} . \tag{5.42b}$$

The unknown parameter vector is $\boldsymbol{\theta} = \begin{bmatrix} r_{\mathrm{brake},1} & r_{\mathrm{brake},2} \end{bmatrix}$ and thus, $\mathcal{H}_1$ is a composite hypothesis. Whereas the non-braking case is a simple hypothesis with $\boldsymbol{\theta}_0 = \mathbf{0}$. For the chosen parametrisation, this results in the acceleration and velocity profiles as visualised in Fig. 5.3. The exemplary parameter values are detailed in Tab. 5.1.

Measurements of the distance are obtained with a Gaussian measurement distribution:

$$y_k = \begin{bmatrix} 1 & 0 \end{bmatrix} \mathbf{x}_k + v_k \ , \qquad v_k \sim \mathcal{N}(0, R) \ . \tag{5.42c}$$

A detector for the two hypotheses is implemented according to the principle of the Wald test (5.17). Due to the Gaussian noise assumption, the closed-form GLS estimator

Table 5.1 Simulation parameter values of a manoeuvre recognition task.

Variable	Value
Initial state mean	$\mathbf{x}_{k-L+1} = \begin{bmatrix} 10\,\mathrm{m} & 0 \end{bmatrix}$
Initial state covariance	$\boldsymbol{\Sigma}_{\mathbf{x}_{k-L+1}} = \boldsymbol{\mathcal{I}}_{\mathbf{x}_{k-L+1}}^{-1} = \mathrm{diag}\left(0.0625\,\mathrm{m}^2, 0.04\,\mathrm{m}^2/\mathrm{s}^2\right)$
Process noise	$\mathbf{Q} = \boldsymbol{\mathcal{I}}_{\mathbf{w}}^{-1} = 0.375\,\mathrm{m}^2/\mathrm{s}^4\mathrm{s}^{-1} \begin{bmatrix} \frac{1}{3}{T_\mathrm{S}}^3 & \frac{1}{2}{T_\mathrm{S}}^2 \\ \frac{1}{2}{T_\mathrm{S}}^2 & T_\mathrm{S} \end{bmatrix}$
Measurement noise	$R = \mathcal{I}_v^{-1} = 0.04\,\mathrm{m}^2$
Brake ramp model	$r_{\mathrm{brake},1} = -2\,\mathrm{m/s}^3\,,\quad t_{\mathrm{ramp},1} = 0.405\,\mathrm{s}\,,$ $r_{\mathrm{brake},2} = -6\,\mathrm{m/s}^3\,,\quad t_{\mathrm{start},2} = 0.440\,\mathrm{s}\,,\qquad t_{\mathrm{ramp},2} = 0.405\,\mathrm{s}$
Sampling time	$T_\mathrm{S} = 0.0675\,\mathrm{s}$

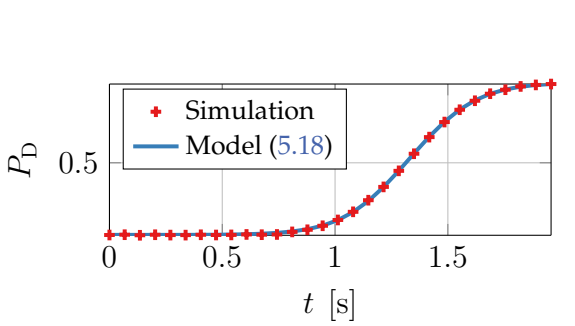
(5.20) is employed to estimate $\boldsymbol{\theta}$. Decision making is then performed with varying threshold values γ where the true and false positive probabilities are evaluated for each window length L. The results are shown in terms of the true positive probability at a specific false positive probability of $P_\mathrm{FA} = 5\,\%$ in Fig. 5.4(a) and the overall ROC curves in Fig. 5.4(c).

Comparing the Monte-Carlo simulation results to the model of the asymptotic GLR test statistic shows a good correspondence. It is furthermore notable from Fig. 5.4(a) that the dependence of the true positive probability P_D on the number of measurements L resembles a sigmoid curve. With the proposed analytic model (5.41) for the increments ΔP_D as shown in Fig. 5.4(b) it is thus possible to identify the point with the highest increase. This can be used for example as a sensible initial value for the minimum required L.

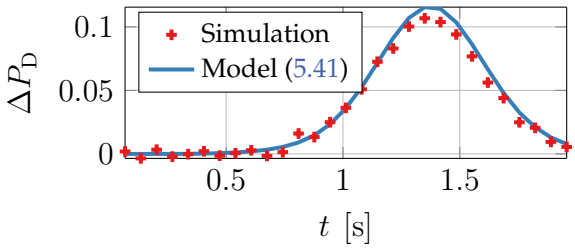
5.1.6 Summary

This section has developed probabilistic models of driver intention recognition, which is a crucial task for achieving long-term trajectory predictions. To this end, the problem has been formulated as a change detection problem in linear state space models. This class of models allows deriving analytical performance bounds of an optimal hypothesis test. A theoretical contribution, not limited to the application of manoeuvre recognition, is the derivation of a recursive version of this upper performance bound. In contrast to previous results, the inversion of matrices with growing dimensions can be avoided and further insight into the time-dependence is gained. The application of the framework is illustrated for an exemplary scenario and the obtained theoretical result are validated in Monte-Carlo simulations.

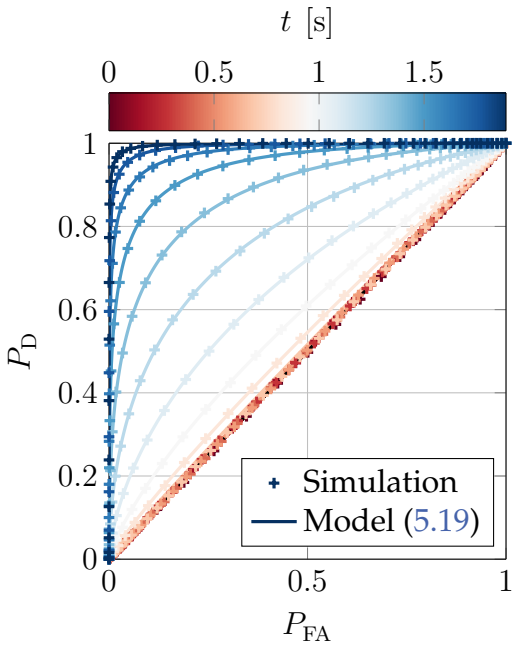
While the developed method is generic with respect to the specific linear system models, future works could extend the concept to non-linear models and M-ary hypothesis tests. Moreover, the GLR test relies on classical parameter estimation principles. If



(a) True positive detection probability.



(b) Increments ΔP_D of the true positive detection probability.



(c) Receiver operating characteristic curves.

Figure 5.4 Simulation results: (a) detection probability P_D, (b) increments ΔP_D for false positive probability $P_\mathrm{FA} = 5\,\%$ and (c) receiver operating characteristic curves.

additional prior knowledge is available, for example context information, it may be extended to a Bayesian approach [106].

5.2 Criticality measures for risk assessment

Algorithms for situation interpretation use the scene representation that is obtained at the perception (Sec. 3.1) and object tracking (Sec. 4.1) layer in order to infer an appropriate strategy for a system intervention. Reasoning about such interventions is often based on a prediction of future trajectories as studied in Sec. 4.2 and Sec. 5.1.

This chapter analyses situation interpretation for autonomous emergency brake (AEB) systems. The specifics of these systems and the implications of uncertainties are firstly introduced in Sec. 5.2.1. An overview of related works about situation analysis for emergency brake systems and probabilistic modelling of uncertainties is then given in Sec. 5.2.2. Examples of such algorithms are introduced in Sec. 5.2.3 and formulated in a common mathematical framework. Based on this, a fully probabilistic, analytical framework for the propagation of measurement and prediction errors is subsequently developed in Sec. 5.2.4 which presents the main contribution of this section. The findings are illustrated with numerical examples in Sec. 5.2.5 and lastly summarised in Sec. 5.2.6.

5.2.1 Introduction

Autonomous emergency brake systems aim to avoid or mitigate accidents due to collisions with other vehicles, static objects or vulnerable road users (for example pedestrians), by means of warnings to the driver or autonomous brake interventions. An AEB system should respond early in the rare case of an impending collision but should not falsely activate during uncritical normal driving. With a limited and uncertain perception of the environment, algorithms for situation interpretation have to achieve these two conflicting goals of performance and robustness.

In general, human-like scene understanding is a challenging task due to the complexity of the environment, especially in urban scenarios. One part of the problem is to estimate a human driver's intentions. As has been previously pointed out in Sec. 5.1 this is still an active research topic. However, for the narrow scope that is addressed by an AEB system – a limited set of critical situations with short prediction time spans and a small set of reaction patterns – tailored analytical solutions have been developed in the past. These algorithms are termed *criticality measures* and have been implemented both in prototypes [95] and series products. Usually, the criticality of a situation is expressed as a normalised scalar measure and decision making reduces to a threshold check.

In this work, criticality measures that aim to quantify the risk of a collision with a preceding vehicle in longitudinal traffic are considered. Three basic examples of criticality measures are derived in Sec. 5.2.3. The main contributions concern statistical modelling of criticality measures. Research on this topic is spurred by two motivations:

- Although uncertainties often cannot be ruled out entirely, knowing probabilistic models thereof can benefit the robustness of a system. Thereby, an intervention decision is generalised from a threshold check to a Bayesian formulation [95, 173].

- At the development stage, sensor accuracy requirements have to be determined in order to achieve a desired collision avoidance performance. For a given braking capability of the vehicle, this reduces to the question of activation timing. Models for the deviation from an ideal timing due to uncertainties can thus be used to obtain requirements on a sensor.

While probabilistic analyses of criticality measures have been developed for the aforementioned motivations, some questions remain unanswered in previous works. As it is further discussed in Sec. 5.2.2, this topic has been previously studied mainly with Monte-Carlo simulations [46, 83] or analytically under the assumption of bounded errors [144].

Criticality measures implicitly contain kinematic prediction models since they assess the risk of a future collision. In addition to measurement errors, inaccuracies of the prediction model, as have been discussed in Sec. 4.2, thus cause an uncertain situation assessment. To the best of the author's knowledge, an analytical treatment of these prediction errors has not been developed in previous works.

Central contributions of this section have been previously published in [221]. Novel extensions to correlated errors and an analysis of the time of activation are included in the following.

5.2.2 Related work

The state of the art is reviewed separately for the development of criticality measures and analyses of the impact of uncertainties.

Criticality measures First, a brief overview of deterministic criticality measures, as analysed in this work, is given. Second, probabilistic criticality measures are reviewed, which have evolved as a natural extension of the deterministic metrics.

In the most basic interpretation, criticality can be measured by a binary indicator whether a collision is unavoidable. This means that all trajectories which lie within a vehicle's acceleration and steering capabilities are checked for possible collisions [100, 169]. In order to facilitate a multi-stage intervention strategy though, for example with early warnings and successive brake interventions of increasing strength [82], a continuous measure of criticality is required.

Popular examples for such continuous scalar metrics are on the one hand time-based and indicate the remaining time until a critical event is predicted to occur, for example the collision (*time-to-collision*) or the last moment at which an accident is still avoidable (*time-to-react*) [83]. On the other hand, measures related to the necessary driver inputs in order to avoid a collision in the last possible moment provide a clear physical interpretation (*brake-threat-number*, *steer-threat-number*) [95]. The latter are further refined in [30] by more accurately modelling the vehicle geometries.

In order to handle stochastic errors, the state estimation uncertainty, which is usually provided by a tracking filter, is propagated to the above mentioned metrics by means of

Monte-Carlo sampling methods in [95, 104]. This idea is extended in [31, 32, 53, 109] to a stochastic future driver behaviour. An extensive Bayesian generalisation of the classical *time-to-collision* metric has been recently suggested in [173]. While these works consider sophisticated probabilistic prediction models, they are all based on numerical methods. Analytical approaches, which are proposed for time-based metrics in [18, 19], do not consider the uncertainty of a prediction.

Analysis of uncertainty in criticality algorithms The influence of sensor noise on criticality measures is evaluated with simulations in [46, 207]. Additionally, [83] studies the distribution of the decision timing. There, prediction errors are either not considered or assumed as deterministically bounded.

A further in-depth simulation-based evaluation is presented in [137] for the *time-to-react* metric. Recordings of real vehicle trajectories are employed in order to study both sensor and prediction errors. This concept is similar to the approach in this work, since the deviations between a prediction model and real driver behaviour are taken into account. The difference is that this thesis focusses on a fully analytical treatment. Empirical data is first used to learn probabilistic models of the prediction uncertainty (Sec. 4.2), which are then analytically propagated to the criticality measures in this section.

A worst case analysis is discussed in [144] where closed-form expressions for the decision timing of two acceleration-based criticality measures are derived. Besides the purely deterministic treatment, predictions are only regarded as a means to compensate for short-time sensor delays prior to calculating the criticality. Thus, errors in the inherent predictions performed by the algorithms themselves are not taken into account.

5.2.3 Derivation of criticality measures

Since numerous proposals for criticality measures have been proposed in the literature, the scope of this section is not to reiterate every individual derivation. For a number of relevant algorithms, these can be found for example in [83, 95]. Instead, it is attempted to formulate a common ground on which these criticality measures can be derived. This abstract framework is then illustrated with examples of three criticality measures, that are the *time-to-collision*, *brake-threat-number* and *time-to-brake*. The added value of the framework is that it enables to derive a generic approach to uncertainty propagation.

From the foremost goal of an AEB system to avoid or mitigate the impact of a collision, finding the point in time where a collision becomes unavoidable is crucial. Speaking not in a strict mathematical sense, the objective of a criticality metric is then to measure how close the current driving situation is to such an event. The motion of the ego-vehicle and object are expressed in dynamic states $\mathbf{x}^{\mathrm{e}}(t)$ and $\mathbf{x}^{\mathrm{o}}(t)$. A collision event can then be defined by a binary indicator function

$$I_{\mathrm{C}}\left(\mathbf{x}^{\mathrm{e}}, \mathbf{x}^{\mathrm{o}}\right) = \begin{cases} 1 & \mathcal{G}\left(\mathbf{x}^{\mathrm{e}}\right) \cap \mathcal{G}\left(\mathbf{x}^{\mathrm{o}}\right) \neq \emptyset \\ 0 & \mathcal{G}\left(\mathbf{x}^{\mathrm{e}}\right) \cap \mathcal{G}\left(\mathbf{x}^{\mathrm{o}}\right) = \emptyset \end{cases} \tag{5.43}$$

which is based on the object geometries $\mathcal{G}\left(\mathbf{x}^{\mathrm{e}}\right)$ and $\mathcal{G}\left(\mathbf{x}^{o}\right)$.

It is assumed that the dynamic states $\mathbf{x}_k^{\mathrm{e}}$, $\mathbf{x}_k^{\mathrm{o}}$ at the current time t_k are known and a collision has not yet occurred. Then, a criticality measure $\kappa\left(\mathbf{x}_k^{\mathrm{e}}, \mathbf{x}_k^{\mathrm{o}}\right)$ can be defined as a norm $\left\|\mathbf{u}^{\mathrm{e}}\left(t\right)\right\|$ of the minimum driver[10] inputs $\mathbf{u}_{\min}^{\mathrm{e}} \leq \mathbf{u}^{\mathrm{e}}\left(t\right) \leq \mathbf{u}_{\max}^{\mathrm{e}}$ that are necessary to avoid a collision in the future. In practice, the problem formulation is limited to a time interval $\left[t_k, t_k + \bar{T}\right]$ where $\bar{T}$ denotes a pre-defined prediction horizon [83]:

$$\begin{aligned} \kappa\left(\mathbf{x}_k^{\mathrm{e}}, \mathbf{x}_k^{\mathrm{o}}\right) = \underset{\mathbf{u}^{\mathrm{e}}(t)}{\text{minimise}} \quad & \left\|\mathbf{u}^{\mathrm{e}}\left(t\right)\right\| \\ \text{subject to} \quad & I_{\mathrm{C}}\left(\mathbf{x}^{\mathrm{e}}\left(t\right), \mathbf{x}^{\mathrm{o}}\left(t\right)\right) = 0 \; \forall t \in \left[t_k, t_k + \bar{T}\right] \; . \end{aligned} \tag{5.44a}$$

The future states[11] of ego-vehicle and object are given by prediction models as follows:

$$\dot{\mathbf{x}}^{\mathrm{e}}\left(t\right) = \mathbf{f}^{\mathrm{e}}\left(\mathbf{x}^{\mathrm{e}}\left(t\right), \mathbf{u}^{\mathrm{e}}\left(t\right)\right) \; , \qquad \mathbf{x}^{\mathrm{e}}\left(t_k\right) = \mathbf{x}_k^{\mathrm{e}} \; , \qquad \mathbf{u}_{\min}^{\mathrm{e}} \leq \mathbf{u}^{\mathrm{e}}\left(t\right) \leq \mathbf{u}_{\max}^{\mathrm{e}} \; , \tag{5.44b}$$

$$\dot{\mathbf{x}}^{\mathrm{o}}\left(t\right) = \mathbf{f}^{\mathrm{o}}\left(\mathbf{x}^{\mathrm{o}}\left(t\right)\right) \; , \qquad \mathbf{x}^{\mathrm{o}}\left(t_k\right) = \mathbf{x}_k^{\mathrm{o}} \; . \tag{5.44c}$$

The problem formulated in (5.44) is very general and in order to obtain closed-form criticality measures, further reformulations are introduced in [83, 95]:

1. Instead of considering two motion states $\mathbf{x}^{\mathrm{e}}\left(t\right)$ and $\mathbf{x}^{\mathrm{o}}\left(t\right)$ in ground-fixed coordinates, a simplification is achieved by using the motion state $\mathbf{x}\left(t\right)$ in a coordinate system relative to the moving ego-vehicle. A prediction model in relative dynamics can be obtained as discussed in Sec. 4.2.7. It is assumed that the state prediction can be expressed in additive form:

$$\mathbf{x}\left(t_k + T\right) = \mathbf{\Phi}\left(t_k + T, t_k\right) \mathbf{x}\left(t_k\right) + \int_{t_k}^{t_k+T} \mathbf{B}\left(\tau\right) \mathbf{u}^{\mathrm{e}}\left(\tau\right) \; \mathrm{d}\tau \; . \tag{5.45}$$

   A collision indicator function in relative coordinates is denoted by $I_{\mathrm{C}}\left(\mathbf{x}\right)$ and is used instead of (5.43). To cope with explicit dependences on the absolute states, for example the orientation angles, rotationally invariant enclosing hulls [28] can be used to approximate the exact shapes.

2. Secondly, the ego-vehicle's driver input vector $\mathbf{u}^{\mathrm{e}}\left(t\right)$ is evaluated component-wise which yields individual criticality metrics per input $u_i^{\mathrm{e}}\left(t\right)$. Evasive actions by accelerating or braking and steering to the left or right are thereby treated separately. For example, the required acceleration to avoid a collision by a longitudinal manoeuvre and the yaw rate for a lateral evasive manoeuvre can be calculated individually and combined to an overall figure later on.[12] Since each input is bounded from

[10] It is important to note that all vehicle inputs refer to the possibilities available to a human driver and not the evasive actions which can be performed by the AEB system. If a situation is critical with respect to the braking deceleration alone but can still be resolved with moderate steering effort, for example a lane change, an autonomous brake intervention is in general not justified.

[11] In this thesis, criticality measures are derived with trajectories being represented in Cartesian coordinates, whereas a lane-centred coordinate system is employed for example in [53].

[12] However, the assumption that no evasive manoeuvre of combined braking and steering exists if neither braking nor steering alone suffice to avoid a collision is violated in some cases [169].

above and below according to (5.44b), the problem is furthermore split into the cases $0 \leq u_i^{\mathrm{e}}(t) \leq u_{\mathrm{max},i}^{\mathrm{e}}$ and $0 \leq -u_i^{\mathrm{e}}(t) \leq -u_{\mathrm{min},i}^{\mathrm{e}}$.

Moreover, instead of a general time-dependent input $u_i^{\mathrm{e}}(t)$, a compact parametrisation is usually introduced. For example, a constant value $u_i^{\mathrm{e}}(t) = u_i^{\mathrm{e}}$ can be assumed. Another possibility, pursued for the derivation of the *time-to-react* metric in [83], is to parametrise the bounded inputs in terms of the time t_{B} at which $u_i^{\mathrm{e}}(t)$ has to take on its maximum possible value while it remains zero before. Thus, the remaining times until a brake-, acceleration-, or steering-manoeuvre has to be performed are obtained and the maximum over these yields the overall *time-to-react*.

3. Thirdly, the optimisation problem in (5.44) of finding the smallest driver input such that no collision occurs for $t \in \left[t_k, t_k + \bar{T}\right]$ is reformulated. To this end, the problem is equivalently expressed as reaching a specific relative motion state $\boldsymbol{\psi}(\mathbf{x}(t_k + T)) = \boldsymbol{\gamma}$ at a future time $t_k + T$ conditional that no collision has occurred until that time instead:

$$\begin{aligned} &\underset{u_i^{\mathrm{e}}(t)}{\text{minimise}} \quad |u_i^{\mathrm{e}}(t)| \quad \text{subject to} \quad I_{\mathrm{C}}(\mathbf{x}(t)) = 0 \; \forall t \in \left[t_k, t_k + \bar{T}\right] \\ &\iff \underset{u_i^{\mathrm{e}}(t)}{\mathrm{sol}} \left\{ \boldsymbol{\psi}(\mathbf{x}(t_k + T)) = \boldsymbol{\gamma} \wedge I_{\mathrm{C}}(\mathbf{x}(t)) = 0 \; \forall t \in [t_k, t_k + T) \right\} . \end{aligned} \tag{5.46}$$

Therefore, $\boldsymbol{\psi}(\cdot)$ and $\boldsymbol{\gamma}$ define a condition so that a collision is just avoided. For example, a rear-end collision between two vehicles is just avoided by braking, if the relative velocity has dropped to zero at the moment that the distance is zero. This condition can be easily extended to obey additional safety margins (5.43). Solving (5.46) yields a criticality value $\kappa(\mathbf{x}_k)$ and the corresponding prediction horizon T_k^* as functions of $\mathbf{x}_k$.

Developing algorithms for criticality assessment from a general problem formulation similar to (5.46) is not new but has been employed for example in [83, 95]. However, this generic framework has to the best of the author's knowledge not been used before to derive analytical methods for uncertainty propagation as will be shown in Sec. 5.2.4.

In the following, three examples of criticality measures are presented to illustrate the framework. For clarity, only longitudinal traffic scenarios with an impending rear-end collision are considered. The motion of the target object relative to the ego-vehicle is described by a state vector $\mathbf{x}$ which comprises longitudinal distance x, relative velocity v_x and relative acceleration a_x. At the initial time step t_k a negative relative velocity $v_{x,k} < 0$ is assumed. It is furthermore assumed that the relative lateral velocity v_y is negligible and the lateral position y therefore remains in the driving corridor $|y| \leq y_{\mathrm{coll}}$.

Example 5.3 (Time-to-collision (TTC))

The relative longitudinal dynamics are described by a constant velocity model ($a_x = 0$). Its time-evolution has been previously derived in example 4.4 and reads:

$$\begin{bmatrix} x(t_k + T) \\ v_x(t_k + T) \end{bmatrix} = \begin{bmatrix} 1 & T \\ 0 & 1 \end{bmatrix} \begin{bmatrix} x(t_k) \\ v_x(t_k) \end{bmatrix} . \tag{5.47}$$

Without considering any potential evasive action, the *time-to-collision* metric can be defined as the solution (5.46) at which the distance $x\left(t_k+T\right)$ becomes zero:

$$t_{\mathrm{tc}}\left(\mathbf{x}_k\right)=\underset{T}{\mathrm{sol}}\left\{x\left(t_k+T\right)=0\right\}=\underset{T}{\mathrm{sol}}\left\{x_k+v_{x,k}\cdot T=0\right\}=-\frac{x_k}{v_{x,k}}\,. \tag{5.48}$$

Example 5.4 (Brake-threat-number (BTN))

The relative dynamics are modelled with constant accelerations. In order to simplify the derivations, a negative relative initial acceleration $a_{x,k}\leq 0$ is assumed. This includes the practically relevant case of driving behind a preceding vehicle where the driver is forced to brake suddenly due to an unforeseen situation.[13] A constant input $u^{\mathrm{e}}\left(t\right)=a_x^{\mathrm{e}}$ models the ego-vehicle's brake deceleration:

$$\begin{bmatrix} x\left(t_k+T\right)\\ v_x\left(t_k+T\right)\\ a_x\left(t_k+T\right)\end{bmatrix}=\begin{bmatrix}1 & T & \frac{1}{2}T^2\\ 0 & 1 & T\\ 0 & 0 & 1\end{bmatrix}\begin{bmatrix} x\left(t_k\right)\\ v_x\left(t_k\right)\\ a_x\left(t_k\right)-a_x^{\mathrm{e}}\end{bmatrix}\,. \tag{5.49}$$

To avoid a collision by braking, the relative velocity v_x needs to be reduced to zero when the distance approaches zero. This yields the following formulation of (5.46):

$$a_{x,\mathrm{req}}^{\mathrm{e}}\left(\mathbf{x}_k\right)=\underset{a_x^{\mathrm{e}}}{\mathrm{sol}}\left\{\begin{bmatrix}x_k+v_{x,k}\cdot T+\frac{1}{2}\left(a_{x,k}-a_x^{\mathrm{e}}\right)\cdot T^2\\ v_{x,k}+\left(a_{x,k}-a_x^{\mathrm{e}}\right)\cdot T\end{bmatrix}=\begin{bmatrix}0\\0\end{bmatrix}\right\}=a_{x,k}-\frac{v_{x,k}^2}{2x_k}\,. \tag{5.50}$$

The prediction horizon is thereby obtained as $T_k^*=-2\frac{x_k}{v_{x,k}}$.

Finally, in order to obtain a normalised criticality measure, $a_{x,\mathrm{req}}^{\mathrm{e}}\left(\mathbf{x}_k\right)$ is usually divided by a minimum acceleration $a_{x,\min}^{\mathrm{e}}<0$. This yields the dimension-less *brake-threat-number*, which can be compared to other evasion strategies, for example by steering. The maximum braking deceleration $a_{x,\min}^{\mathrm{e}}$ depends on the tire-road friction coefficient and may be estimated on-line [216] or set to a constant overapproximation that corresponds to ideal conditions.

Example 5.5 (Time-to-brake (TTB))

As a third measure of how critical a situation is with respect to braking, the *time-to-brake* t_{tb} is considered. Only the case of an initially unaccelerated motion ($a_x=0$) is studied here but further details on other cases can be found in [81]. The deceleration

[13] A number of special cases are introduced by the assumption of an accelerated relative motion. For example, the velocity of an already decelerating vehicle will only decrease until it reaches standstill. It is not in the scope of this work to discuss all of these cases, but it is referred to [83, 95] for a comprehensive treatment.

of the ego-vehicle is parametrised via the time t_B at which the maximum brake deceleration $a^\mathrm{e}_{x,\min} < 0$ is applied:

$$u^\mathrm{e}(t) = \begin{cases} 0 & t < t_\mathrm{B} \\ a^\mathrm{e}_{x,\min} & t \geq t_\mathrm{B} \end{cases} . \tag{5.51}$$

Similar to (5.50), the following conditions define the TTB:

$$\begin{aligned} t_\mathrm{tb}(\mathbf{x}_k) &= \underset{t_\mathrm{B}}{\mathrm{sol}} \left\{ \begin{bmatrix} x_k + v_{x,k} \cdot T - \frac{1}{2} a_{x,\min} \cdot (T - t_\mathrm{B})^2 \\ v_{x,k} - a^\mathrm{e}_{x,\min} \cdot (T - t_\mathrm{B}) \end{bmatrix} = \begin{bmatrix} 0 \\ 0 \end{bmatrix} \right\} \\ &= -\frac{x_k}{v_{x,k}} - \frac{v_{x,k}}{2a^\mathrm{e}_{x,\min}} . \end{aligned} \tag{5.52}$$

The time that remains before a full-brake manoeuvre has to be initiated is thus naturally smaller than the TTC from (5.48). Values $t_\mathrm{tb}(\mathbf{x}_k) \leq 0$ result if $a^\mathrm{e}_{x,\mathrm{req}}(\mathbf{x}_k) \leq a^\mathrm{e}_{x,\min}$ in (5.50).

It has been shown that different criticality measures which are reported in the literature can be expressed in a general criterion (5.46) together with a prediction model (5.45). In the next section, this representation will be used to derive probabilistic models of uncertainty in the criticality estimates. Both errors in the estimated state $\mathbf{x}_k$ and the prediction $\mathbf{x}(t_k + T^*_k)$ are considered.

While only basic examples of criticality measures have been shown here, more sophisticated formulae can be derived in the same way, if higher order motion models are employed.

5.2.4 Statistical modelling of criticality measures

Having introduced the necessary background on criticality measures, statistical models of these algorithms are developed in this section. Firstly, the error between the estimated and the true criticality value will be analysed in Sec. 5.2.4.1. Such deviations occur due to errors in the estimated states but also due to deviations between a predicted future trajectory and the true one. The framework will be exemplified for the *time-to-collision* and the *brake-threat-number* as previously introduced in example 5.3 and 5.4, respectively. Here, simple closed-form models are obtained. Secondly, correlations between errors over time are subsequently modelled in Sec. 5.2.4.2.

Thirdly, a probabilistic model of the time at which the estimated criticality exceeds an activation threshold is derived in Sec. 5.2.4.3. In a previous analysis [83], only independent errors are assumed. If correlated errors are taken into account, a more general approach is required. As will be shown, the results developed here form a generalisation of previous works.

Lastly, the fact that criticality measures are often defined piece-wise in order to avoid singularities is included in the probabilistic model. For example, the *time-to-collision*

metric is only defined between two approaching vehicles and becomes negative or infinite for relative velocities greater than or equal to zero. This corresponds to the case that no future collision is predicted from the current state. Therefore, probabilistic modelling of criticality measures relates to determining the probability of a collision between two extended objects. In general situations, different approaches based on Monte-Carlo methods are known. A novel closed-form approximation that is tailored to rear-end situations will be developed in Sec. 5.2.4.4.

5.2.4.1 Uncertainty propagation in criticality measures

A Gaussian estimate $\mathbf{x}_k \sim \mathcal{N}(\hat{\mathbf{x}}_k, \boldsymbol{\Sigma}_\mathbf{x}(t_k))$ of the dynamic motion state is assumed, from which a criticality measure $\kappa(\mathbf{x}_k)$ is calculated. In order to model the influence of uncertainties on the criticality measure, Gaussian error propagation can be used to obtain the distribution $\kappa(\mathbf{x}_k) \sim \mathcal{N}\left(\mu_\kappa(t_k), \sigma_\kappa^2(t_k)\right)$. To first order, the expected value is $\mu_\kappa(t_k) = \kappa(\hat{\mathbf{x}}_k)$. In a classical approach, the variance $\sigma_\kappa^2(t_k)$ is calculated by linearising the explicit function $\kappa(\mathbf{x}_k)$ around the expected value $\hat{\mathbf{x}}_k$. As introduced in Sec. 2.2, the covariance of $\mathbf{x}_k$ is then propagated as follows:

$$\sigma_\kappa^2(t_k) = \left(\nabla_{\mathbf{x}_k}\kappa(\mathbf{x}_k)\big|_{\hat{\mathbf{x}}_k}\right) \cdot \boldsymbol{\Sigma}_\mathbf{x}(t_k) \cdot \left(\nabla_{\mathbf{x}_k}\kappa(\mathbf{x}_k)\big|_{\hat{\mathbf{x}}_k}\right)^\mathrm{T} . \tag{5.53}$$

However, only the initial state uncertainty but not the prediction errors are taken into account with this classical approach. In contrast, the implicit prediction over a state-dependent time horizon T_k^* is made explicit by (5.46). Assuming that this prediction model is given in the form of (5.45), uncertainties can be modelled as additive Gaussian white process noise. As has been previously shown in Sec. 4.2.6, a Gaussian model of the predicted state at a future time instant $t_k + T$ can be inferred. The resulting covariance is recalled here:

$$\boldsymbol{\Sigma}_\mathbf{x}(t_k + T) = \boldsymbol{\Phi}(t_k + T, t_k)\,\boldsymbol{\Sigma}_\mathbf{x}(t_k)\,\boldsymbol{\Phi}^\mathrm{T}(t_k + T, t_k) + \mathbf{Q}(t_k + T, t_k) \ , \tag{5.54a}$$

$$\text{with} \qquad \mathbf{Q}(t_k + T, t_k) = \int_{t_k}^{t_k+T} \boldsymbol{\Phi}(t_k + T, \tau)\,\mathbf{L}\mathbf{S}\mathbf{L}^\mathrm{T}\boldsymbol{\Phi}^\mathrm{T}(t_k + T, \tau)\ \mathrm{d}\tau \ . \tag{5.54b}$$

Therefore, one obtains the variance $\sigma_\kappa^2(t_k)$ under consideration of both state estimation and prediction uncertainty by linearising the criticality measure around the mean of the predicted state $\hat{\mathbf{x}}(t_k + T_k^*)$.

To this end, the unknown variables in the criticality definition (5.46), which usually comprise[14] the prediction horizon T_k^* and the criticality measure $\kappa(\mathbf{x}_k)$, are written in the form of a vector $\boldsymbol{\alpha} \coloneqq \begin{bmatrix} T_k^* & \kappa(\mathbf{x}_k) \end{bmatrix}^\mathrm{T}$. In order to obtain the Jacobian with respect to $\mathbf{x}(t_k + T_k^*)$, the implicit function theorem is applied [40]:

$$\nabla_{\mathbf{x}\left(t_k+T_k^*\right)}\boldsymbol{\alpha} = -\Big[\nabla_{\boldsymbol{\alpha}}\left(\boldsymbol{\psi}\left(\mathbf{x}(t_k + T_k^*)\right) - \boldsymbol{\gamma}\right)\Big]^{-1}\Big[\nabla_{\mathbf{x}\left(t_k+T_k^*\right)}\left(\boldsymbol{\psi}\left(\mathbf{x}(t_k + T_k^*)\right) - \boldsymbol{\gamma}\right)\Big] \ . \tag{5.55}$$

[14] For the *time-to-collision* metric (5.48), the prediction horizon T_k^* itself defines the criticality.

With the Jacobian evaluated at the expectations $\mathbb{E}\left[T_k^*\right] \approx T_k^*\left(\hat{\mathbf{x}}_k\right)$, $\kappa\left(\hat{\mathbf{x}}_k\right)$ and $\hat{\mathbf{x}}\left(t_k + T_k^*\right)$, the covariance of $\boldsymbol{\alpha}$ is obtained as:

$$\boldsymbol{\Sigma}_{\boldsymbol{\alpha}} = \begin{bmatrix} \sigma_{T_k^*}^2 & \rho_{T_k^*\kappa} \\ \rho_{\kappa T_k^*} & \sigma_\kappa^2 \end{bmatrix} = \left(\nabla_{\mathbf{x}\left(t_k+T_k^*\right)} \boldsymbol{\alpha} \Big|_{\hat{\mathbf{x}}\left(t_k+T_k^*\right)} \right) \cdot \boldsymbol{\Sigma}_{\mathbf{x}}\left(t_k + T_k^*\right) \cdot \left(\nabla_{\mathbf{x}\left(t_k+T_k^*\right)} \boldsymbol{\alpha} \Big|_{\hat{\mathbf{x}}\left(t_k+T_k^*\right)} \right)^{\mathrm{T}}. \tag{5.56}$$

The element σ_κ^2 contains the desired result. This general framework will be illustrated for the *time-to-collision* from example 5.3 and the *brake-threat-number* that was introduced in example 5.4.

Example 5.6 (Uncertainty propagation in TTC)

According to the definition in (5.48), the predicted time T_k^* at which a collision occurs defines a criticality measure $t_{\mathrm{tc}}\left(\mathbf{x}_k\right)$. This metric is calculated from a state vector $\mathbf{x}_k = \begin{bmatrix} x_k & v_{x,k} \end{bmatrix}^{\mathrm{T}}$ which comprises the distance and relative velocity. Calculating the gradient (5.55) of T_k^* with respect to the predicted state $\mathbf{x}\left(t_k + T_k^*\right)$ yields:

$$\nabla_{\mathbf{x}\left(t_k+T_k^*\right)} T_k^* = -\left[\nabla_{T_k^*}\left(x_k + v_{x,k} \cdot T_k^*\right)\right]^{-1} \cdot \left[\nabla_{\mathbf{x}\left(t_k+T_k^*\right)} x\left(t_k + T_k^*\right)\right] = \begin{bmatrix} -\frac{1}{v_{x,k}} & 0 \end{bmatrix}. \tag{5.57}$$

The predicted state covariance $\boldsymbol{\Sigma}_{\mathbf{x}}\left(t_k + T_k^*\right)$ of the underlying CV model is given in (4.74b). Together with the gradient (5.57), this yields the following closed-form result for (5.56):

$$\begin{aligned} \operatorname{var}\left(t_{\mathrm{tc}}\left(\mathbf{x}_k\right)\right) &= \left(\nabla_{\mathbf{x}_k} t_{\mathrm{tc}}\left(\mathbf{x}_k\right) \big|_{\hat{\mathbf{x}}_k} \right) \cdot \boldsymbol{\Sigma}_{\mathbf{x}}\left(t_k\right) \cdot \left(\nabla_{\mathbf{x}_k} t_{\mathrm{tc}}\left(\mathbf{x}_k\right) \big|_{\hat{\mathbf{x}}_k} \right)^{\mathrm{T}} - \frac{1}{3\hat{v}_{x,k}^2} \left(t_{\mathrm{tc}}\left(\hat{\mathbf{x}}_k\right)\right)^3 S_x \\ \text{where} \qquad \nabla_{\mathbf{x}_k} t_{\mathrm{tc}}\left(\mathbf{x}_k\right) \big|_{\hat{\mathbf{x}}_k} &= \begin{bmatrix} -\frac{1}{\hat{v}_{x,k}} & \frac{\hat{x}_k}{\hat{v}_{x,k}^2} \end{bmatrix}. \end{aligned} \tag{5.58}$$

This can be identified as the classical result (5.53) plus an additional term for the process noise influence. The latter is proportional to $\left(t_{\mathrm{tc}}\left(\hat{\mathbf{x}}_k\right)\right)^3$ and inversely proportional to $3\hat{v}_{x,k}^2$. Therefore, the criticality estimate is less affected by prediction uncertainties in situations with small $t_{\mathrm{tc}}\left(\hat{\mathbf{x}}_k\right)$ as well as those with high speeds $v_{x,k}$.

Example 5.7 (Uncertainty propagation in BTN)

The definition of the function $\boldsymbol{\psi}\left(\mathbf{x}\left(t_k + T\right)\right)$ and the terminal state $\boldsymbol{\gamma}$ are given in (5.50). This time, the state $\mathbf{x}_k = \begin{bmatrix} x_k & v_{x,k} & a_{x,k} \end{bmatrix}^{\mathrm{T}}$ includes the relative acceleration $a_{x,k}$. The Jacobian (5.55) is obtained as follows where $\boldsymbol{\alpha} \coloneqq \begin{bmatrix} T_k^* & a_{x,\mathrm{req}}^{\mathrm{e}}\left(\mathbf{x}_k\right) \end{bmatrix}^{\mathrm{T}}$:

$$\nabla_{\mathbf{x}\left(t_k+T_k^*\right)} \boldsymbol{\alpha} = -\begin{bmatrix} v_{x,k} + \left(a_{x,k} - a_{x,\mathrm{req}}^{\mathrm{e}}\left(\mathbf{x}_k\right)\right) \cdot T_k^* & -\frac{T_k^{*2}}{2} \\ a_{x,k} - a_{x,\mathrm{req}}^{\mathrm{e}}\left(\mathbf{x}_k\right) & -T_k^* \end{bmatrix}^{-1} \cdot \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \end{bmatrix}. \tag{5.59}$$

Due to the assumption of zero mean noise processes, (5.50) yields the expected values of T_k^* and $a_{x,\mathrm{req}}^{\mathrm{e}}\left(\hat{\mathbf{x}}_k\right)$ in terms of $\hat{\mathbf{x}}_k$. Inserting these into the second row of the Jacobian (5.59) yields the desired gradient of $a_{x,\mathrm{req}}^{\mathrm{e}}$:

$$\nabla_{\mathbf{x}\left(t_k+T_k^*\right)} a_{x,\mathrm{req}}^{\mathrm{e}}\left(\mathbf{x}_k\right)\Big|_{\hat{\mathbf{x}}\left(t_k+T_k^*\right)} = \begin{bmatrix} \frac{\hat{v}_{x,k}^2}{2\hat{x}_k^2} & 0 & 0 & 0 \end{bmatrix} . \tag{5.60}$$

The covariance $\boldsymbol{\Sigma}_{\mathbf{x}}\left(t_k+T\right)$ of a state prediction in the constant acceleration model follows according to (4.72):

$$\boldsymbol{\Sigma}_{\mathbf{x}}\left(t_k+T\right) = \begin{bmatrix} 1 & T & \frac{1}{2}T^2 \\ 0 & 1 & T \\ 0 & 0 & 1 \end{bmatrix} \cdot \boldsymbol{\Sigma}_{\mathbf{x}}\left(t_k\right) \cdot \begin{bmatrix} 1 & T & \frac{1}{2}T^2 \\ 0 & 1 & T \\ 0 & 0 & 1 \end{bmatrix}^{\mathrm{T}} + \begin{bmatrix} \frac{1}{20}T^5 & \frac{1}{8}T^4 & \frac{1}{6}T^3 \\ \frac{1}{8}T^4 & \frac{1}{3}T^3 & \frac{1}{2}T^2 \\ \frac{1}{6}T^3 & \frac{1}{2}T^2 & T \end{bmatrix} S_x \, . \tag{5.61}$$

Together with the gradient (5.60), this yields the following closed-form result:

$$\begin{aligned} &\mathrm{var}\left(a_{x,\mathrm{req}}\left(\mathbf{x}_k\right)\right) = \left(\nabla_{\mathbf{x}_k} a_{x,\mathrm{req}}\left(\mathbf{x}_k\right)\big|_{\hat{\mathbf{x}}_k}\right) \cdot \boldsymbol{\Sigma}_{\mathbf{x}}\left(t_k\right) \cdot \left(\nabla_{\mathbf{x}_k} a_{x,\mathrm{req}}\left(\mathbf{x}_k\right)\big|_{\hat{\mathbf{x}}_k}\right)^{\mathrm{T}} - \frac{2\hat{x}_k}{5\hat{v}_{x,k}} S_x \\ &\text{where} \qquad \nabla_{\mathbf{x}_k} a_{x,\mathrm{req}}\left(\mathbf{x}_k\right)\big|_{\hat{\mathbf{x}}_k} = \begin{bmatrix} \frac{\hat{v}_{x,k}^2}{2\hat{x}_k^2} & -\frac{\hat{v}_{x,k}}{\hat{x}_k} & 1 \end{bmatrix} . \end{aligned} \tag{5.62}$$

The first term of this expression equals the result which is obtained from the classical approach (5.53) whereas the second term denotes the additional influence of the process noise.

To summarise the result, (5.56) provides a closed-form expression for the combined propagation of Gaussian state estimation and prediction errors to a criticality measure. The general result has been exemplified for the TTC and BTN criticality measures. It has been shown that the resulting compact models are generalisations of the classical approach (5.53).

Concerning the interpretation of the obtained variance it has to be noted that the true criticality is different for each realisation of a stochastic trajectory. If only the state estimate is uncertain on the other hand, the variance propagation describes the errors compared to the one true criticality value. In contrast, the variance under state estimation and prediction uncertainty describes the distribution of the predicted criticality given the uncertain state and models. Modelling this distribution is thus useful for Bayesian decision making.

5.2.4.2 Correlated errors

In order to model uncertainties over multiple time steps, correlations have to be analysed in addition to the previous models.

It is assumed that two Gaussian state estimates $\mathbf{x}_k \sim \mathcal{N}(\hat{\mathbf{x}}_k, \mathbf{\Sigma}_\mathbf{x}(t_k))$ and $\mathbf{x}_{k+1} \sim \mathcal{N}(\hat{\mathbf{x}}_{k+1}, \mathbf{\Sigma}_\mathbf{x}(t_{k+1}))$ from time steps $t_{k+1} > t_k$ are available. Moreover, these estimates are assumed to be correlated with a cross-covariance $\mathbf{\Sigma}_\mathbf{x}(t_k, t_{k+1}) \coloneqq \mathrm{cov}(\mathbf{x}_k, \mathbf{x}_{k+1})$. Correlation may be induced if state estimates are formed by aggregating sensor measurement information from multiple time steps. Even though the individual measurements might be uncorrelated, the state estimates may become correlated.

A second source of correlations is due to the influence of prediction errors on the criticality estimate. Starting from the first time step t_k, a prediction of the future state $\mathbf{x}(t_k + T_k^*)$ is calculated. When a prediction is started from a later time step t_{k+1} until $t_{k+1} + T_{k+1}^*$, parts of the first trajectory are included in the second prediction as well.[15] Hence, the prediction errors made for a specific trajectory will be partly identical and thus correlations are caused.

The variance of a criticality measure $\kappa_k \coloneqq \kappa(\mathbf{x}_k)$ is given in (5.56). A similar approach is now pursued to obtain the cross-covariance $\mathrm{cov}(\kappa_k, \kappa_{k+1})$ of κ_k and κ_{k+1}. First, the cross-covariance $\mathbf{\Sigma}_\mathbf{x}(t_k + T_k^*, t_{k+1} + T_{k+1}^*)$ between the two predicted states $\mathbf{x}(t_k + T_k^*)$ and $\mathbf{x}(t_{k+1} + T_{k+1}^*)$ is calculated. The criticality values κ_k and κ_{k+1} are implicitly based on these state predictions. Secondly, the cross-covariance between the state predictions is propagated to criticality estimates by linearisation.

It is assumed that the state prediction model can be expressed in the form of (4.72) where $\mathbf{w}(t)$ is white Gaussian process noise that is independent of the state estimates. One obtains for the first prediction from t_k until $t_k + T_k^*$ [107]:

$$\mathbf{x}(t_k + T_k^*) = \mathbf{\Phi}(t_k + T_k^*, t_k)\,\mathbf{x}(t_k) + \int_{t_k}^{t_k + T_k^*} \mathbf{\Phi}(t_k + T_k^*, \tau)\,\mathbf{L}\mathbf{w}(\tau)\;\mathrm{d}\tau \tag{5.63a}$$

The second prediction starts at t_{k+1} and reads:

$$\mathbf{x}(t_{k+1} + T_{k+1}^*) = \mathbf{\Phi}(t_{k+1} + T_{k+1}^*, t_{k+1})\,\mathbf{x}(t_{k+1}) + \int_{t_{k+1}}^{t_{k+1} + T_{k+1}^*} \mathbf{\Phi}(t_{k+1} + T_{k+1}^*, \tau)\,\mathbf{L}\mathbf{w}(\tau)\;\mathrm{d}\tau\,. \tag{5.63b}$$

When the cross-covariance between $\mathbf{x}(t_k + T_k^*)$ and $\mathbf{x}(t_{k+1} + T_{k+1}^*)$ is calculated, all products of the state estimates and $\mathbf{w}(t)$ as well as integrals of $\mathbf{w}(t)$ with non-overlapping boundaries vanish due to the assumption of white noise processes. The overall result comprises the cross-covariance $\mathbf{\Sigma}_\mathbf{x}(t_k, t_{k+1})$ between the two state estimates as well as

[15] Without loss of generality, it is assumed that $T_k^* \geq T_{k+1}^*$ here. That is, the situation becomes more critical as time progresses and thus the prediction horizon T_{k+1}^* is smaller than T_k^*. Since $t_{k+1} > t_k$, the state predictions over the time interval $\left[t_{k+1}, t_{k+1} + T_{k+1}^*\right]$ are included in both estimates. However, the derivation can be performed in a similar way if this overlapping interval has to be defined with $T_k^* \leq T_{k+1}^*$.

a second term which relates to the process noise during the common part of the two predictions:

$$\begin{aligned}\boldsymbol{\Sigma}_{\mathbf{x}}\left(t_k+T_k^*,t_{k+1}+T_{k+1}^*\right) = \boldsymbol{\Phi}\left(t_k+T_k^*,t_k\right)\boldsymbol{\Sigma}_{\mathbf{x}}\left(t_k,t_{k+1}\right)\boldsymbol{\Phi}^{\mathrm{T}}\left(t_{k+1}+T_{k+1}^*,t_{k+1}\right)\\ +\boldsymbol{\Phi}\left(t_k+T_k^*,t_{k+1}+T_{k+1}^*\right)\mathbf{Q}\left(t_{k+1}+T_{k+1}^*,t_{k+1}\right)\end{aligned}. \tag{5.64}$$

The process noise covariance $\mathbf{Q}\left(t_{k+1}+T_{k+1}^*,t_{k+1}\right)$ is defined in (5.54b). Finally, the cross-covariance of the criticality is obtained by linearisation where the gradients have been derived in (5.55):

$$\begin{aligned}\operatorname{cov}\left(\kappa_k,\kappa_{k+1}\right) = &\left(\nabla_{\mathbf{x}\left(t_k+T_k^*\right)}\kappa\left(\mathbf{x}_k\right)\Big|_{\hat{\mathbf{x}}\left(t_k+T_k^*\right)}\right)\cdot\boldsymbol{\Sigma}_{\mathbf{x}}\left(t_k+T_k^*,t_{k+1}+T_{k+1}^*\right)\\ &\cdot\left(\nabla_{\mathbf{x}\left(t_{k+1}+T_{k+1}^*\right)}\kappa\left(\mathbf{x}_{k+1}\right)\Big|_{\hat{\mathbf{x}}\left(t_{k+1}+T_{k+1}^*\right)}\right)^{\mathrm{T}}.\end{aligned} \tag{5.65}$$

The obtained models of the distribution at a single time instant yield the probability that a criticality measure exceeds an activation threshold. It is thus possible to model the time of activation, which will be studied in the following section.

5.2.4.3 Uncertainty in decision timing

We will now consider situations that span multiple time steps $t_i,\ i=1,\ldots,k$. An object is measured and tracked by the ego-vehicle during this time interval, which leads to a sequence of Gaussian state estimates $\mathbf{x}_1,\ldots,\mathbf{x}_k$ with means $\hat{\mathbf{x}}_1,\ldots,\hat{\mathbf{x}}_k$ and estimation error covariances $\boldsymbol{\Sigma}_{\mathbf{x}}\left(t_1\right),\ldots,\boldsymbol{\Sigma}_{\mathbf{x}}\left(t_k\right)$. The cross-covariances $\boldsymbol{\Sigma}_{\mathbf{x}}\left(t_1,t_2\right),\ldots,\boldsymbol{\Sigma}_{\mathbf{x}}\left(t_{k-1},t_k\right)$ describe the correlation between subsequent state estimates.

At each time instant, a criticality measure $\kappa_i \coloneqq \kappa\left(\mathbf{x}_i\right)$ is calculated from these estimates. We assume that the considered traffic situation is critical and it is therefore desired that an emergency brake intervention is activated at a time t_{B}. In terms of the criticality measure, this means that $\kappa\left(\mathbf{x}\left(t_{\mathrm{B}}\right)\right) > \kappa_{\max}$ where $\kappa_{\max}$ is a pre-defined threshold.[16]

A criticality measure that is calculated from uncertain state estimates $\mathbf{x}_i$ becomes a random variable $\kappa\left(\mathbf{x}_i\right)$. It has been shown in the previous sections how a Gaussian approximation $\kappa_i \sim \mathcal{N}\left(\mu_\kappa\left(t_i\right),\sigma_\kappa^2\left(t_i\right)\right)$ can be derived. Due to the randomness of the sequence $\kappa_1,\ldots,\kappa_k$, the time of activation t_{B} is random as well. The following analysis therefore focusses on finding the probability $\mathrm{P}\left(t_{\mathrm{B}} \leq t_k\right)$ that the emergency brake has been activated by a time step t_k. If this distribution is shifted by the ideal time of activation, one obtains the distribution of the activation jitter Δt_{B} which is an indicator of how reliable the system acts.

The key information for finding $\mathrm{P}\left(t_{\mathrm{B}} \leq t_k\right)$ is the distribution of the sequence of criticality estimates $\kappa_1,\ldots,\kappa_k$. In a previous investigation [83], statistically independent

[16] In the general case, a $\kappa\left(\mathbf{x}\right) \geq \kappa_{\max}$ threshold decision will be assumed whereas for specific examples such as the TTC, falling below a lower value defines an activation event. Since both formulations can be transformed to each other, the choice is determined by easier interpretation of a criticality measure.

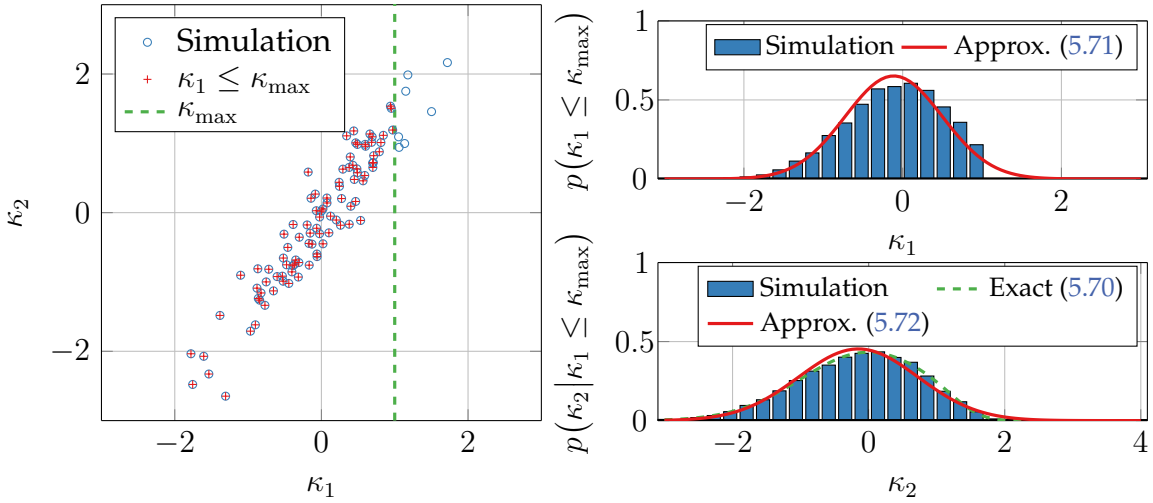


Figure 5.5 Example of truncating a bivariate Gaussian with $\kappa_1 \sim \mathcal{N}(0, 0.5)$, $\kappa_2 \sim \mathcal{N}(1, 1)$ and $\mathrm{cov}\,(\kappa_1, \kappa_2) = 0.475$. The marginal density $p\big(\kappa_1 \big| \kappa_1 \leq \kappa_{\max}\big)$ with $\kappa_{\max} = 1$ (top right) is compared to a Gaussian approximation (5.71). The conditional marginal density (bottom right) of $\kappa_2 | \kappa_1 \leq \kappa_{\max}$ is distributed according to (5.70) (green). The derived Gaussian approximation (5.72) is shown in red.

Gaussians have been assumed. However, as it has been previously discussed, it stands to reason that correlations over time exists. Therefore, the problem will be first formulated in the general case, where the sequence of criticality values $\kappa_1, \ldots, \kappa_k$ is distributed according to a multivariate distribution $p\big(\kappa_1, \ldots, \kappa_k\big)$. The probability that an activation event has occurred until t_k is:

$$\mathrm{P}\big(t_\mathrm{B} \leq t_k\big) = 1 - \int_{-\infty}^{\kappa_{\max}} \cdots \int_{-\infty}^{\kappa_{\max}} p\big(\kappa_1, \ldots, \kappa_k\big)\ \mathrm{d}\kappa_1 \ldots\ \mathrm{d}\kappa_k\ . \tag{5.66}$$

In the following paragraphs, two specific models for $p\big(\kappa_1, \ldots, \kappa_k\big)$ will be detailed. First, a Gauss-Markov process is considered where the Gaussian distribution of κ_i is fully described with knowledge of the immediate predecessor value κ_{i-1}. Second, the special case of independent Gaussians will be studied. The validity of the derived models is assessed with Monte-Carlo simulations in the third paragraph.

Gauss-Markov process One frequent special case arises if the sequence $\kappa_1, \ldots, \kappa_k$ fulfils the Markov property, that is $p\big(\kappa_i \big| \kappa_{i-1}, \ldots, \kappa_1\big) = p\big(\kappa_i \big| \kappa_{i-1}\big)$, $i = 2, \ldots, k$. Hence, the joint density $p\big(\kappa_1, \ldots, \kappa_k\big)$ simplifies to

$$p\big(\kappa_1, \ldots, \kappa_k\big) = \left[\prod_{i=2}^{k} p\big(\kappa_i \big| \kappa_{i-1}\big)\right] \cdot p\big(\kappa_1\big)\ . \tag{5.67}$$

Moreover, the integration in (5.66) can be written as

$$\mathrm{P}\big(t_\mathrm{B} \leq t_k\big) = 1 - \left[\prod_{i=2}^{k} \int_{-\infty}^{\kappa_{\max}} p\big(\kappa_i \big| \kappa_{i-1} \leq \kappa_{\max}\big)\ \mathrm{d}\kappa_i\right] \cdot \int_{-\infty}^{\kappa_{\max}} p\big(\kappa_1\big)\ \mathrm{d}\kappa_1\ . \tag{5.68}$$

The densities $p\left(\kappa_i \middle| \kappa_{i-1} \leq \kappa_{\max}\right)$ are referred to as conditional marginals. Unfortunately, solving the iterated integration in (5.68) is still challenging. In the following, an approximate solution for the special case of Gaussian densities will be derived.

We assume Gaussian marginal densities, that is $p\left(\kappa_i\right) = \mathcal{N}\left(\kappa_i; \mu_\kappa\left(t_i\right), \sigma_\kappa^2\left(t_i\right)\right)$. The cross-covariance between two time steps $\operatorname{cov}\left(\kappa_i, \kappa_{i-1}\right)$ is denoted by $\rho_\kappa\left(t_i, t_{i-1}\right)$, $i = 2, \ldots, k$. According to (2.12), the conditional densities are Gaussians:

$$p\left(\kappa_i \middle| \kappa_{i-1}\right) = \mathcal{N}\left(\kappa_i; \mu_\kappa\left(t_i \middle| t_{i-1}\right), \sigma_\kappa^2\left(t_i \middle| t_{i-1}\right)\right) \ , \tag{5.69a}$$

where

$$\mu_\kappa\left(t_i \middle| t_{i-1}\right) = \mu_\kappa\left(t_i\right) + \rho_\kappa\left(t_i, t_{i-1}\right) \sigma_\kappa^{-2}\left(t_{i-1}\right) \left[\kappa_{i-1} - \mu_\kappa\left(t_{i-1}\right)\right] \ , \tag{5.69b}$$

$$\sigma_\kappa^2\left(t_i \middle| t_{i-1}\right) = \sigma_\kappa^2\left(t_i\right) - \rho_\kappa^2\left(t_i, t_{i-1}\right) \sigma_\kappa^{-2}\left(t_{i-1}\right) \ . \tag{5.69c}$$

However, the conditional marginal densities $p\left(\kappa_i \middle| \kappa_{i-1} \leq \kappa_{\max}\right)$ which are required to solve (5.68) are non-Gaussian. In the case of $i = 2$, one has the following probability density function [22] which is illustrated in Fig. 5.5:

$$\begin{aligned} p\left(\kappa_2 \middle| \kappa_1 \leq \kappa_{\max}\right) = & \left(\int_{-\infty}^{\kappa_{\max}} \mathcal{N}\left(\kappa_1; \mu_\kappa\left(t_1\right), \sigma_\kappa^2\left(t_1\right)\right) \, \mathrm{d}\kappa_1 \right)^{-1} \cdot \mathcal{N}\left(\kappa_2; \mu_\kappa\left(t_2\right), \sigma_\kappa^2\left(t_2\right)\right) \\ & \cdot \int_{-\infty}^{\kappa_{\max}} \mathcal{N}\left(\kappa_1; \mu_\kappa\left(t_1 \middle| t_2\right), \sigma_\kappa^2\left(t_1 \middle| t_2\right)\right) \, \mathrm{d}\kappa_1 \end{aligned} \tag{5.70}$$

In order to enable a recursive integration of (5.68), a Gaussian approximation of (5.70) will be derived in the following. Due to the high relevance to many problems of efficiently integrating a multivariate Gaussian density function, similar methods have been developed for example in [149, 203].

In principle, two ways can be followed to obtain a Gaussian approximation of the conditional marginal density $p\left(\kappa_i \middle| \kappa_{i-1} \leq \kappa_{\max}\right)$. On the one hand, the first two moments $\mu_\kappa\left(t_i \middle| \kappa_{i-1} \leq \kappa_{\max}\right)$ and $\sigma_\kappa^2\left(t_i \middle| \kappa_{i-1} \leq \kappa_{\max}\right)$ could be explicitly calculated from (5.70), which is pursued for example in [22, 132].

A different approach is to find a Gaussian approximation of the truncated density $p\left(\kappa_{i-1} \middle| \kappa_{i-1} \leq \kappa_{\max}\right)$ first. Afterwards, the conditional marginal density $p\left(\kappa_i \middle| \kappa_{i-1} \leq \kappa_{\max}\right)$ is calculated from the conditional density $p\left(\kappa_i \middle| \kappa_{i-1}\right)$ from (5.69) and the truncated density using the law of total probability. As will be shown in the following, this approach leads to the same result as the first method.

First, the truncated density is approximated by a Gaussian [22]:

$$p\left(\kappa_{i-1} \middle| \kappa_{i-1} \leq \kappa_{\max}\right) \approx \mathcal{N}\left(\kappa_{i-1}; \mu_\kappa\left(t_{i-1} \middle| \kappa_{i-1} \leq \kappa_{\max}\right), \sigma_\kappa^2\left(t_{i-1} \middle| \kappa_{i-1} \leq \kappa_{\max}\right)\right) \ , \tag{5.71a}$$

where

$$\mu_\kappa\left(t_{i-1} \middle| \kappa_{i-1} \leq \kappa_{\max}\right) = \mu_\kappa\left(t_{i-1}\right) - \sigma_\kappa\left(t_{i-1}\right) \lambda_\kappa\left(t_{i-1}\right) \ , \tag{5.71b}$$

$$\sigma_\kappa^2\left(t_{i-1} \middle| \kappa_{i-1} \leq \kappa_{\max}\right) = \sigma_\kappa^2\left(t_{i-1}\right) \left[1 - \lambda_\kappa\left(t_{i-1}\right) \left(\beta_\kappa\left(t_{i-1}\right) + \lambda_\kappa\left(t_{i-1}\right)\right)\right] \ , \tag{5.71c}$$

$$\lambda_\kappa\left(t_{i-1}\right) := \frac{\mathcal{N}(\beta_\kappa\left(t_{i-1}\right);0,1)}{\int_{-\infty}^{\beta_\kappa\left(t_{i-1}\right)}\mathcal{N}(\kappa;0,1)\ \mathrm{d}\kappa}\ , \qquad \beta_\kappa\left(t_{i-1}\right) := \frac{\kappa_{\max}-\mu_\kappa\left(t_{i-1}\right)}{\sigma_\kappa\left(t_{i-1}\right)}\ . \tag{5.71d}$$

Effectively, the mean of κ_{i-1} after truncation is shifted away from the upper bound at $\kappa_{\max}$ and the variance decreases.

Second, a Gaussian approximation of the conditional marginal density is calculated from the conditional density $p\left(\kappa_i\middle|\kappa_{i-1}\right)$ and the approximated truncated density (5.71). Both distributions are Gaussians and applying the law of total probability yields according to (2.13):

$$p\left(\kappa_i\middle|\kappa_{i-1}\le\kappa_{\max}\right) \approx \mathcal{N}\left(\kappa_i;\mu_\kappa\left(t_i\left|\kappa_{i-1}\le\kappa_{\max}\right.\right),\sigma_\kappa^2\left(t_i\left|\kappa_{i-1}\le\kappa_{\max}\right.\right)\right)\ , \tag{5.72a}$$

where

$$\begin{aligned}
\mu_\kappa\left(t_i\left|\kappa_{i-1}\le\kappa_{\max}\right.\right) &= \mu_\kappa\left(t_i\right)+\frac{\rho_\kappa\left(t_i,t_{i-1}\right)}{\sigma_\kappa^2\left(t_{i-1}\right)}\cdot\left[\mu_\kappa\left(t_{i-1}\left|\kappa_{i-1}\le\kappa_{\max}\right.\right)-\mu_\kappa\left(t_{i-1}\right)\right] \\
&\overset{(5.71b)}{=} \mu_\kappa\left(t_i\right)-\frac{\rho_\kappa\left(t_i,t_{i-1}\right)}{\sigma_\kappa}\lambda_\kappa\left(t_{i-1}\right)\ , \qquad (5.72b)\\
\sigma_\kappa^2\left(t_i\left|\kappa_{i-1}\le\kappa_{\max}\right.\right) &= \sigma_\kappa^2\left(t_i\right)+\frac{\rho_\kappa^2\left(t_i,t_{i-1}\right)}{\sigma_\kappa^4\left(t_{i-1}\right)}\cdot\left[\sigma_\kappa^2\left(t_{i-1}\left|\kappa_{i-1}\le\kappa_{\max}\right.\right)-\sigma_\kappa^2\left(t_{i-1}\right)\right] \\
&\overset{(5.71c)}{=} \sigma_\kappa^2\left(t_i\right)-\frac{\rho_\kappa^2\left(t_i,t_{i-1}\right)}{\sigma_\kappa^2\left(t_{i-1}\right)}\cdot\left[\lambda_\kappa\left(t_{i-1}\right)\left(\beta_\kappa\left(t_{i-1}\right)+\lambda_\kappa\left(t_{i-1}\right)\right)\right]\ . \qquad (5.72c)
\end{aligned}$$

This final result is the same as the one derived in [22] where the true non-Gaussian density (5.70) is approximated by its first two moments.

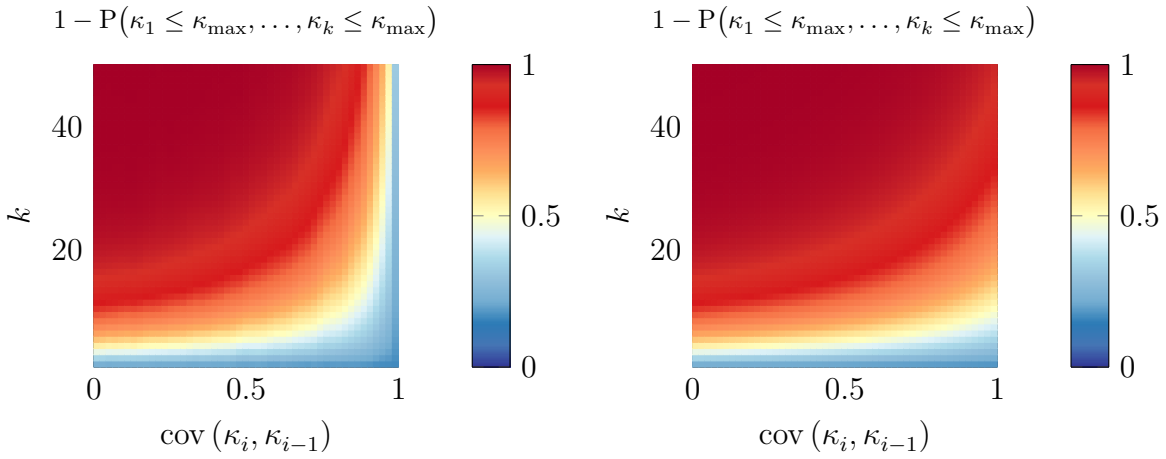
Based on the results $\mu_\kappa\left(t_{i-1}\left|\kappa_{i-2}\le\kappa_{\max}\right.\right)$ and $\sigma_\kappa^2\left(t_{i-1}\left|\kappa_{i-2}\le\kappa_{\max}\right.\right)$ from each previous iteration, one can recursively calculate the approximation for $i=2,\dots,k$. Finally, (5.68) becomes a product of integrals of one-dimensional Gaussians:

$$\begin{aligned}
\mathrm{P}\left(t_{\mathrm{B}}\le t_k\right) = 1-\prod_{i=2}^{k}\int_{-\infty}^{\kappa_{\max}}\mathcal{N}\left(\kappa_i;\mu_\kappa\left(t_i\left|\kappa_{i-1}\le\kappa_{\max}\right.\right),\sigma_\kappa^2\left(t_i\left|\kappa_{i-1}\le\kappa_{\max}\right.\right)\right)\ \mathrm{d}\kappa_i \\
\cdot\int_{-\infty}^{\kappa_{\max}}\mathcal{N}\left(\kappa_1;\mu_\kappa\left(t_1\right),\sigma_\kappa^2\left(t_1\right)\right)\ \mathrm{d}\kappa_1\ .
\end{aligned} \tag{5.73}$$

Independent Gaussians Solving (5.66) is drastically simplified if $\kappa_1,\dots,\kappa_k$ is a sequence of independent Gaussian random variables. Then, the joint density $p\left(\kappa_1,\dots,\kappa_k\right)$ can be factorised and the integration performed individually:

$$\mathrm{P}\left(t_{\mathrm{B}}\le t_k\right) = 1-\prod_{i=1}^{k}\int_{-\infty}^{\kappa_{\max}}\mathcal{N}\left(\kappa_i;\mu_\kappa\left(t_i\right),\sigma_\kappa^2\left(t_i\right)\right)\ \mathrm{d}\kappa_i\ . \tag{5.74}$$

This result has been obtained in [83] by first modelling the activation logic as a two state Markov chain and applying the Chapman-Kolmogorov equation. Note that the approximate solution (5.73) reduces to (5.74) in the uncorrelated case. This is because the conditional marginal density (5.72) equals the unconditional one if $\rho_\kappa\left(t_i,t_{i-1}\right)=0$.



(a) Monte-Carlo integration of (5.66). **(b)** Closed-form approximation (5.73).

Figure 5.6 Probability of exceeding a threshold $\kappa_{\max} = 1$ until the kth element of a random sequence defined by a Gauss-Markov process. The process is defined by $\kappa_i \sim \mathcal{N}(0, 1)$ and varying cross-covariance $\mathrm{cov}\,(\kappa_i, \kappa_{i-1}) \in [0, 1]\ \forall i$. Comparing the numerical Monte-Carlo estimate (a) to the approximate solution (5.73) in (b) reveals deviations for high values of the correlation, that is $\mathrm{cov}\,(\kappa_i, \kappa_{i-1}) \to 1$. Nevertheless, the importance of considering the correlation is indicated by the non-linear contours.

Monte-Carlo simulations The purpose of the numerical analysis is to compare the probability of an activation given by the integral (5.66) to the approximation (5.73). This is done for a Gauss-Markov process with varying correlation coefficient. Thereby, insight on the general limitations of this approximation is obtained.

Sequences $\kappa_1, \ldots, \kappa_k$ of a random variable from a Gauss-Markov process up to $k = 50$ are generated. The individual elements are identically distributed Gaussians $\kappa_i \sim \mathcal{N}(0, 1)$. Simulations are performed for different values $\mathrm{cov}\,(\kappa_i, \kappa_{i-1}) \in [0, 1]$ of the cross-covariance.[17] A threshold of $\kappa_{\max} = 1$ defines an activation event. For each of $N_{\mathrm{sim}} = 1 \times 10^4$ independent sequences it is evaluated whether $\kappa_{\max}$ has been exceeded at least once until the kth element of the sequence. Thus, the fraction of all sequences, for which this is true, yields a Monte-Carlo estimate of the probability $\mathrm{P}(t_{\mathrm{B}} \leq t_k)$.

The estimated probability is visualised over the cross-covariance in Fig. 5.6 and compared to the analytical model (5.73). If $\mathrm{cov}\,(\kappa_i, \kappa_{i-1}) = 0$, one has the case of independent Gaussians and the approximate model reduces to the exact solution (5.74). For a fixed sequence length k, the highest probability of an activation is obtained in this case but lower values result as the cross-covariance increases. This stresses that neglecting the correlation leads to inaccurate results.

For high correlations $\mathrm{cov}\,(\kappa_i, \kappa_{i-1}) \to 1$, the approximate model fails to produce correct estimates. Due to the strong coupling between subsequent values, the first element κ_1

[17] Since $\mathrm{var}\,(\kappa_i) = 1$, the correlation coefficient $\mathrm{cor}\,(\kappa_i, \kappa_{i-1}) = \frac{\mathrm{cov}(\kappa_i, \kappa_{i-1})}{\mathrm{var}(\kappa_i)}$ thus varies within the possible spectrum from independence to total positive correlation.

alone determines $\kappa_2, \ldots, \kappa_k$. Thus, the overall probability of exceeding $\kappa_{\max}$ once does not depend on the length of the sequence.

5.2.4.4 Probability of a collision

Previously, the distribution of a criticality measure has been studied under the assumption that a collision will definitely occur. In order to find the overall unconditional distribution, the probability of a collision needs to be considered.

The initial formulation of criticality in (5.44) is very general and includes the possibility that no further driver action at all is needed to avoid a collision within the prediction horizon. This might be the case if a preceding vehicle is driving on a different lane or with a higher velocity. In order to derive specific risk metrics, the problem is transformed to finding the necessary driver inputs which lead to the last uncritical state $\boldsymbol{\gamma}$ prior to a collision. The transformation (5.46) implicitly assumes that a collision will occur otherwise. Therefore, certain restrictions on the state have to be imposed in order to obtain sensible criticality values $\kappa\left(\mathbf{x}_k\right)$. For example, the TTC (5.48) is defined only for negative relative velocities $v_{x,k} < 0$ and possesses a singularity for $v_{x,k} = 0$. Formally, a criticality measure $\kappa\left(\mathbf{x}_k\right)$ is thus defined on a subset $\mathbf{x}_k \in \mathcal{X}_\mathrm{C}$ of the entire state-space.

Concerning the statistical model from Sec. 5.2.4.1, the previously derived Gaussian $\mathcal{N}\left(\kappa\left(\mathbf{x}_k\right); \mu_\kappa\left(t_k\right), \sigma_\kappa^2\left(t_k\right)\right)$ therefore defines a conditional distribution $p\left(\kappa\left(\mathbf{x}_k\right) \middle| \mathbf{x}_k \in \mathcal{X}_\mathrm{C}\right)$. In order to obtain the unconditional distribution $p\left(\kappa\left(\mathbf{x}_k\right)\right)$ it is necessary to know the probability $\mathrm{P}\left(\mathbf{x}_k \in \mathcal{X}_\mathrm{C}\right)$. However, since the future trajectory is assumed uncertain in this work, a straightforward definition of $\mathcal{X}_\mathrm{C}$ is not possible. Instead, the fundamental problem of determining the probability of a collision between two objects which are on uncertain trajectories $\mathbf{x}^\mathrm{e}\left(t\right)$ and $\mathbf{x}^\mathrm{o}\left(t\right)$ has to be solved.

Given the joint probability density $p_{(t)}\left(\mathbf{x}^\mathrm{e}, \mathbf{x}^\mathrm{o}\right) := p\left(\mathbf{x}^\mathrm{e}\left(t\right), \mathbf{x}^\mathrm{o}\left(t\right)\right)$ and a collision indicator function $I_\mathrm{C}\left(\mathbf{x}^\mathrm{e}, \mathbf{x}^\mathrm{o}\right)$ from (5.43), the instantaneous probability of a collision event $C_{(t)}$ at a specific point in time t can be calculated as follows [50]:

$$\mathrm{P}\left(C_{(t)}\right) = \int_{\mathbf{x}^\mathrm{o}} \int_{\mathbf{x}^\mathrm{e}} I_\mathrm{C}\left(\mathbf{x}^\mathrm{e}, \mathbf{x}^\mathrm{o}\right) p_{(t)}\left(\mathbf{x}^\mathrm{e}, \mathbf{x}^\mathrm{o}\right) \, \mathrm{d}\mathbf{x}^\mathrm{e} \, \mathrm{d}\mathbf{x}^\mathrm{o} \, . \tag{5.75}$$

Depending on the involved probability density and geometries, evaluating this integral may become intractable and must be approached by Monte-Carlo techniques. Furthermore, $\mathrm{P}\left(C_{(t)}\right)$ defines the probability of a collision at a single time instant t conditional that the previous trajectory has been collision-free. In order to obtain the overall collision probability $\mathrm{P}\left(C_{\left(t_k, t_k+\bar{T}\right)}\right)$ within a time interval $t \in \left[t_k, t_k + \bar{T}\right]$, the integration has to be extended over the time domain.

A number of previous works have investigated this problem, but most of them rely on numerical Monte-Carlo methods. To alleviate the computational burden, various re-parametrisations or assumptions on the general problem formulation have been reported and a brief overview will be given in the following. As a novel contribution, an approximation of an approach described in [145] will be proposed. This simplifies the problem to an integration over a one-dimensional Gaussian density function.

Most works introduce a discrete-time formulation of the problem. Then, a discrete time Bayes filter[18], for example the unscented [34] or extended Kalman filter [90], can be used to calculate the state predictions $p\left(\mathbf{x}_k^{\mathrm{e}}, \mathbf{x}_k^{\mathrm{o}}\right)$ at future time steps. Non-iterative predictions in continuous time, such as the linearisation approach from Sec. 4.2, are employed in [141, 145]. Another work uses stochastic reachable sets [4].

With the state prediction densities, the integration over the state space can be performed by Monte-Carlo techniques [173] or grid-based calculations [4, 148]. Alternatively, a re-parametrisation can be introduced first in order to simplify the calculation. For example, collisions can be defined on a two-dimensional vector space [90], as a one-dimensional distance function [18] or in two-dimensional relative coordinates [28].

The difficulty of this problem is due to the dependence between the states over time. At each time instant, the distribution of the current state is conditional that no collision has occurred before. This can be identified as the same notorious integration as in (5.66) but with the added complexity of multivariate density functions. A technique based on conditional marginals, similar to the one discussed in the previous section, is proposed in [150] to efficiently calculate the probability.

Suitable for potential rear-end collisions in longitudinal traffic, [145] develops a completely analytical formulation. Here, the motion is separated in longitudinal and lateral direction. Moreover, simple rectangular shapes and zero heading angles are assumed. A similar setting is assumed in [141] but a further projection on a constant non-zero heading angle is taken into account. By considering the lateral and longitudinal motion separately, the calculation is structured into two steps:

1. Firstly, the probability density $p_{t_{\mathrm{tc}}}\left(t_{\mathrm{tc}}\right)$ of the point in time t_{tc} where both vehicles share the same longitudinal position x has to be obtained.

2. Secondly, the probability density $p_y\left(y\left(\tau\right)\right)$ of the lateral distance y at a time instant τ is integrated over a critical corridor $|y| \leq y_{\mathrm{coll}}$ conditional on the time t_{tc}. The width y_{coll} is defined by the vehicles' shapes.

In the case of independent motion in x and y, this reads [145]:

$$\mathrm{P}\left(C_{\left(t_k, t_k+\bar{T}\right)}\right) = \int_{t_k}^{t_k+\bar{T}} \int_{-y_{\mathrm{coll}}}^{y_{\mathrm{coll}}} p_y\left(y\left(\tau\right)\middle|\tau\right) p_{t_{\mathrm{tc}}(\mathbf{x}_k)}\left(\tau\right) \ \mathrm{d}y \ \mathrm{d}\tau \ . \tag{5.76}$$

In the following, a further simplification of this bivariate non-Gaussian integration to a one-dimensional Gaussian is proposed. To this end, it is assumed that the uncertainty in $t_{\mathrm{tc}}\left(\mathbf{x}_k\right)$ can be neglected. This assumption simplifies (5.76) to:

$$\mathrm{P}\left(C_{\left(t_k, t_k+\bar{T}\right)}\right) = \int_{-y_{\mathrm{coll}}}^{y_{\mathrm{coll}}} p_y\left(y\left(t_{\mathrm{tc}}\left(\hat{\mathbf{x}}_k\right)\right)\right) \ \mathrm{d}y \ . \tag{5.77}$$

The simplification to a one-dimensional Gaussian, whose integral is available for example using lookup-tables, is certainly beneficial for real-time applications. The goodness of this approximation will be further evaluated with Monte-Carlo simulations in Sec. 5.2.5.3.

[18] However, since the objective is to calculate a prediction, the measurement step is left out and the algorithm consists of an iterated calculation of the prediction step (4.4).

Table 5.2 Simulation parameter values for uncertainty propagation in the TTC.

Variable	Value
State estimation mean	$\hat{\mathbf{x}}_k = \begin{bmatrix} \hat{x}_k & -10\,\mathrm{m/s} \end{bmatrix}^{\mathrm{T}}$, $\hat{x}_k \in \{30\,\mathrm{m}, 20\,\mathrm{m}, 10\,\mathrm{m}\}$
State estimation covariance	$\boldsymbol{\Sigma}_{\mathbf{x}}(t_k) = \mathrm{diag}\left(0.25\,\mathrm{m}^2, 0.0625\,\mathrm{m}^2/\mathrm{s}^2\right)$
Process noise	$S_x = 2 \cdot 0.375\,\mathrm{m}^2/\mathrm{s}^4\mathrm{s}^{-1}$

5.2.5 Numerical examples

The following three sections detail simulation results which are used to illustrate and validate the analytical models from Sec. 5.2.4. Firstly, the distribution of criticality measures under state estimation and prediction uncertainty is considered in Sec. 5.2.5.1. Subsequently, the time of activation is analysed in Sec. 5.2.5.2. Thirdly, the proposed model for the collision probability is assessed in 5.2.5.3.

5.2.5.1 Uncertainty propagation in criticality measures

The distribution of the TTC criticality measure from example 5.3 is analysed in the following due to its intuitive interpretation.

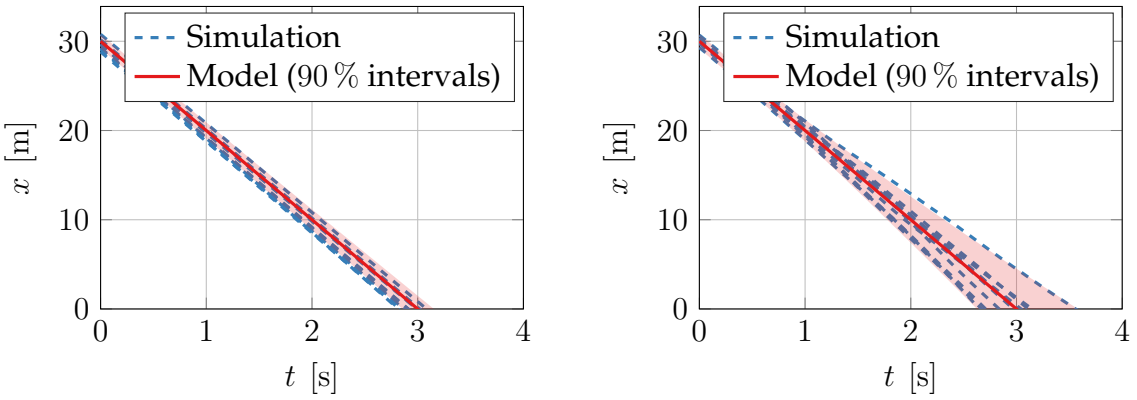
The simulation setup comprises simulated trajectories which are defined in relative coordinates and state estimates affected by Gaussian noise. TTC values are calculated from these noisy estimates. The scenario consists of a one-dimensional motion with a relative velocity of $-36\,\mathrm{km/h}$. Simulations are performed for three different mean distances $\hat{x}_k \in \{30\,\mathrm{m}, 20\,\mathrm{m}, 10\,\mathrm{m}\}$. Examples of the resulting trajectories are shown in Fig. 5.7 based on the parameter values as given in Tab. 5.2.

For each simulated trajectory, the true value of the TTC is given by the time of crossing the x-axis in Fig. 5.7. A Gaussian model of the distribution of the TTC measure has been derived in example 5.6. To first order, the expected value is $t_{\mathrm{tc}}(\hat{\mathbf{x}}_k)$. The variance $\mathrm{var}(t_{\mathrm{tc}}(\mathbf{x}_k))$ on the other hand follows from (5.58).

Comparing the deterministic (Fig. 5.7(a)) and stochastic (Fig. 5.7(b)) trajectories reveals that the TTC is more widespread in the second case. This difference can be recognised in the histograms in Fig. 5.8 as well, where the simulated values that result for different initial distances are shown. As is expected, an increased variation due to the uncertainty of the trajectory is caused at higher distances, that is a longer prediction horizon. Finally, the analytical models agree well with the numerical results.

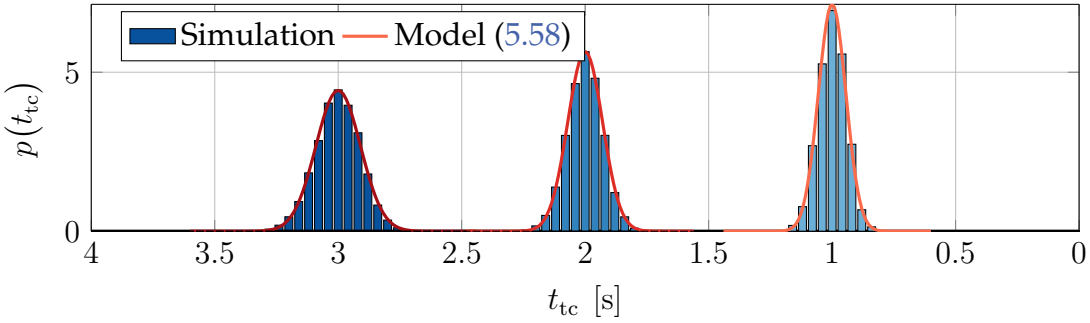
5.2.5.2 Uncertainty in decision timing

Having analysed the distribution of the time-to-collision at single time instants, it is now studied how the probability of exceeding an activation threshold evolves over time, as it is discussed in Sec. 5.2.4.3. Thus, the entire derivation of the TTC distribution, including its time correlation and the approximate solution of the integration is evaluated.

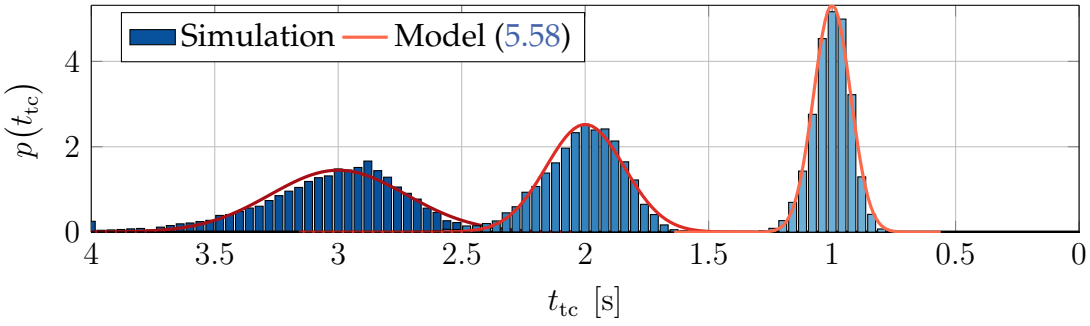


(a) Without process noise. **(b)** With process noise.

Figure 5.7 Distance x over time for a trajectory with a mean initial value $\hat{x}_k = 30\,\mathrm{m}$ and a relative velocity of $\hat{v}_{x,k} = -36\,\mathrm{km/h}$. Trajectories are generated with randomly sampled initial values and with (a) a deterministic or (b) uncertain motion model. The analytical propagation from example 4.4 is visualised by dashed and dash-dotted lines for the mean trajectory and 90% quantiles, respectively. The time of crossing the x-axis is the time-to-collision which becomes more widespread in the case of process noise.



(a) Without process noise.



(b) With process noise.

Figure 5.8 Time-to-collision for trajectories with three different mean initial distances $\hat{x}_k \in \{30\,\mathrm{m}, 20\,\mathrm{m}, 10\,\mathrm{m}\}$ and relative velocity $\hat{v}_{x,k} = -10\,\mathrm{m/s}$. For each of the three initial distances, $N_{\mathrm{sim}} = 1 \times 10^3$ trajectories are simulated without (a) and with (b) process noise. The times of crossing the x-axis as visualised in Fig. 5.7 are depicted as histograms. For comparison, the analytical model derived in example 5.6 is shown and a good correspondence is obtained. Slight deviations of the simulated distributions from a Gaussian are caused by the non-linear mapping to the criticality. Comparing (a) and (b) reveals that the influence of uncertain trajectory predictions increases with the prediction horizon, that is the t_{tc}.

Table 5.3 Simulation parameter values for evaluating the decision timing based on the TTC. In order to minimise the influence of time discretisation on the activation timing, the sampling time is intentionally chosen smaller than in other numerical examples.

Variable	Value
Initial state mean	$\hat{\mathbf{x}}_1 = \begin{bmatrix} 20\,\mathrm{m} & -10\,\mathrm{m/s} \end{bmatrix}^{\mathrm{T}}$
State estimation covariance	$\boldsymbol{\Sigma}_{\mathbf{x}} := \boldsymbol{\Sigma}_{\mathbf{x}}(t_i) = \mathrm{diag}\left(0.25\,\mathrm{m}^2, 0.0625\,\mathrm{m}^2/\mathrm{s}^2\right)\ \forall i$
State error correlation	$\mathrm{cov}(\mathbf{x}_i, \mathbf{x}_{i-1}) = 0.3 \cdot \boldsymbol{\Sigma}_{\mathbf{x}}\ \forall i$
Sampling time	$T_{\mathrm{S}} = 0.006\,75\,\mathrm{s}$
Activation threshold	$t_{\mathrm{tc,min}} = 0.8\,\mathrm{s}$

For the example of the TTC metric, a brake activation is naturally defined when falling below a threshold $t_{\mathrm{tc}}(\mathbf{x}_k) < t_{\mathrm{tc,min}}$. Here, a value of $t_{\mathrm{tc,min}} = 0.8\,\mathrm{s}$ is chosen.[19]

The true trajectories are assumed as deterministic now and therefore no process noise is included. Otherwise, the ideal time of intervention would become a random variable itself. A mean initial distance of $\hat{x}_1 = 20\,\mathrm{m}$ and a relative velocity $\hat{v}_{x,1} = -10\,\mathrm{m/s}$ are chosen. Ideally, the criticality threshold $t_{\mathrm{tc,min}} = 0.8\,\mathrm{s}$ is thus reached at $t_{\mathrm{B,ideal}} = 1.2\,\mathrm{s}$. All parameters which define the simulation are given in Tab. 5.3. Two cases are investigated: First, the state estimation errors are sampled independently with $\mathrm{cov}(\mathbf{x}_i) = \boldsymbol{\Sigma}_{\mathbf{x}}$ and $\mathrm{cov}(\mathbf{x}_i, \mathbf{x}_{i-1}) = \mathbf{0}\ \forall i$. In the second case, correlated sequences of the estimation errors are sampled from a Gauss-Markov process with $\mathrm{cov}(\mathbf{x}_i, \mathbf{x}_{i-1}) = 0.3 \cdot \boldsymbol{\Sigma}_{\mathbf{x}}\ \forall i$.

For the first case, Fig. 5.9(a) shows the variance and cross-covariance over time and compares the simulated values to the analytical models. Criticality estimates are calculated from the noisy states and the resulting time of activation is shown in Fig. 5.10(a). Although the noise has zero mean, the distribution of t_{B} is clearly biased towards early activations, an effect that was previously noted in [83] as well. In the absence of correlations, (5.74) is an exact solution of the integration and agrees well with the simulation results.

Secondly, the simulation is conducted for sequences with correlated estimation errors. The propagation of variance and cross-covariance in the TTC is visualised in Fig. 5.9(b) and compared to the analytical models. Both quantities are again well described by the theoretical result. Since the covariance is chosen identical to the first simulation, the model (5.74) of independent Gaussian errors remains the same. However, the simulated values of the activation time shown in Fig. 5.10(b) reveal a wider distribution. This effect can therefore be attributed to the correlation of the estimation errors. Concerning the previous analysis of the approximate model (5.73) in Fig. 5.6, the correlation coefficient

[19] Thus, the density after truncation from below $p\left(t_{\mathrm{tc},i-1} \middle| t_{\mathrm{tc},i-1} \geq t_{\mathrm{tc,min}}\right)$ has to be approximated. The result is similar to the density bounded from above in (5.71). β_κ in (5.71d) remains the same but with the lower bound κ_{min} inserted. Only a slightly different definition of $\lambda_\kappa := \left(\mathcal{N}(\beta_\kappa; 0, 1)\right)\left(1 - \int_{-\infty}^{\beta_\kappa} \mathcal{N}(\kappa; 0, 1)\ \mathrm{d}\kappa\right)^{-1}$ is needed.

lies within a region where good approximations are expected. It is thus not surprising that the simulated values are matched well by the model.

5.2.5.3 Probability of a collision

As has been highlighted in Sec. 5.2.4.4, calculating the probability of a future collision between extended objects in general dynamic scenes is almost impossible without Monte-Carlo methods. In the special case of longitudinal scenarios and simplified geometries, the problem can be tailored to integrating over a univariate Gaussian density function (5.77). The accuracy of this appealingly simple approach is evaluated in the following.

The analytical model (5.77) relies on a representation of the situation in the moving Cartesian coordinate system relative to the ego-vehicle. A collision is then defined in (5.76) by crossing the y-axis at a lateral position $|y| \leq y_{\mathrm{coll}}$. Since the tolerable margin in general depends not only on the dimensions but also on the orientation angle of the object, y_{coll} has to be defined with assumptions on this angle.

Here, rectangular shapes with lengths l_{ego} and l_{obj} as well as widths w_{ego} and w_{obj} for ego-vehicle and object, respectively, are assumed. A lower value of the critical corridor's width is obtained for exactly parallel trajectories at the time of passing as:

$$y_{\mathrm{coll,underapprox}} = \frac{1}{2}\left(w_{\mathrm{ego}} + \min\left(w_{\mathrm{obj}}, l_{\mathrm{obj}}\right)\right) \ . \tag{5.78}$$

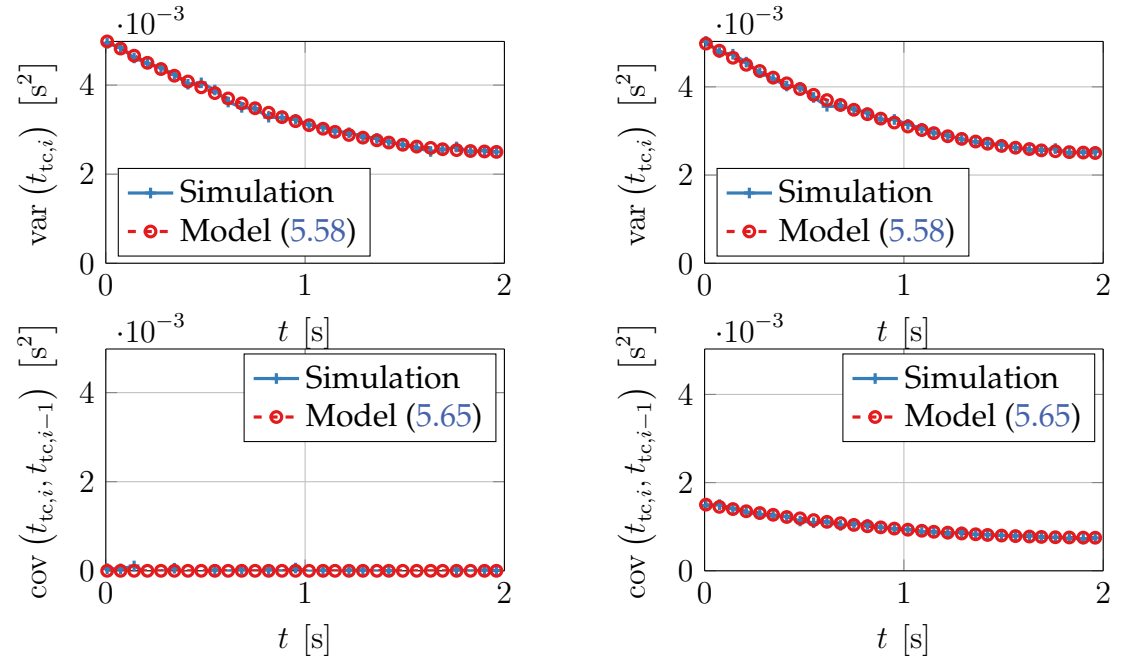
On the other hand, if the object's direction of motion deviates from a parallel course towards the ego-vehicle, a sensible value is given by the sum of the diagonals:

$$y_{\mathrm{coll,overapprox}} = \frac{1}{2}\left(\sqrt{l_{\mathrm{ego}}^2 + w_{\mathrm{ego}}^2} + \sqrt{l_{\mathrm{obj}}^2 + w_{\mathrm{obj}}^2}\right) \ . \tag{5.79}$$

The chosen simulation setup is similar to the one from Sec. 4.2.8.1: While the ego-vehicle moves on a straight trajectory according to the CTRA model, an object crosses into the driving corridor with a constant velocity. The motion of both traffic participants is driven by process noise and only an uncertain estimate of the object's initial state is known. Details on the selected parameter values are given in Tab. 5.4 and exemplary trajectories are visualised in Fig. 5.11(a).

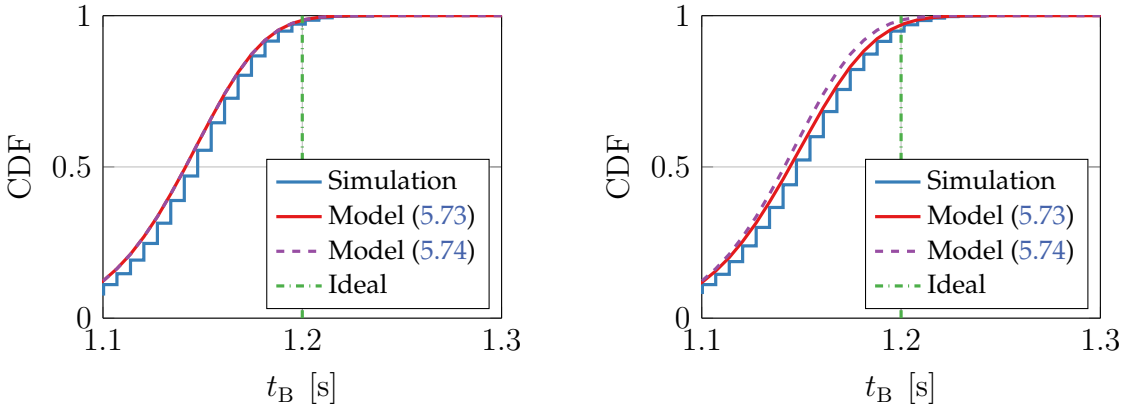
Monte-Carlo simulations with $N_{\mathrm{sim}} = 1 \times 10^3$ independent trajectories per initial state yield numerical estimates of the collision probabilities. These reference values are shown in Fig. 5.12(a) over the object's initial position (x_0, y_0). As is expected, the highest risk results when starting immediately in front of the ego-vehicle. Since the object moves perpendicular to the vehicle's trajectory, the contour lines are tilted. Fig. 5.11(b) shows the results for one initial lateral offset $y_0 = 0\,\mathrm{m}$. Since the trajectories cross at an angle of approximately 90°, the simulated values are closer to the model prediction for the underapproximation (5.78) than the conservative approximation (5.79).

Overall, a reasonable prediction of the collision probability is obtained with the analytical approach. Limitations of the accuracy are caused by an unknown angle and the assumption of mostly longitudinal motion.



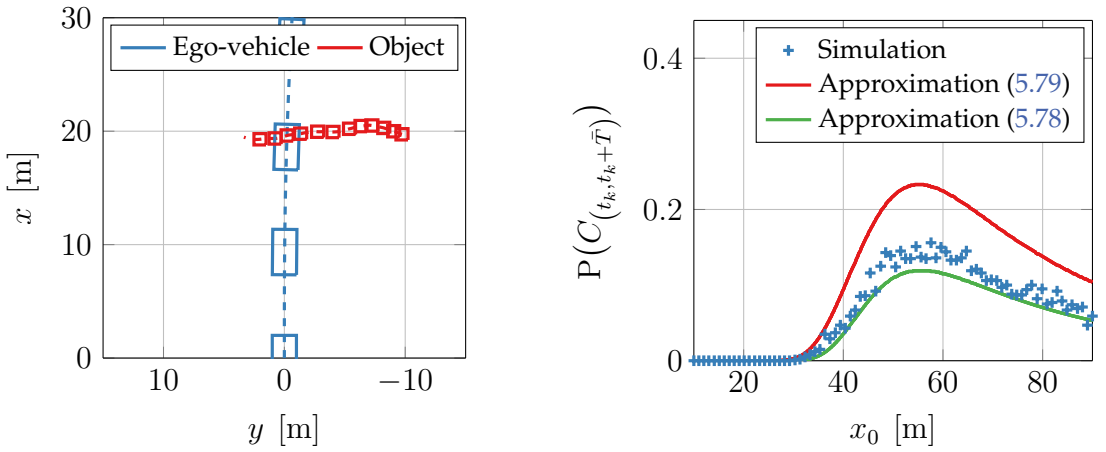
(a) Uncorrelated state estimates. **(b)** Correlated state estimates.

Figure 5.9 Variance var $(t_{\mathrm{tc},i})$ (upper row) and cross-covariance cov $(t_{\mathrm{tc},i}, t_{\mathrm{tc},i-1})$ (lower row) over time. The propagation of correlated state estimation errors to the criticality is derived according to Sec. 5.2.4.2.



(a) Uncorrelated state estimates. **(b)** Correlated state estimates.

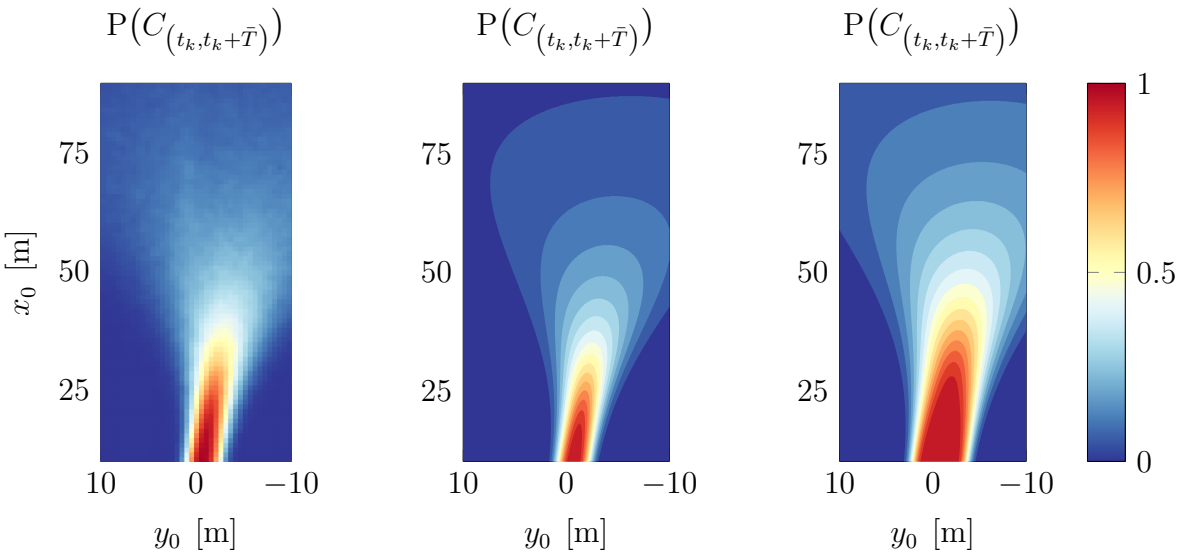
Figure 5.10 The time t_{B} at which the TTC criticality measure exceeds the threshold $t_{\mathrm{tc,min}} = 0.8\,\mathrm{s}$ is shifted to earlier activations due to noisy state estimates. For uncorrelated state estimation errors (a), the probability can be calculated exactly according to (5.74). If errors are correlated over time (b), calculating the exact probability is intractable but the recursive approximation (5.73) yields an estimate thereof.



(a) Exemplary situation

(b) Collision probability for varying initial distance x_0 and $y_0 = 0\,\mathrm{m}$.

Figure 5.11 Monte-Carlo simulation results of the collision probability. An exemplary realisation of the simulated trajectories is shown in (a). For a varying initial distance x_0, the estimated collision probability is shown in (b). The numerical results are compared to the analytical model (5.77) with the underapproximation (5.78) and the conservative approximation (5.79) for the parametrisation of the critical corridor width y_{coll}.



(a) Simulation.

(b) Model (5.78).

(c) Model (5.79).

Figure 5.12 Probability of a collision for different mean initial object positions (x_0, y_0). Monte-Carlo simulation results in (a) serve as reference for the developed analytical model (5.77). Model predictions are calculated for two variants of the critical corridor width, the lower bound (5.78) in (b) and the conservative approximation (5.79) in (c).

Table 5.4 Simulation parameter values for evaluating the collision probability.

	Variable	Value
Ego-vehicle	Initial state mean	$\hat{\mathbf{x}}^{\mathrm{e}}(t_k) = \begin{bmatrix} 0\,\mathrm{m} & 0\,\mathrm{m} & 13.89\,\mathrm{m/s} & 0^\circ & 0\,\mathrm{m/s^2} & 0\,\mathrm{rad/s} \end{bmatrix}^{\mathrm{T}}$
	Initial state covariance	$\boldsymbol{\Sigma}^{\mathrm{e}}_{\mathbf{x}}(t_k) = \mathbf{0}_{6\times 6}$
	Process noise	$S^{\mathrm{e}}_a = 0.224\,\mathrm{m^2/s^6 s^{-1}}$, $S^{\mathrm{e}}_\omega = 0.0038\,\mathrm{rad^2/s^4 s^{-1}}$
	Vehicle dimensions	$w_{\mathrm{ego}} = 2\,\mathrm{m}$, $l_{\mathrm{ego}} = 4\,\mathrm{m}$
Object	Initial state mean	$\hat{\mathbf{x}}^{\mathrm{o}}(t_k) = \begin{bmatrix} x_0 & y_0 & 0\,\mathrm{m/s} & 1\,\mathrm{m/s} \end{bmatrix}^{\mathrm{T}}$, $x_0 \in [10\,\mathrm{m}, 80\,\mathrm{m}]$, $y_0 \in [-10\,\mathrm{m}, 10\,\mathrm{m}]$
	Initial state covariance	$\boldsymbol{\Sigma}^{\mathrm{o}}_{\mathbf{x}}(t_k) = \mathrm{diag}\left(0.25\,\mathrm{m^2}, 0.25\,\mathrm{m^2}, 0.0625\,\mathrm{m^2/s^2}, 0.0625\,\mathrm{m^2/s^2}\right)$
	Process noise	$S_x = 0.25\,\mathrm{m^2/s^4 s^{-1}}$, $S_y = 0.25\,\mathrm{m^2/s^4 s^{-1}}$
	Object dimensions	$w_{\mathrm{obj}} = l_{\mathrm{obj}} = 1\,\mathrm{m}$
	Sampling time	$T_{\mathrm{S}} = 0.0675\,\mathrm{s}$
	Prediction horizon	$\bar{T} = 8\,\mathrm{s}$

5.2.6 Summary

Criticality measures are examples of algorithms for situation interpretation with a widespread application in emergency brake systems. In order to develop robust systems, previous works have investigated the properties of these algorithms under inevitable measurement uncertainties. Both numerical and analytical approaches have been used to this end.

However, a crucial part of a criticality assessment is given by implicit prediction models. Since possible uncertainties in such predictions have not been addressed in previous analyses, this section extends the state of the art with a method for the analytical propagation of measurement and prediction errors.

In order to obtain a method that is applicable beyond a particular algorithm, several criticality measures have been first reviewed and formulated in a generic framework. Subsequently, Gaussian error propagation in closed form has been developed based on this framework and exemplified for two specific algorithms.

A second contribution concerns the time at which a criticality measure exceeds an activation threshold. Previous analyses have modelled this activation timing under the assumption that criticality estimates are independently distributed over time. The problem has been extended to the more general case of a Markovian time-series where the errors are correlated. Since the exact probability is practically intractable to calculate, an approximate solution that consists of recursive Gaussian approximations has been developed. Comparisons with Monte-Carlo simulations indicate that the approximation yields more accurate results than a naïve model which does not take correlations into account. Limitations of the approximation occur for strong correlations.

The models developed in this section can be connected with other parts of the thesis and therefore lead to further conclusions. For example, instead of using an assumed value for the covariance of the state estimates, the time-dependent model given by the Cramér-Rao bound from Sec. 4.1 can be employed. Thus, a credible and throughout connection from the parameters of the sensor to the criticality estimation can be established. Parametrised models for prediction errors on the other hand can be identified with the expectation maximisation principle as described in Sec. 4.2. A comprehensive analysis of an automatic emergency brake system which comprises these aspects will be presented in Sec. 7.2.

6 Signal processing methods for the generation of reference data

This chapter presents and evaluates concepts for post processing of laser scanner measurements in order to obtain an accurate environment representation (reference data). It is organised in five sections: First, the need for reference data and the motivation to use post processing to this end are outlined in Sec. 6.1. Second, the problem is formulated in Sec. 6.2 and known on-line processing approaches are discussed. This leads to a novel concept for off-line processing that is proposed in Sec. 6.3. An implementation of the method is then experimentally evaluated as detailed in Sec. 6.4. Finally, the findings are summarised and directions for further enhancements discussed in Sec. 6.5.

6.1 Introduction

As has been seen throughout the previous sections, it is unrealistic to expect that the entirety of an intelligent vehicle's workspace may be formalised in a complete and accurate model. Therefore, tests in real-world situations remain an important part of a system's development and approval process. The following sections will discuss technical implications of such tests and devise novel approaches to the specific task of reference data generation.

According to the framework that has been developed in collaboration with ZOFKA et *al.* [220], testing involves three major tasks: First, test acceptance criteria have to be defined and formalised quantitatively in terms of a metric. Second, reference information is required to assess a system on an absolute scale. For example, ground truth distances to relevant objects may be provided by a highly accurate reference sensor. The estimation accuracy of a system under test can be evaluated by comparing the estimated values to the ground truth, for instance in terms of the mean squared error metric. Third, all relevant conditions under which a test is to be conducted, called test scenarios, have to be formalised. Especially for higher automated driving functions, the variety of traffic situations which have to be tested is abundant. One approach is to generate a parametrised description of these situations from empirical data [210]. This again requires an accurate representation of the driving environment, especially of other vehicles' trajectories.

In principle, reference data may be obtained from raw sensor measurements with the same methods as in a real-time capable signal processing chain. However, real-time operation is usually not a requirement for the generation of reference data and

instead, raw measurements may be post processed. Besides lower restrictions on the computational effort, this enables a backward propagation of information.

The following sections will discuss the benefits of post processing on the example of a laser scanner (LIDAR[1]) sensor. Due to an inherently high accuracy of the measurement principle, LIDAR sensors are frequently used in the driver assistance domain to obtain reference data [62, 156]. Solutions to the classical problem of on-line processing of automotive LIDAR measurements have been presented in numerous works, for instance in [101, 102, 194]. However, to the best of the author's knowledge, no detailed evaluations of the advantages of off-line processing have yet been published apart from a sketch of the idea given in [113].

6.2 Background on laser scanner signal processing

Laser scanner sensors obtain an environment representation in the form of a sparse depth image. Each depth value corresponds to one reflection *(scan point)* of a laser beam. In this work, only 2-D LIDAR sensors with a narrow vertical opening angle are considered. These perceive a planar depth image of the environment whereas 3-D sensors additionally give height information in high resolution. The beams are rotated by small angular increments and cover a horizontal opening angle of usually more than 90° per cycle. Due to a high angular resolution, extended objects are usually represented by multiple adjacent scan points, especially at short distances. Fig. 6.1 illustrates this measurement principle. Although the raw scan points are usually measured with a fairly high accuracy, a lack of spatial and temporal associations renders an immediate use impractical:

- The association between scan points which stem from the same object is not known.
- Scan points from multiple cycles belonging to the same object seen at different time steps are not associated.
- Moreover, scan points represent mere distances. Neither velocity information nor characteristic features of an object, as for example in images, are included.

The objective of a LIDAR signal processing chain is therefore to infer a compact and information rich environment representation from the raw measurements. According to the aforementioned aspects, this is usually achieved by three algorithmic steps as visualised in Fig. 6.2(a):

1. Scan points which are likely to belong to the same object are grouped into *segments*. This segmentation is usually based on distance measures, for example the Euclidean distance in Cartesian coordinates or, taking the angular discretisation into account, in polar coordinates [101, 102].

[1] **L**ight **d**etection **a**nd **r**anging.

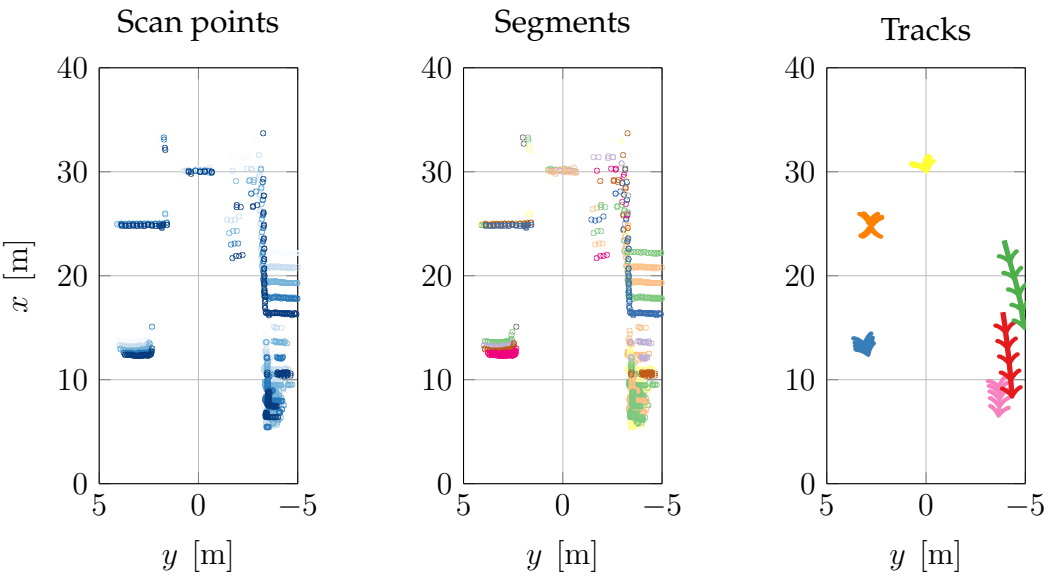


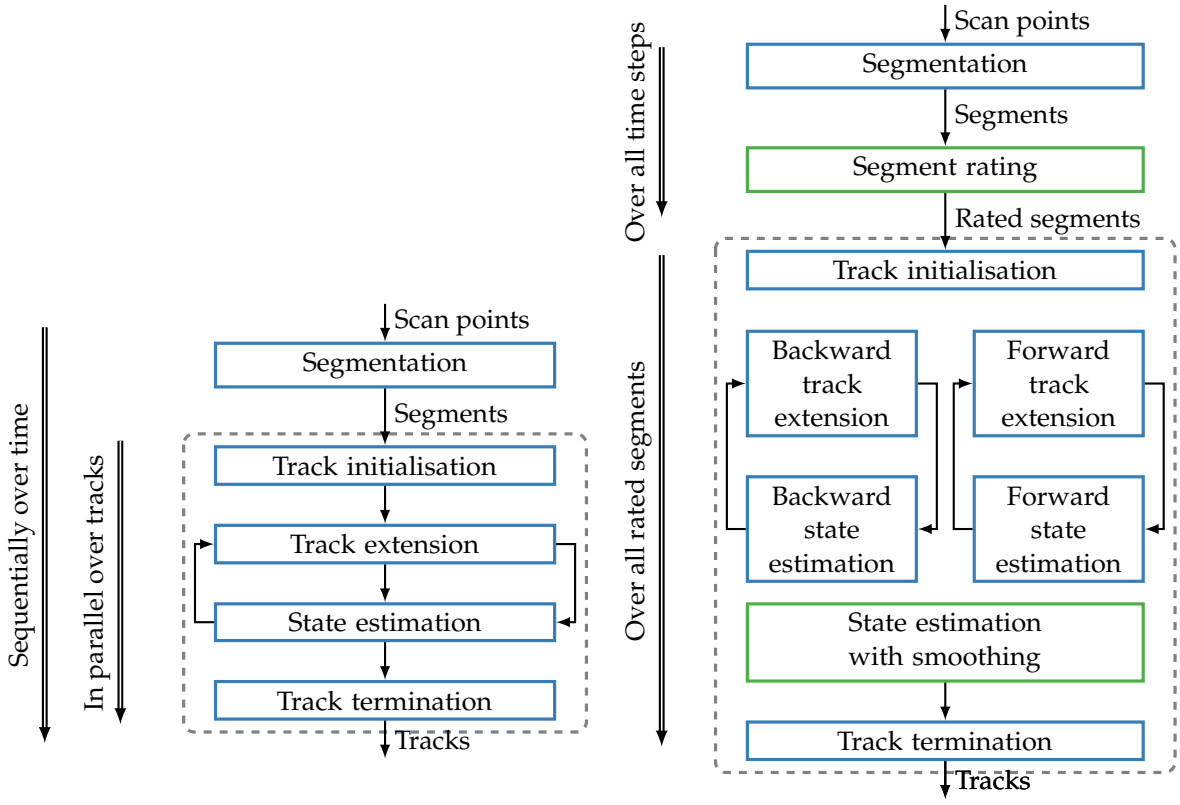
Figure 6.1 Visualisation of exemplary LIDAR measurements from an urban road with three lanes. The left image shows how the scan points evolve over a period of 1 s, from light to dark colours. After the first processing step, segments as shown in the central column result. Tracks (right column) are initialised from these segments and a continuous trajectory is estimated as indicated by coloured arrows.

2. When a temporal association between measurements from the same object is established, a *track* results. The decision to initialise a new track with scan points that cannot be assigned to one of the previously observed tracks is based on relevance criteria. For example, relevant measurements that stem from a vehicle may be distinguished from clutter if they show a distinct shape [101] or a consistent motion [194].

3. A model-based state estimator, for example an extended Kalman filter (Sec. 4.1.3.4), is employed to infer unobservable motion parameters such as velocities and accelerations for each track. To this end, scan points that have been associated to the track are condensed to a compact feature vector, for example the coordinates of the centroid [101]. The feature vector serves as a measurement for the recursive state estimator. This results in an estimate of the motion state trajectory with associated covariance for each object.

Note that the extension of a track in the second step and the motion state estimation are closely linked. For the extension of a track, scan points are searched close to locations that are expected from the current motion. Thus, a successful track extension depends on the accuracy of the motion state estimate.

For real-time applications, these algorithmic steps can only make use of past measurement information. Batch processing of an entire sequence of scan points on the other hand offers additional flexibility. The following section will discuss how this can be leveraged to achieve a more accurate and complete estimate of vehicle trajectories.



(a) On-line approach. **(b)** Off-line approach.

Figure 6.2 Outline of laser scanner signal processing. In a classical on-line approach (a), tracks are initialised, extended or terminated per time step based on the currently available measurements. One core principle of the proposed off-line approach (b) is to defer the decision for a track initialisation until sufficient evidence is available. To this end, an additional rating of the segment quality is introduced. Tracks are initialised for the best rated segments and subsequently extended in forward and backward direction. A second important part is to perform an additional backward smoothing of the state estimates.

6.3 Approaches to off-line signal processing

Classical approaches to on-line processing of laser scanner measurements are extended to novel approaches for batch processing in this section. A detailed account on the method's development and a prototypical implementation are given by WALKLING and HOCH in the supervised theses [224, 226]. Furthermore, parts of the following content can also be found in the publication [215].

The proposed approach is founded on two principle ideas:

1. Relevant objects in dynamic traffic situations are usually visible from both far and close distances. This and additional dynamic occlusions influence the number of scan points that are obtained from an object at a specific time. An off-line approach

can level this varying amount of information by backward propagation from later to earlier time steps.

2. For real-time capable operation, handling entire sets of scan points is often impractical due to limited computational means. Therefore, the raw scan point measurements are usually condensed to an abstract representation, for example bounding boxes [213]. Such restrictions can be relaxed for post processing. Thus, model-free algorithms which operate directly on scan points can be applied.

Following these ideas, it is proposed to extend the classical approach to a concept that is shown in Fig. 6.2(b). The foremost difference is that the decision for the initialisation of a track is deferred to an arbitrary time step where one has the highest quality of measurement information. A track is then extended in both backward and forward direction to those time steps where fewer and sparse scan points from the same object are available. This approach is only possible in post processing where measurements can be processed in an arbitrary order instead of strictly sequentially.

In order to enable a thoughtful choice of the initial time step, a quantitative measure is employed to rate the quality of all segments from the entire sequence of measurements. This continuous measure considers two characteristics of the scan points that belong to a segment:

1. A high number of scan points indicates a rigid object with a relevant size.

2. The goodness of fit between the scan points and a model of the typical appearance of relevant objects is measured in terms of the residual error. For example, cuboid objects, such as vehicles, appear as one, two or three perpendicular line segments in the 2-D depth image [101].

The central idea is then to initialise tracks only where sufficient evidence indicates the presence of a relevant object. This ensures that the initial motion state estimates can be obtained from a sufficient number of scan points. Therefore, the criteria for a track initialisation are checked for each segment in the order of the quality rating. One criterion is that a consistent and plausible motion exists over neighbouring time steps. To this end, the rotation and translation between scan point clouds are estimated with the help of methods from Sec. 3.2.3.

If a track has been successfully initialised, it is extended in both forward and backward direction. To extend a track, all scan points in neighbouring time steps are matched against predictions according to the state estimates at the current boundaries of the track. In order to increase the accuracy of these state estimates and thus the chances of a successful track extension,[2] the first few iterations are performed in alternating order. This approach is illustrated in Fig. 6.3.

[2]This is especially problematic for non-linear models, where a state estimate is used as the linearisation point. If the estimate is far away from the true state, the linearised model might be very inaccurate and the filter diverges.

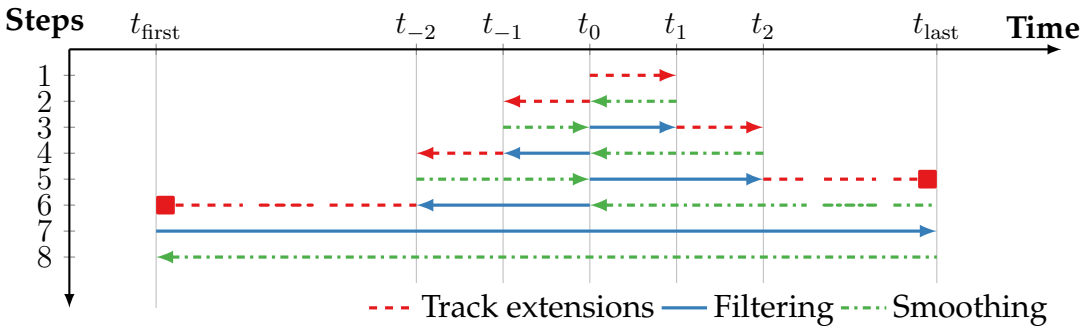


Figure 6.3 Sequence of track extensions and state estimation updates. During the first steps (1–4), the track extension is performed in forward and backward direction in alternating order. After this start-up phase, further extensions are conducted individually per direction (5–6), until the track termination criterion is met. Lastly, the state is estimated by a final filtering and smoothing operation over the entire track (7–8).

Instead of extending a track separately in forward and backward direction, the motion state estimates are refined each time that an extension has been performed. To this end, a new estimate of the state at the initial time step is calculated according to the backward pass of an extended Kalman smoother from (4.58a). Subsequently, the filtering steps in the opposite direction are repeated with this new initial state.

A track is terminated in either direction if a consistent match with the available scan points cannot be established. The final state estimate is then obtained by a forward filtering and backward smoothing pass over the whole trajectory. Furthermore, all scan points that have been associated to the current track are removed from the pool of rated segments. Then, the track initialisation and extension routine is repeated with the next best rated segment from the remaining set.

In summary, the proposed approach to off-line reference data generation consists of methods specifically tailored to LIDAR sensors, but also of other generic concepts. The segment quality rating relies on the specific characteristics of laser scan points. On the other hand, the iterative backward propagation of information is based on the generic Kalman smoother algorithm and thus applicable to other measurement principles.

6.4 Evaluation

The goal of the following experimental evaluations is to reveal how off-line processing can improve the track quality. One aspect of track quality is the accuracy of the state estimates, which is assessed in Sec. 6.4.1. A second measure of track quality, analysed in Sec. 6.4.2, is the average track length [174]. This metric indicates if tracks are initialised with a delay or terminated prematurely when comparing two algorithms on the same dataset.

In order to evaluate the advantages gained by off-line processing, an on-line capable algorithm as visualised in Fig. 6.2(a) is used as a baseline. This implementation performs

Figure 6.4 Experimental vehicles for evaluation of the state estimation accuracy. The laser scanner sensor is mounted at the front of the first vehicle. Both vehicles carry inertial measurement units coupled with differential GPS for precise self-localisation.

track extensions solely in the forward direction. Moreover, state estimates are obtained by an extended Kalman filter whereas the off-line algorithm uses an additional smoothing pass. Both filters are designed for a CTRA motion model (Sec. A.2).

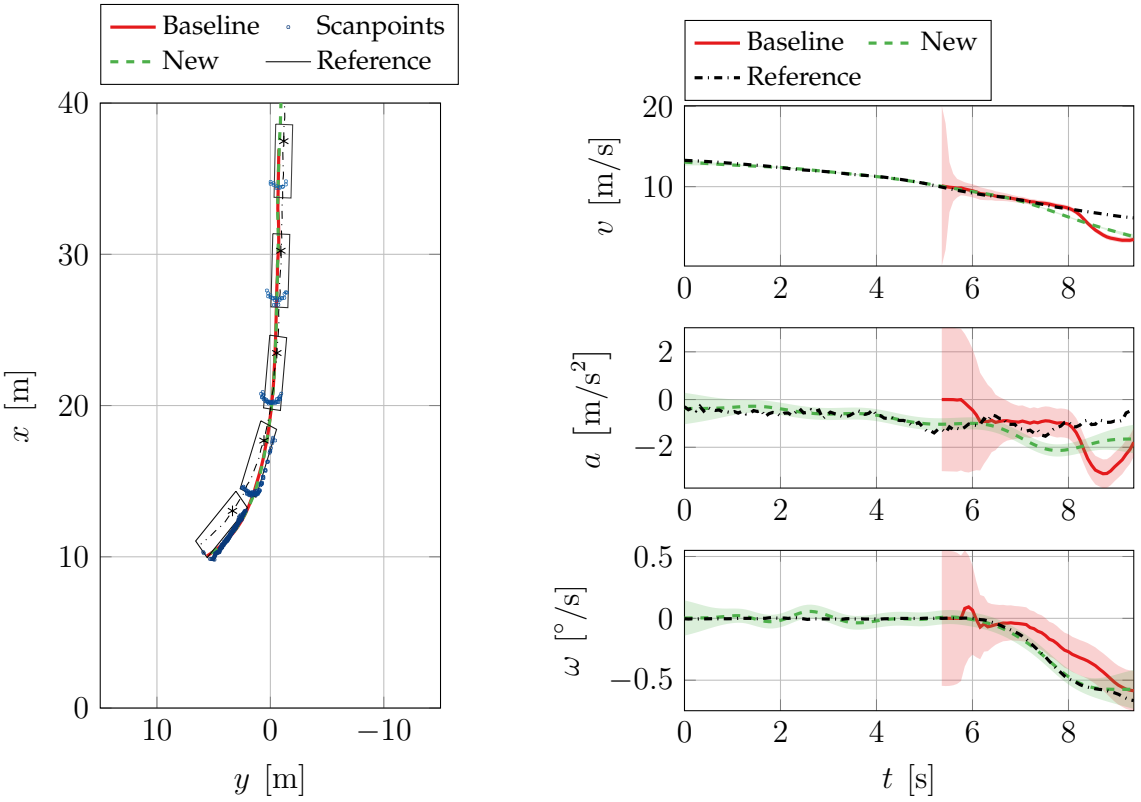
All remaining parts of the two implementations, for example the method used for segmentation, as well as the criteria for the initialisation and termination of tracks, are identical. The rationale for this choice of the baseline algorithm is that the evaluation shall focus on the methodological differences and not the hardly comparable parametrisations of two different algorithms.

6.4.1 State estimation accuracy

In order to quantitatively assess errors in the estimated motion state, a source of sufficiently accurate ground truth information is needed. Following the approach presented in [29], two experimental vehicles shown in Fig. 6.4 are equipped with dedicated inertial measurement units[3] (IMU) coupled with differential GPS for precise self-localisation. Experiments are conducted on a closed test track. The ego-vehicle, which carries a 2-D laser scanner with four beams[4], remains stationary at the origin of a Cartesian coordinate system. Laser scanner measurements and the estimated trajectories are given in these coordinates. Since the global poses of both vehicles are known, ground truth values for the target vehicle's trajectory can be calculated and compared to the estimates.

The manoeuvre that is driven during the experiment consists of an accelerated approach and turn across in front of the ego-vehicle. Thereby, accelerations as well as rotations are included. This trajectory and the motion state are shown in Fig. 6.5. It can be seen that the track which results from post processing starts much earlier than the

[3] *Automotive Dynamic Motion Analyzer (ADMA)* by GeneSys Elektronik GmbH.
[4] *ibeo LUX 2010* by Ibeo Automotive Systems GmbH.



(a) Trajectory **(b)** States

Figure 6.5 The target vehicle approaches the ego-vehicle in a left turn across manoeuvre. The ego-vehicle remains stationary at the origin in (a). Here, the laser scanner measurements (blue) and trajectory estimates (red, green) are visualised (for visual clarity, only every 10th cycle is shown). Ground truth positions are illustrated in black. Estimates of velocity, acceleration and yaw rate with their estimated covariance as indicated by the shaded $\pm 1\sigma$ intervals are shown in (b).

one given by the baseline algorithm. A second observation that is visible from Fig. 6.5(b) concerns the improved accuracy of the state estimates. With an additional smoothing pass, the naturally high uncertainty at the beginning of the track is remedied.

In order to quantitatively assess this difference, the experiment is repeated ten times. The sample mean $\mu_{\Delta(\cdot)}$ and sample standard deviation $s_{\Delta(\cdot)}$ of the differences $\Delta(\cdot)$ to the ground truth values in velocity v, acceleration a and yaw rate ω are calculated over all cycles. As an additional figure for comparison, the state estimates of the new algorithm prior to the smoothing pass are included. Thus, the benefit that is gained by backward smoothing of the same measurements is evaluated. The results in Tab. 6.1 reveal that smoothing alone reduces the standard deviation of the estimates by 15…58 %. The improvement is even higher when comparing with the baseline algorithm. Due to the later track initialisation, fewer measurements are available for the state estimation which negatively affects the accuracy.

Table 6.1 Error statistics for five turn across manoeuvres. With forward processing alone (first row), shorter tracks and higher standard deviations are obtained. The new algorithm with backward tracking and smoothing consistently achieves the highest accuracy. In comparison to the same estimates prior to smoothing (middle row), the standard deviation is reduced by 15…58 %.

		$v\,[\mathrm{m/s}]$		$a\,\left[\mathrm{m/s^2}\right]$		$\omega\,[\mathrm{rad/s}]$	
Approach	**Cycles**	$\mu_{\Delta v}$	$s_{\Delta v}$	$\mu_{\Delta a}$	$s_{\Delta a}$	$\mu_{\Delta \omega}$	$s_{\Delta \omega}$
Baseline	523	−0.557	1.310	−0.038	1.21	0.050	0.128
New (filtered)	1212	−0.227	0.849	−0.003	0.767	0.021	0.095
New (smoothed)	1212	−0.312	0.723	−0.185	0.601	0.003	0.040

6.4.2 Track length

The purpose of the second experiment is to compare how much information is extracted from the raw LIDAR measurements. To this end, the number and lengths of the generated tracks are compared.

Recordings from real traffic scenes are used to achieve a more realistic setting than on a test track. Divided in categories of different street types – urban and rural roads as well as motorways – three datasets comprising 30 min each are analysed.

Before the track lengths can be evaluated in a meaningful way, a selection of relevant objects has to be extracted. The objective is to focus on moving road users, for example cars or trucks, but to exclude the roadside infrastructure. Making this distinction can be achieved by a number of heuristics or machine learning techniques. Since the details of such approaches are not in the scope of this work, classification results are obtained by a built-in routine of the employed LIDAR sensor. Only tracks which have been confirmed as vehicles are included in the evaluation. Moreover, the median track lengths are only compared for objects which are present in the results of both the baseline and the proposed algorithm. Multiple short tracks from the baseline algorithm may be associated to the same longer track given by the new algorithm.

Tab. 6.2 shows the resulting figures and median track lengths per dataset. It can be seen that on average, 15…31 % longer tracks are achieved by backwards track extension. This figure is lower than the difference in the overall cycle count of the test track results in Tab. 6.1. However, those results were obtained in an artificial setting with the ideal condition of an unobstructed view. The longest tracks are achieved on motorways where the road layout causes fewer occlusions than on narrow and curvy streets.

6.5 Summary and outlook

This chapter has discussed and evaluated algorithms for post processing of laser scanner measurements. The resulting tracks can be used as reference values for an empirical

Table 6.2 Statistics on tracks found in laser scanner recordings with a duration of 30 min per street type. On motorways, fewer and longer tracks are initiated due to the separation of the two traffic directions. Overall, the median track lengths can be increased by the backward tracking approach.

		Urban	**Rural**	**Motorways**
Number of all tracks	Baseline	2086	2375	1273
	New	2505	2727	1597
Number of tracks confirmed as vehicles	Baseline	319	222	150
	New	403	271	182
Number of common tracks	Baseline	286	178	129
	New	308	182	127
Median length of common tracks	Baseline	2.4 s	2.0 s	6.4 s
	New	3.1 s (+31.1 %)	2.5 s (25.0 %)	7.8 s (+20.5 %)

evaluation of other sensors or an automatic generation of test scenario descriptions. The main principle is to use backward propagation of measurement information to enhance the track quality. Experimental evaluations of a prototypical implementation demonstrate the advantages of this approach.

The methods discussed in this chapter could be enhanced by additional means that leverage the potential of post processing in other algorithmic steps. One example is a dedicated analysis of all generated tracks in order to identify an erroneous merging of separate objects. For instance, two motorcyclist who drive side by side may be falsely regarded as one vehicle. However, if the trajectories separate at a later time, this knowledge could be propagated backwards to initialise two tracks right from the beginning.

Moreover, this work has not studied the inference of semantic class labels or the estimation of object contour models. Both tasks are in general challenging in an on-line approach since the available scan points are often sparse. Nevertheless, post processing of a sequence offers the possibility to infer this information once and to propagate it to the entire the track.

7 Applications

7.1 Accuracy requirements for localisation in urban automated driving

Landmark-based map matching enables self-localisation of intelligent vehicles if satellite navigation is unavailable. The achievable localisation accuracy is related to the choice of sensors for measuring landmark positions. Based on the previously developed models, this section investigates the relation between parameters of a stereo vision sensor and the localisation accuracy.

First, the task is further outlined in Sec. 7.1.1, followed by a review of related works in Sec. 7.1.2. Thereafter, model assumptions and the localisation problem formulation are stated in Sec. 7.1.3. Sensor parameter constraints are then derived in Sec. 7.1.4. The validity of the obtained results is demonstrated with simulations in Sec. 7.1.5. An outlook on possible future extensions in Sec. 7.1.6 concludes this section.

7.1.1 Introduction

The goal of an efficient engineering process is to quickly find a system design which meets the application requirements and to minimise the overall costs at the same time. In the domain of intelligent vehicles, the system design includes among other aspects the selection of exteroceptive sensors, their characteristics (for example measurement accuracy and field of view) and the implementation of suitable algorithms on an embedded device. Costs can be attributed to the chosen sensor, the electronic control unit but also the number and duration of iterations in the development process. It has been motivated in Sec. 1.1 that (analytical) models can help to accelerate this process. This claim will be exemplified in the following with a model-based derivation of sensor parameter constraints for a localisation task. As the overview in Fig. 7.1 shows, the analysis is based on the statistical models of stereo cameras from Sec. 3.1 and feature-based localisation from Sec. 3.2.

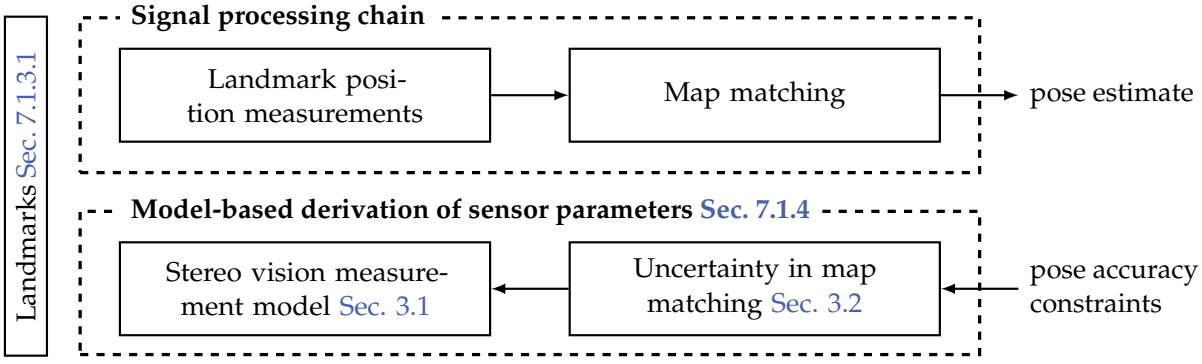


Figure 7.1 Signal processing chain of feature-based localisation. Uncertainty in the pose estimates is modelled according to the lower part. Inverting these models will lead to requirements on the sensor parameters in order to reach a desired localisation accuracy.

The objective is to localise the ego-vehicle with a pre-defined maximum allowed variance of the pose estimates. To this end, a known map which comprises the positions of georeferenced landmarks (for example lane markings [155] or traffic signs [158]) is assumed to be available. These landmarks can be partially observed by a stereo vision camera. Thus, the current vehicle pose relative to the map can be estimated by matching the information from measurements and map.

The goal of this section is to derive a relationship between the pose estimation uncertainty and the parameters of the stereo vision camera, for example the disparity estimation error. However, the achievable localisation accuracy also depends on other factors, such as the number and spacing of landmarks in the sensor's field of view.

According to the discussion from Sec. 1.2, there are two approaches for finding sensor parameter values so that the accuracy requirements are met. On the one hand, a prototypical system can be evaluated as a *black-box* in the real world. This approach yields results under realistic conditions but has the drawbacks that it is costly to implement the prototypical sensors and to conduct the field tests. In general, only a single sensor configuration can be assessed at a time. Therefore, many iterations are required to find a sensible combination of parameter values in a potentially multidimensional space.

On the other hand, *white-* or *grey-box* models can be employed to solve this problem purely analytically or numerically [35]. Even though these models might rely on idealised assumptions and simplified representations of real-world phenomena, they can achieve a sensible initial parametrisation. This can be further refined with empirical evaluations. For example, optimistic assumptions might be introduced for all necessary approximations. Thus, the obtained parameters values of the sensor measurement noise are the necessary minimum requirements.

As outlined in Sec. 1.2, one can obtain the propagation of sensor measurement noise in the pose estimation uncertainty by either numerical or analytical methods. In the first case, random realisations of the sensor noise and all other variable influences are drawn and a non-parametric distribution of the pose estimates is obtained. Therefore, an inefficient grid search on the parameter space of the sensor parameters has to be conducted.[1] In contrast, an analytical model can be inverted under certain conditions which yields explicit expressions for the sensor parameters. This method, which has been previously published in [211], will be further elaborated in the following.

7.1.2 Related work

While approaches for landmark-based localisation are usually evaluated and compared to other methods in terms of the localisation error, only few works investigate the relation

[1] Note that in the general case, the pose estimates are obtained by solving the matrix-weighted Procrustes problem from Sec. 3.2.3.2. Since no closed-form solution is available, this requires a numerical, iterative approach. Thus, a Monte-Carlo simulation has to iterate over an iterative algorithm even for a single combination of all parameter values. In contrast, the analytical model from Sec. 3.2.4.2 provides a closed-form expression for the variance. This is helpful since the computational burden increases if a further level of iterations, namely over the system and scene parameters, is introduced.

to variable parameters.

To the best of the author's knowledge, the most comprehensive discussion can be found in [14]. There, the influence of the landmark configurations is analysed and strategies for their optimal placement are developed. This approach supposes that landmarks are artificial and may be freely placed in the environment. While this assumption is reasonable in a controlled environment, such as a factory, this does not generalise to the road environment. Instead of the landmark configuration, parameters of the sensors have to be optimised in order to reach a desired localisation accuracy. Additionally, the variability of landmark occurrences has to be taken into account.

A second remarkable work by BANSAL et *al.* [9] presents an empirical evaluation of localisation errors in relation to the parameters of a camera-based system. In order to relax the usual restriction that only a single sensor can be analysed at a time, images are first recorded by an omnidirectional panoramic camera. Subsequently, a virtual camera configuration with an arbitrary mounting angle and field of view can be emulated from this data. Since empirical measurements are used, this approach promises a high degree of realism. However, the method is limited to a fixed type and mounting position of the (virtual) sensor. These limitations are overcome in a purely model-based approach as is pursued in the following.

7.1.3 Models and problem formulation

The models which will be used to derive the localisation accuracy constraints will be introduced in this section. This includes a probabilistic representation of landmarks and stereo vision measurements.

Notations and coordinate systems are first defined. A global map is represented as a two-dimensional Cartesian coordinate system with an arbitrary origin, as is visualised in Fig. 7.2. This map contains a set $\mathcal{M} = \{\mathbf{m}_j\}_{j=1:M}$ of M landmarks with known positions $\mathbf{m}_j \in \mathbb{R}^2$. Furthermore, an arbitrarily oriented and possibly curved road is assumed. An exact model of the road in the map is not required since instead, we will focus on the world as seen from a vehicle on this road.

To this end, a Cartesian vehicle coordinate system is introduced which is shifted by a translation $\mathbf{t}$ relative to the origin of the map and rotated by an angle θ. Thus, $\mathbf{t}$ is the position of the vehicle in the map and θ the current driving direction. This pose is to be estimated from sensor measurements of the landmark positions relative to the vehicle. The N currently visible landmarks in the vehicle coordinate system $\mathbf{p}_i \in \mathbb{R}^2$ form a set $\mathcal{P} = \{\mathbf{p}_i\}_{i=1:N}$.

Three models will be detailed in the following. Firstly, the spatial occurrence of landmarks is modelled according to a generic framework in Sec. 7.1.3.1. Secondly, the sensor measurement model of a stereo vision camera, as detailed in Sec. 3.1, is recalled in Sec. 7.1.3.2. In the third part Sec. 7.1.3.3, the pose estimation task will be formulated. A model of the pose estimation uncertainty has been previously derived in Sec. 3.2.

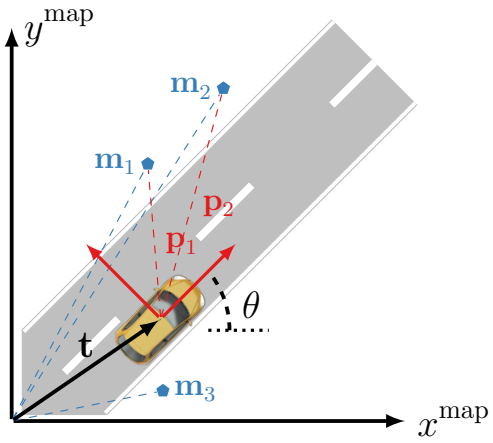


Figure 7.2 Illustration of the map (black) and vehicle (red) coordinate systems. Three landmarks with known locations $\{\mathbf{m}_1, \mathbf{m}_2, \mathbf{m}_3\}$ are part of the scene. Two of these are measured by the vehicle's front-facing sensor as $\{\mathbf{p}_1, \mathbf{p}_2\}$. The vehicle's pose $(\mathbf{t}, \theta)$ is estimated by matching the feature locations.

7.1.3.1 Landmarks

The central requirement on landmarks is that they are stationary and can be reproducibly detected by the available exteroceptive sensor. Different domain- and sensor-specific features are used to this end.

Lane markings [155] or traffic signs [158] are typical candidates for frequent stationary elements in the road environment. Their distinct appearance facilitates the development of detection algorithms. Other types of landmarks are given by abstract features that are derived from raw sensor measurements. For instance, image features such as the DIRD descriptor can be employed [115]. Since the certainty of a pose estimate increases with the number of available landmarks, a combination of different types is also a viable approach [208].

In the following, relevant types of landmarks are assumed to be given by distinct road markings or belong to the roadside infrastructure. In order to take the environmental variability into account, a probabilistic model of the occurrence of landmarks is employed. However, it is not the scope of this thesis to develop models of the distribution of certain real-world infrastructure elements and the interplay with a specific sensor. This topic is pursued by ROHDE *et al.* [212]. Therefore, the framework that is proposed in the following is a generic preliminary approach only.

In order to minimise the complexity of the parametric model, landmarks are described in the coordinate system of the vehicle, and thus relative to the road, instead of in the global map. The rationale is that the map coordinates are in general not aligned in a specific way with the course of the road. Thus, the occurrence of roadside-related landmarks is easier described relative to the road itself.

The only relevant relation between road and map is assumed to be a rotation by an angle θ_{road}. Since this alignment can be arbitrary, a uniform distribution $\theta_{\text{road}} \sim \mathcal{U}(0, 2\pi)$

is employed. It is furthermore supposed that the vehicle is following the road and thus the vehicle orientation θ equals θ_{road}.

The position of a landmark in the vehicle coordinate system is denoted by $\mathbf{m}_j^{\text{veh}}$, $j = 1, \ldots, M$. Note that although they are represented in the same coordinate system, there are two differences between the $\mathbf{m}_j^{\text{veh}}$ and the sensor measurements $\mathbf{p}_i$: Firstly, landmarks are modelled independently of the sensor's field of view. Thus, the overall number M is in general higher than the number of visible landmarks N. Secondly, the measured position of a landmark $\mathbf{p}_i$ is affected by sensor noise whereas $\mathbf{m}_j^{\text{veh}}$ denotes the true position. In this section, only the distribution of the true landmark positions is modelled whereas the measurement model is detailed subsequently.

Firstly, the longitudinal distance Δm_x^{veh} between two landmarks along the road is assumed to follow an exponential distribution with a rate parameter λ_{LM}:

$$\Delta m_x^{\text{veh}} \sim \lambda_{\text{LM}} \exp\left(-\lambda_{\text{LM}} \Delta m_x^{\text{veh}}\right) , \qquad \Delta m_x^{\text{veh}} \geq 0 . \tag{7.1a}$$

A remarkable consequence of this assumption is that it easily generalises to the case of $l = 1, \ldots, L$, independent kinds of landmarks with individual rates $\lambda_{\text{LM},l}$. Then, the minimum distance between any two landmark occurrences is distributed exponentially as well with an overall rate $\lambda_{\text{LM}} = \sum_{l=1}^{L} \lambda_{\text{LM},l}$.

Secondly, a Gaussian distribution with a variance of σ_y^2 is assumed to model the occurrence in the lateral direction. Landmarks are spaced around a mean value μ_{m_y}, for example at the road boundary for traffic signs:

$$m_y^{\text{veh}} \sim \mathcal{N}\left(\mu_{m_y}, \sigma_y^2\right) . \tag{7.1b}$$

An exemplary landmark distribution is visualised in Fig. 7.3(a).

As will be seen in the following, it is partially required to resort to numerical methods even with this simple formulation. Therefore, the assumptions of the preliminary model are not crucial for the upcoming derivations and can be safely relaxed to allow for more sophisticated models.

7.1.3.2 Measurement model

A perception system comprises the physical measurement principle of one or multiple sensors as well as the employed algorithms for landmark detection with respective parametrisations. Its relevant characteristics are the probability of detection and the measurement noise covariance. These may furthermore depend on a landmark's type, its position relative to the sensor and other external effects such as dynamic occlusions.

In order to make this generic characterisation more explicit, a stereo vision sensor is considered. Specifically, this work focusses on the distance dependence of the measurement noise while other effects are modelled under idealised assumptions.

Firstly, the capability of a sensor to detect landmarks in the surroundings is modelled spatially in terms of a field of view. The field of view is defined in a coordinate system

relative to the sensor. For simplicity, but without loss of generality, it is assumed that this frame coincides with the vehicle coordinates. The area in which landmarks are detected is parametrised in terms of a maximum range $r_{\text{sens,max}}$ and opening angle $\alpha_{\text{sens,max}}$. Thus, a landmark in vehicle coordinates $\mathbf{m}_j^{\text{veh}}$ is detected if:

$$\left\|\mathbf{m}_j^{\text{veh}}\right\| \leq r_{\text{sens,max}} \qquad \text{and} \qquad -\frac{\alpha_{\text{sens,max}}}{2} \leq \angle \mathbf{m}_j^{\text{veh}} \leq \frac{\alpha_{\text{sens,max}}}{2} \,. \tag{7.2}$$

This deterministic model could be generalised to a continuous probability of detection. It is furthermore supposed that the correspondences between measured landmarks and those in the map are ideally known. Therefore, a subset $\tilde{\mathcal{M}} \subseteq \mathcal{M}$ contains the map locations of all $i = 1, \ldots, N$ landmarks $\tilde{\mathbf{m}}_i$ which correspond to the currently observed features $\mathbf{p}_i$.

Secondly, the uncertain position measurement of a landmark is modelled as a Gaussian random variable $\mathbf{p}_i \sim \mathcal{N}\left(\boldsymbol{\mu}_{\mathbf{p}_i}, \boldsymbol{\Sigma}_{\mathbf{p}_i}\right)$. For the assumed stereo vision camera, the covariance $\boldsymbol{\Sigma}_{\mathbf{p}_i}$ depends on the landmark position. Similar to the derivation in Sec. 3.1.4.4, independent measurement noise in disparity d and image column number u is transformed to Cartesian coordinates by linearisation:

$$\boldsymbol{\Sigma}_{\mathbf{p}_i} = \frac{\mu_{p_{x,i}}^2}{\left(c_{\text{k}} b_{\text{w}}\right)^2} \begin{bmatrix} \mu_{p_{x,i}}^2 \sigma_d^2 & \mu_{p_{x,i}} \mu_{p_{y,i}} \sigma_d^2 \\ \mu_{p_{x,i}} \mu_{p_{y,i}} \sigma_d^2 & \mu_{p_{y,i}}^2 \sigma_d^2 + b_{\text{w}}^2 \sigma_u^2 \end{bmatrix} \approx \underbrace{\frac{\sigma_d^2}{\left(c_{\text{k}} b_{\text{w}}\right)^2}}_{=:\xi_{\text{SV}}} \cdot \underbrace{\begin{bmatrix} \mu_{p_{x,i}}^4 & \mu_{p_{x,i}}^3 \mu_{p_{y,i}} \\ \mu_{p_{x,i}}^3 \mu_{p_{y,i}} & \mu_{p_{x,i}}^2 \mu_{p_{y,i}}^2 \end{bmatrix}}_{=:\boldsymbol{\Sigma}_{\mathbf{p}_i}^0} \,. \tag{7.3}$$

This camera model is defined by the constant parameters base-width b_{w}, focal length in pixels c_{k} and measurement noise variances σ_d^2 and σ_u^2 in disparity and image column, respectively. Furthermore, (7.3) introduces an approximation that is based on two assumptions:

1. σ_d is of the same order of magnitude as σ_u. This is reasonable since the disparity value d is estimated from the difference of noisy image column values u.
2. $\mu_{p_{y,i}}^2 \gg 0.5 b_{\text{w}}$. For landmarks which are located sideways to the vehicle's driving path, the lateral position $\mu_{p_{y,i}}$ is multiple times larger than the base-width b_{w} between the two camera image planes.

In the approximate model, all sensor parameters are combined in a scalar coefficient ξ_{SV}. This will simplify the model inversion that is necessary to find suitable parameter values in Sec. 7.1.4.

7.1.3.3 Landmark-based localisation

Given the two sets $\tilde{\mathcal{M}}, \mathcal{P}$ which contain the positions of $i = 1, \ldots, N$ corresponding landmarks in the map and vehicle coordinates, the pose parameters $\mathbf{t}$ and θ are to be estimated. The definition from (3.31) is recalled here using the notation of this section:

$$\tilde{\mathbf{m}}_i = \mathbf{R}\mathbf{p}_i + \mathbf{t} \,, \qquad i = 1, \ldots, N \,. \tag{7.4}$$

The measurement noise is anisotropic and heteroscedastic as seen from the covariance (7.3). Therefore, the matrix-weighted Procrustes problem from Sec. 3.2.3.2 has to be solved with an appropriate numerical scheme.

The pose estimates' variance has been analytically derived in Sec. 3.2.4.2. Note that since the map is assumed to be accurate, only the gradients with respect to the sensor measurements have to be obtained:

$$\sigma_\theta^2 = \sum_{i=1}^{N} \left(\nabla_{\mathbf{p}_i} \hat{\theta}\right) \boldsymbol{\Sigma}_{\mathbf{p}_i} \left(\nabla_{\mathbf{p}_i} \hat{\theta}\right)^{\mathrm{T}} , \tag{7.5a}$$

$$\boldsymbol{\Sigma}_{\mathbf{t}} = \left(\frac{\partial}{\partial \theta} \hat{\mathbf{t}}\right) \sigma_\theta^2 \left(\frac{\partial}{\partial \theta} \hat{\mathbf{t}}\right)^{\mathrm{T}} + \sum_{i=1}^{N} \left(\nabla_{\mathbf{p}_i} \hat{\mathbf{t}}\right) \boldsymbol{\Sigma}_{\mathbf{p}_i} \left(\nabla_{\mathbf{p}_i} \hat{\mathbf{t}}\right)^{\mathrm{T}} . \tag{7.5b}$$

The complete expressions for the derivatives are given by (3.54) inserted in (3.47) and (3.55). They are independent of the landmark coordinates $\mathbf{m}$ in the map. Thus, the estimation error is not affected by the arbitrary definition of a map's origin.

7.1.4 Derivation of sensor parameter requirements

The covariance $\boldsymbol{\Sigma}_{\mathbf{t}}$ of the estimated vehicle position $\hat{\mathbf{t}}$ depends on the covariance $\boldsymbol{\Sigma}_{\mathbf{p}_i}$ of the individual landmark observations and therefore the sensor parameters. Given a requirement on the estimate's covariance matrix $\boldsymbol{\Sigma}_{\mathbf{t}}$, for example upper bounds on the diagonal elements, necessary requirements on the sensor parameter values are to be derived. In the general case, inverting the model (7.5b) is not trivial due to the involved matrix multiplications and possibly non-linear relationships. However, in a special case where the relevant sensor parameters can be expressed as a scalar coefficient in front of the measurement error covariance, a closed-form solution is possible.

As an example of this special case, the approximate measurement model of the stereo vision camera from (7.3) will be considered. Inserting this model into (7.5b) yields the following result. Here, $\boldsymbol{\Sigma}_{\mathbf{p}_i}^0$ denotes a part which depends only on the landmark configuration:

$$\boldsymbol{\Sigma}_{\mathbf{t}} = \xi_{\mathrm{SV}} \underbrace{\left[\sum_{i=1}^{N} \left(\frac{\partial}{\partial \theta} \hat{\mathbf{t}}\right) \left(\nabla_{\mathbf{p}_i} \hat{\theta}\right) \boldsymbol{\Sigma}_{\mathbf{p}_i}^0 \left(\nabla_{\mathbf{p}_i} \hat{\theta}\right)^{\mathrm{T}} \left(\frac{\partial}{\partial \theta} \hat{\mathbf{t}}\right)^{\mathrm{T}} + \left(\nabla_{\mathbf{p}_i} \hat{\mathbf{t}}\right) \boldsymbol{\Sigma}_{\mathbf{p}_i}^0 \left(\nabla_{\mathbf{p}_i} \hat{\mathbf{t}}\right)^{\mathrm{T}}\right]}_{=:\boldsymbol{\Sigma}_{\mathbf{t}}^0} . \tag{7.6}$$

In a second step, a requirement on $\boldsymbol{\Sigma}_{\mathbf{t}}$ has to be solved for the sensor parameters. In general, requirements may be expressed as an inequality on a vector-valued function of $\boldsymbol{\Sigma}_{\mathbf{t}}$. For example, it may be required that the variances of the components t_x and t_y, that are the diagonal elements $\Sigma_{\mathbf{t},11}$ and $\Sigma_{\mathbf{t},22}$, remain below some tolerable limits $\bar{\sigma}_x^2$ and $\bar{\sigma}_y^2$:

$$\begin{bmatrix} \Sigma_{\mathbf{t},11} \\ \Sigma_{\mathbf{t},22} \end{bmatrix} = \xi_{\mathrm{SV}} \begin{bmatrix} \Sigma_{\mathbf{t},11}^0 \\ \Sigma_{\mathbf{t},22}^0 \end{bmatrix} \overset{!}{\leq} \begin{bmatrix} \bar{\sigma}_x^2 \\ \bar{\sigma}_y^2 \end{bmatrix} \Rightarrow \xi_{\mathrm{SV}}^{-1} \geq \max\left(\frac{\Sigma_{\mathbf{t},11}^0}{\bar{\sigma}_x^2}, \frac{\Sigma_{\mathbf{t},22}^0}{\bar{\sigma}_y^2}\right) . \tag{7.7}$$

The requirement on ξ_{SV} may be further broken down into requirements on individual parameters, for example the disparity noise σ_d^2.

It has to be remarked that the appealing simplicity of (7.7) depends on how the requirements on $\boldsymbol{\Sigma}_{\mathbf{t}}$ are expressed. For other relevant representations, for example inequalities on the eigenvalues (length of the principal axes of the covariance ellipses) or the determinant (proportional to the squared area of the covariance ellipses), similar factorised results can be obtained.

The landmark-dependent part $\boldsymbol{\Sigma}_{\mathbf{t}}^0$ in (7.6) is assumed as constant and known. However, as it has been motivated in Sec. 7.1.3.3, the landmark configuration is highly variable and modelled with a probability distribution. Thus, the covariance $\boldsymbol{\Sigma}_{\mathbf{t}}$ can be regarded as a random variable. Instead of a deterministic requirement (7.7), one demands that an inequality is satisfied $1-\alpha$ percent of the time:

$$\mathrm{P}\left(\xi_{\mathrm{SV}} \begin{bmatrix} \Sigma_{\mathbf{t},11}^0 \\ \Sigma_{\mathbf{t},22}^0 \end{bmatrix} \leq \begin{bmatrix} \bar{\sigma}_x^2 \\ \bar{\sigma}_y^2 \end{bmatrix}\right) \overset{!}{\geq} 1-\alpha \Rightarrow \xi_{\mathrm{SV}}^{-1} \geq Q_\eta\left(1-\alpha\right) \ , \tag{7.8}$$

where $Q_\eta\left(1-\alpha\right)$ refers to the $1-\alpha$ quantile of the random variable

$$\eta = \max\left(\frac{\Sigma_{\mathbf{t},11}^0}{\bar{\sigma}_x^2}, \frac{\Sigma_{\mathbf{t},22}^0}{\bar{\sigma}_y^2}\right) \ . \tag{7.9}$$

The quantile function follows from the probability distribution of the number and spacing of landmarks that are visible to the sensor. Therefore, it depends on the landmark occurrence in the world as modelled by (7.1) and the limited field of view of the sensor as defined in (7.2). Due to the further non-linear mapping to the translation error variance in (7.6), an analytical expression is unlikely to exist in other than very simple scenarios. Therefore, (7.8) is numerically evaluated in two steps.

Firstly, the cumulative probability distribution $\mathrm{P}\left(\eta\right)$ of η is approximated by Monte-Carlo simulations. To this end, $e = 1, \ldots, N_{\mathrm{s}}$ independent map feature samples $\mathcal{M}^{(e)}$ are drawn using the model (7.1). After filtering these according to the sensor's field of view (7.2), the sets of visible landmarks $\tilde{\mathcal{M}}^{(e)}$ are obtained. These sets are inserted into the model of the location estimate's variance (7.6) which yields N_{s} values for $\boldsymbol{\Sigma}_{\mathbf{t}}^0$ and thus $\eta^{(e)}$, $e = 1, \ldots, N_{\mathrm{s}}$. Finally, a non-parametric estimate $\hat{P}\left(\eta\right)$ of the cumulative distribution $P\left(\eta\right)$ is given by the quantiles of the sample. Secondly, this numerical distribution is solved for ξ_{SV}^{-1} by finding the argument at which the cumulative probability takes on the desired value $1-\alpha$ according to (7.8).

Since the Monte-Carlo estimate relies on randomly drawn landmark samples, the estimated probabilities $\hat{P}\left(\eta\right)$ are random as well. The fluctuation of the result depends on the number of samples N_{s}. Without requiring further knowledge on the distribution of the samples, the CHERNOFF bound provides an inequality on the absolute estimation error [184]:

$$1-\delta \coloneqq \mathrm{P}\left(\left|\hat{P}\left(\eta\right) - P\left(\eta\right)\right| \leq \epsilon\right) \leq 2\exp\left(-2N_{\mathrm{s}}\epsilon^2\right) \ . \tag{7.10}$$

This probability inequality can be interpreted and utilised as follows: The left-hand side defines the probability δ, that the estimated probability $\hat{P}\left(\eta\right)$ differs by more than

Table 7.1 Simulation parameter values for evaluating the localisation uncertainty. The simulated trajectory represents an idealised straight road and is defined by $(\mathbf{t}_k, \theta_k)$ for the time steps $k = 1, \ldots, 1 \times 10^4$. Note that the disparity noise variance σ_d^2 is left unspecified since this is the parameter which is to be obtained.

Variable	Value
Trajectory	$\mathbf{t}_k = \begin{bmatrix} x_k & -1.275\,\mathrm{m} \end{bmatrix}^\mathrm{T}$, $x_k = 0\,\mathrm{m}, 1\,\mathrm{m}, \ldots 10\,\mathrm{km}$, $\theta_k = 0^\circ\ \forall k$
Landmark distribution (7.1)	$\lambda_\mathrm{LM} = 0.125\,\mathrm{m}^{-1}$, $\mu_{m_y} = \{-1.275\,\mathrm{m}, 6.775\,\mathrm{m}\}$, $\sigma_y = 0.5\,\mathrm{m}$
Measurement model (7.2)-(7.3)	$r_\mathrm{sens,max} = 40\,\mathrm{m}$, $\alpha_\mathrm{sens,max} = 80^\circ$ $c_\mathrm{k} b_\mathrm{w} = 121\,\mathrm{m \cdot pel}$, $\sigma_u = 5\,\mathrm{pel}$
Localisation accuracy requirements (7.8)	$\bar{\sigma}_x = \bar{\sigma}_y = 0.1\,\mathrm{m}$, $1 - \alpha = 0.9$, $1 - \delta = 0.9$

ϵ from the true value $P(\eta)$. Thus, δ can be regarded as a pre-defined confidence level. Then, (7.10) can be reformulated to obtain either the number of samples N_s or the bound ϵ of the deviation as a function of the other. For a given number of iterations N_s, the estimation error ϵ is bounded with a probability of $1 - \delta$ as follows:

$$\epsilon \leq \frac{1}{\sqrt{N_\mathrm{s}}} \log\left(\frac{2}{\delta}\right) . \tag{7.11}$$

It needs to be remarked that if a vehicle is driving along a specific trajectory, the sets of visible landmarks are correlated over time. Effectively, the correlation depends on the overlap of the sensor's field of view in subsequent time-steps. For simplicity, this correlation is not explicitly included in the models from Sec. 7.1.3. Thus, the landmark samples $\mathcal{M}^{(e)}$ in the numerical evaluation of (7.8) can be drawn independently. However, the obtained results only reflect the average behaviour on very long trajectories where the sensor's field of view is small compared to the dimensions of the world.

7.1.5 Numerical example

After the theoretical framework for a model-based derivation of sensor accuracy requirements has been introduced, its application is now demonstrated for a specific scenario and system configuration. To this end, the parameter values from Tab. 7.1 are assumed for the landmark distribution and the stereo vision sensor. The goal is to find a suitable value for the disparity measurement variance σ_d^2 to achieve a localisation accuracy of $\bar{\sigma}_x = \bar{\sigma}_y = 0.1\,\mathrm{m}$ in $1 - \alpha = 90\,\%$ of all cases.

To allow a simplified parametrisation, an ideally straight road is assumed where landmarks are placed at the road boundaries. A short part of the overall trajectory is visualised in Fig. 7.3.

In order to obtain the sensor parametrisation, (7.8) is numerically calculated by sampling of $N_\mathrm{s} = 1 \times 10^4$ landmark configurations. The distribution of the diagonal entries

$\Sigma^0_{\mathbf{t},11}$ and $\Sigma^0_{\mathbf{t},22}$ of the covariance is shown in Fig. 7.4(a). The maximum of these two components divided by the accuracy requirements yields the random variable η from (7.9). Its distribution is displayed in Fig. 7.4(b). According to (7.11), the expected deviation to the true probabilities is smaller than $\epsilon = 0.03$ with a probability of $1 - \delta = 0.9$. The value of η at which the cumulative distribution equals the desired $1 - \alpha$ can be used to find the coefficient ξ_{SV}^{-1} of the sensor covariance matrix in (7.3).

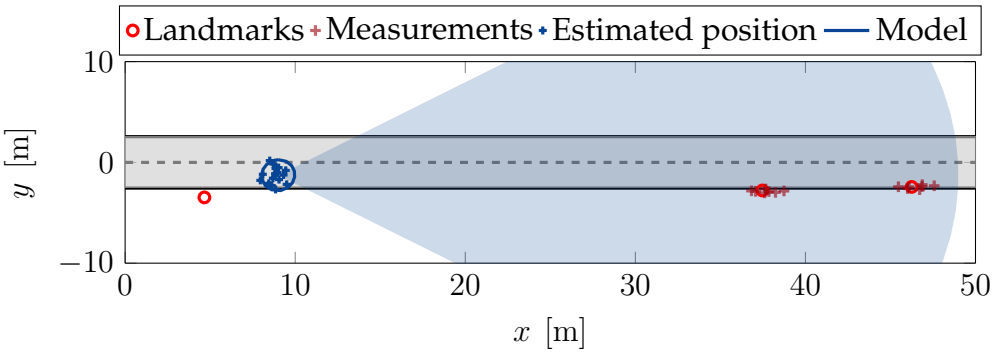
The parameter σ_d^2 of the stereo camera is proportional to the coefficient ξ_{SV}. Therefore, the eventual result is $\sigma_d^2 = 0.632 \cdot 0.01\mathrm{pel}^2$. It has to be stressed that the entire derivation only requires evaluating the analytical expression of the model N_{s} times and to compute the cumulative distribution once.

A Monte-Carlo simulation with $N_{\mathrm{sim}} = 1 \times 10^3$ iterations is now performed in order to validate the obtained value of σ_d^2. Thereby, the map matching problem is repeatedly solved along a 10 km long trajectory with a step size of 1 m. Noisy landmark measurements are drawn using the exact model of the sensor noise covariance from (7.3) whereas the analytical parameter derivation relies on the approximate form. The localisation accuracy achieved in the simulations is shown in Fig. 7.5. The longitudinal and lateral accuracy requirements $\bar{\sigma}_x$ and $\bar{\sigma}_y$ are individually met in 96.5 % and 88.1 % of the time and jointly for 88 %. These results are close to the demanded $1 - \alpha = 90\,\%$ and therefore confirm the derived sensor parameter value.

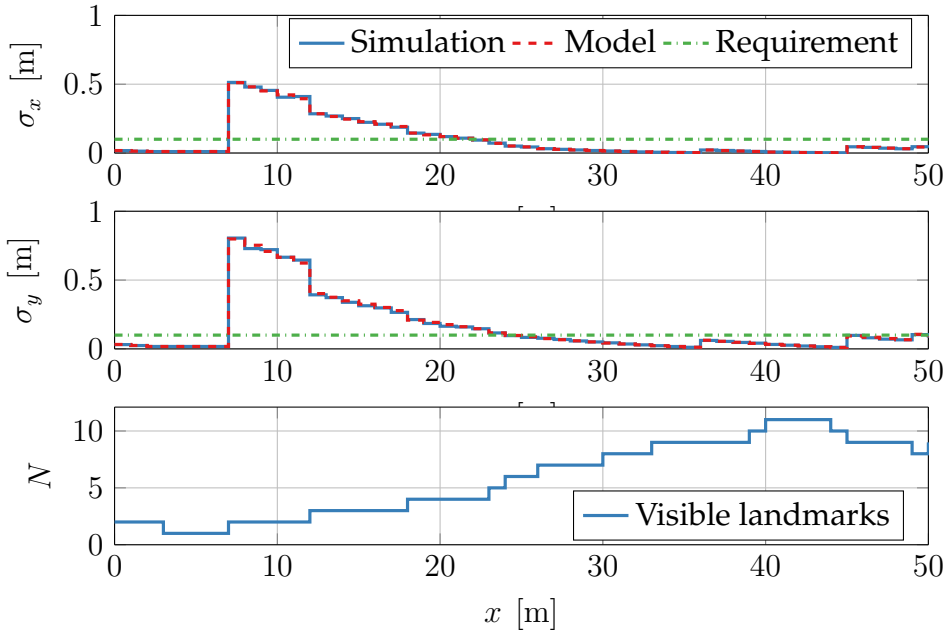
7.1.6 Summary and outlook

This section has outlined a model-based method for deriving sensor accuracy requirements for landmark-based vehicle localisation. Maximum admissible values on the pose estimation uncertainty are defined on the level of the map matching task. These are compared to the uncertainty model from Sec. 3.2 and the analytical expressions are solved for the sensor parameters. This procedure has been exemplified for a stereo vision sensor as previously detailed in Sec. 3.1.

Future works could extend this framework to more complex system designs. For example, map matching of landmarks is usually not the only source of information but fused with position estimates from (visual) odometry or satellite navigation [195]. In order to predict the localisation accuracy after information fusion and temporal filtering, the Cramér-Rao bound from Sec. 4.1 can be derived for this problem. The steady-state value of the CRB, which is asymptotically achieved after the filter has converged, gives a best-case performance estimate. First results on this topic are reported by ROHDE *et al.* [212]. Whereas the localisation accuracy from map matching per time step, as has been studied here, provides a conservative performance estimate.

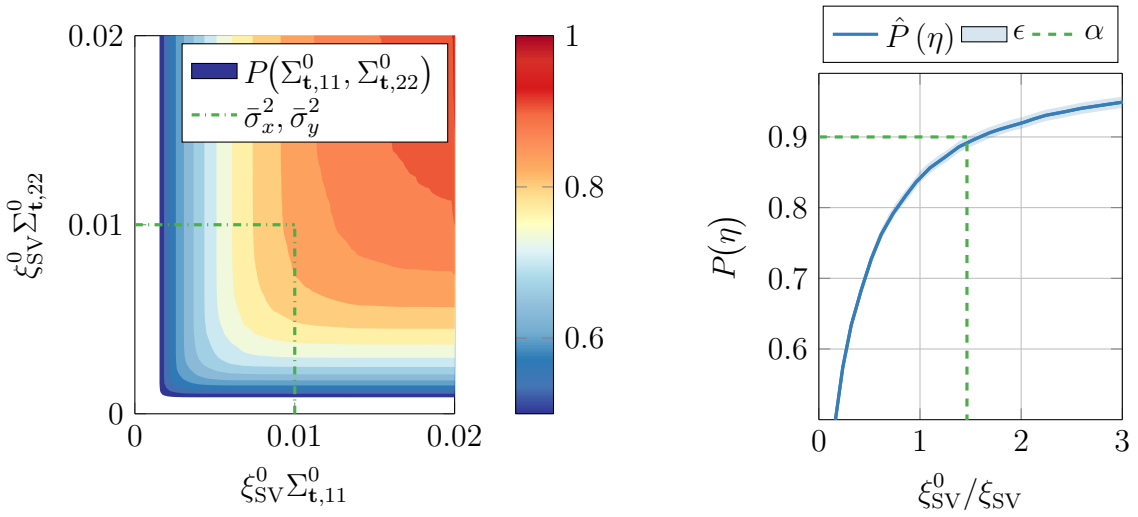


(a) Illustration of the simulation scenario at one time step. Two landmarks (red circles) are in the sensor's field of view. Their location is measured by a stereo vision camera (red crosses), where the anisotropic measurement noise characteristic becomes obvious. Due to this noise, the estimated positions (blue crosses) vary. The covariance of the position estimates is modelled by (7.5b) and is visualised in the form of the 90% confidence ellipse.



(b) The vehicle moves along the road and estimates its position in each simulated time step. Thus, the number of landmarks in the sensor's field of view (bottom) and their distances to the stereo camera are constantly varying. This causes a changing variance of the estimated lateral and longitudinal position (top and middle). The goal of the model-based design is to find sensor noise parameters so that the accuracy requirements (green) are met for a desired proportion $1 - \alpha$ of the time.

Figure 7.3 Illustration of the landmark-based localisation approach (a) and exemplary simulation results (b).



(a) Distribution of $\Sigma^0_{\mathbf{t},11}$ and $\Sigma^0_{\mathbf{t},22}$ for randomly sampled landmark configurations. The dashed lines indicate the desired localisation accuracy which should be met in $1-\alpha = 90\%$ of all cases.

(b) Distribution of η from (7.9). The crossing with $1-\alpha = 0.9$ reveals that $\xi_{\mathrm{SV}} \stackrel{!}{=} 0.632 \cdot \xi^0_{\mathrm{SV}}$ in order to meet the requirements.

Figure 7.4 The dependence of the localisation accuracy on the landmark configuration is obtained by numerically evaluating the analytical model (7.5b) for different landmark configurations according to Sec. 7.1.3.1. A sensor parameter value ξ_{SV} is then obtained as outlined in Sec. 7.1.4. All values are normalised by a pre-defined ξ^0_{SV} which corresponds to $\sigma_d = 0.1\,\mathrm{pel}$.

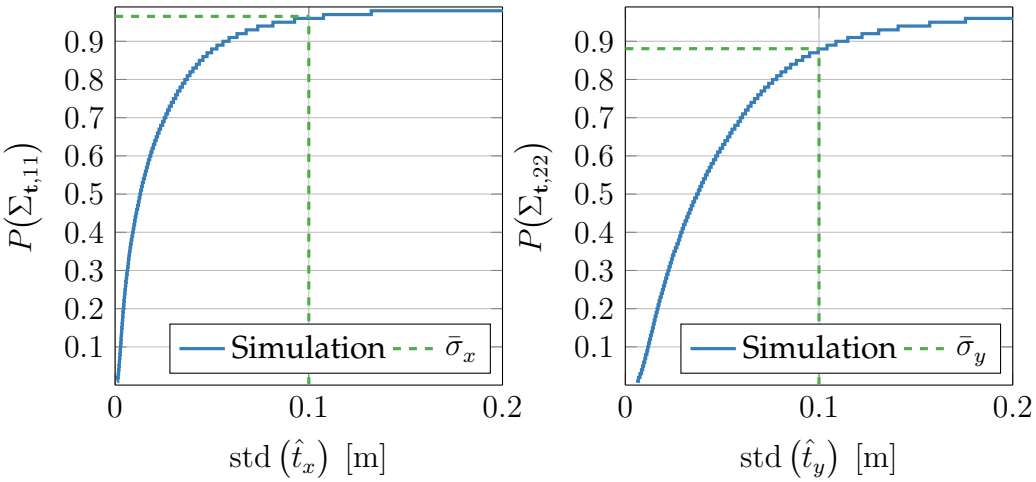


Figure 7.5 Monte-Carlo simulation results of the localisation accuracy. The cumulative frequency of the standard deviation of the longitudinal ($\hat{t}_x$) and lateral ($\hat{t}_y$) position estimates are shown. These results are obtained from a trajectory with a length of $10\,\mathrm{km}$ and $N_{\mathrm{sim}} = 1 \times 10^3$ independent simulations. The requirements $\bar{\sigma}_x$ and $\bar{\sigma}_y^2$ are met in approximately 90% of all cases. This confirms the chosen value of the disparity measurement variance σ_d^2 that was predicted by the model.

7.2 Performance bounds of an autonomous emergency brake

After the previous section has demonstrated how constraints on sensor parameters can be derived from accuracy requirements, it is now investigated how uncertainties in a given system affect its effectiveness. To this end, an exemplary signal processing chain of an autonomous emergency brake (AEB) system is considered. The goal is to analyse how the collision prevention capabilities are affected by uncertainties in measurements, state estimates and predictions.

The considered AEB and the overall approach to its analysis are introduced in Sec. 7.2.1. Differences and similarities to existing works on this topic are subsequently detailed in Sec. 7.2.2. Essentially, the capability of an AEB system to prevent an impending collision depends on the parameters of the situation and the quality of information that is available for the timing of a brake intervention. How these aspects are modelled is described in Sec. 7.2.3. Subsequently, the effect of uncertainties on the AEB performance is analysed in Sec. 7.2.4. The main findings are summarised in the concluding Sec. 7.2.5.

The publication [222] comprises central parts of the following material.

7.2.1 Introduction

On a high level of abstraction, an AEB system is described by the concatenation of an environment sensor for the perception of obstacles (for example a stereo camera), a tracking filter for estimating the motion state (for example an extended Kalman filter) and an algorithm for triggering an emergency brake intervention (for example a criticality measure). This signal processing chain is outlined in the upper part of Fig. 7.6.

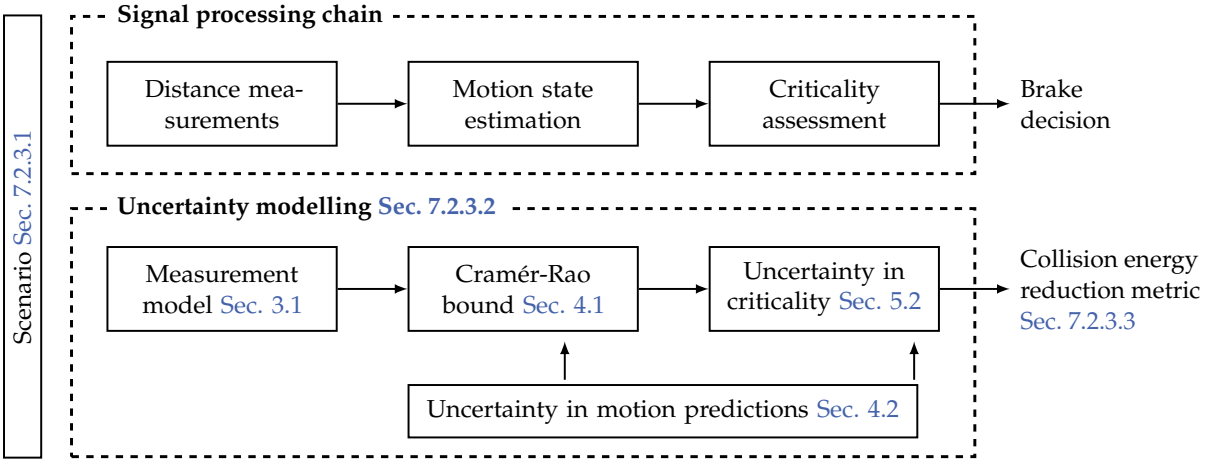


Figure 7.6 Signal processing chain of AEB system (top) and overview of probabilistic models (below). The foundations of these models have been developed in the referenced sections of this thesis.

After probabilistic models of uncertainty in all of these components have been derived in this thesis, this section analyses the propagation in an overall AEB signal processing chain. The central question is how uncertainties compromise the collision avoidance performance and how this varies among different scenarios. Especially upper optimistic bounds on a system's effectiveness are to be derived. Such a study can be useful for many purposes [144], for example the definition of requirements, sensitivity analyses or for adjusting the system parameters.

In order to evaluate the impact of uncertainty on an AEB system, a number of models have to be formulated:

- First, the driving situation under which the system should be analysed needs to be defined. For clarity, only the relation between the ego-vehicle and one preceding object vehicle in longitudinal traffic are considered. In order to obtain more comprehensive results than only an evaluation for particular trajectories, a parametrisable model is introduced to describe entire families of trajectories. An empirical distribution of the initial motion state is retrieved from real-world traffic datasets.
- Second, the AEB system and parametrisation have to be formally specified. A stereo vision sensor is assumed that obtains measurements of the distance to the preceding vehicle. These measurements are affected by distance-dependent noise, as has been discussed in Sec. 3.1. The propagation of the measurement uncertainty to the state estimation layer is modelled using the Cramér-Rao lower bound that has been presented in Sec. 4.1. This yields an optimistic approximation of the estimation covariance of any practical filter. Lastly, for situation assessment, the *brake-threat-number (BTN)* from Sec. 5.2.3 is employed. Using this algorithm guarantees that, under ideal conditions, a collision is avoided by an emergency brake intervention in the last possible moment.
- Third, a method to evaluate the difference between the ideal and uncertain case needs to be defined. To this end, all uncertainty is first propagated to the decision making component, that is the criticality measure. Instead of a deterministic threshold on the criticality, a Bayesian brake activation criterion is employed [173]. Compared to the deterministic baseline, additional time has to pass before the plausibility is sufficiently high and the brake activated. To objectively evaluate this delay, its effect on the future motion of the two vehicles is calculated and subsumed in terms of the relative collision energy.

Before these models are further detailed in Sec. 7.2.3, similar previously published analyses are reviewed in Sec. 7.2.2 and contrasted with the chosen approach.

7.2.2 Related work

Previous related publications can be differentiated by whether numerical or analytical strategies are pursued to evaluate the effects of uncertainty.

Simulation-based studies are presented in [137] for vehicle collision avoidance systems and pedestrian protection in [27, 186]. The latter works compare the expected benefit of different evasive manoeuvres (braking, steering or combined manoeuvres) under uncertain future motion of a pedestrian. Different levels of uncertainty, which could correspond to different sensors, are evaluated in [186]. The probability and severity of pedestrian injuries are used as evaluation metrics. A comprehensive analysis of collision avoidance systems in longitudinal traffic, as is studied in this work, is presented in [137]. The analysis is founded on large amounts of recorded vehicle trajectories. Thus, the results can be aggregated over empirical distributions of the scenario parameters which will be similarly performed in Sec. 7.2.3.1. A drawback shared by these simulation-based works is, as has been discussed in Sec. 1.2, that only numerical results for a specific system can be obtained.

In contrast to the aforementioned works, a rigorous analytical derivation of worst case performance bounds of an AEB system is presented [144]. The study introduces a parametric trajectory model with piece-wise constant accelerations, which is adapted in this work. Moreover, the same criticality measure, the BTN, is assumed for decision making[2]. The central differences between this work and [144] are:

- Errors in the state estimates are modelled with deterministic upper bounds in [144], whereas a fully probabilistic model is used in this work. These probabilistic models are derived by propagating uncertainty in the entire signal processing chain. Therefore, more detailed models can be achieved, for example by taking sensor-specific characteristics of the measurement uncertainty into account.

- Since only upper bounds on the errors are considered in [144], the worst case performance is derived. This work on the other hand studies the opposite case, that are optimistic lower bounds on the error covariances. Thus, the feasibility of a performance requirement as well as minimum accuracy constraints can be evaluated.

- A degradation of the AEB system performance is measured in terms of the activation timing in [144]. Further consequences of such delays on the collision mitigation capabilities are not considered. This work furthermore analyses the effectiveness of the system in terms of reducing the collision energy.

Therefore, albeit the same objective of analytically finding fundamental limitations in AEB systems is pursued, this work employs more detailed models and derives conceptually complementary results compared to [144].

[2] Overall, multiple criticality measures that correspond to different evasive actions in longitudinal and lateral direction are analysed in [144]. These measures are eventually combined by taking the minimum value. All findings of this work concern solely the longitudinal criticality but could be reiterated for additional measures.

7.2.3 Models, parametrisations and evaluation metric

This section describes the foundations of the analysis and comprises three parts. First, a formal representation of the considered scenarios is described in Sec. 7.2.3.1. Second, modelling the uncertainty propagation in the signal processing chain is based on results from previous sections of this work. The essential aspects are recalled and referenced in Sec. 7.2.3.2. Finally, the impact of uncertainties will be evaluated in terms of the collision energy reduction. This physically motivated metric is detailed in Sec. 7.2.3.3.

7.2.3.1 Scenario representation

The primary type of accidents which are addressed by an AEB with a front-facing sensor are collisions with a preceding vehicle. According to [85], these contribute to 30% of all accidents on German roads. In-depth analyses reveal that the cause of approximately two thirds of these accidents is that the driver of a preceding vehicle is suddenly forced to brake unexpectedly strong for some reason. Probably due to inattentiveness and close margins, the driver of a following vehicle is not able to react quickly enough to avoid a collision. In the remaining cases, accidents are caused by a driver who collides with a slower vehicle at constant velocity due to speeding.

Therefore, following the works [137, 144], the longitudinal relative motion between two vehicles that are involved in a rear-end collision is modelled with piece-wise constant accelerations. One advantage of such a compact parametric model is that analytical results can be derived for a wide range of different scenarios. In a top-down approach, challenging regions in the parameter space can be systematically identified.

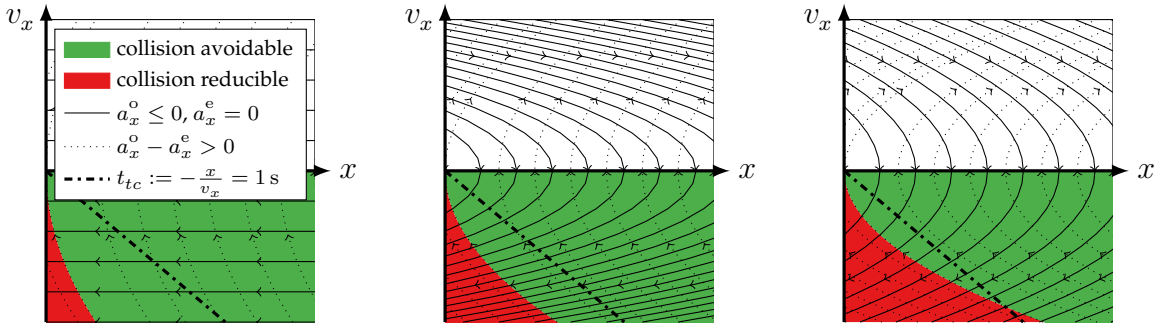
In principle, trajectories of the two considered vehicles could be modelled individually, or, as is pursued here, relative to the ego-vehicle. This choice brings the advantage that fewer variables are needed to describe a scenario. The relevant parameters are the initial distance $x_0 > 0$, the relative speed $v_{x,0} \leq 0$ and accelerations. It is assumed that the preceding vehicle brakes with a constant acceleration $a_x^{\mathrm{o}} \leq 0$ at time $t = 0$. An emergency brake manoeuvre of the ego-vehicle starts at the time of intervention $t_{\mathrm{B}} \geq 0$ with a constant acceleration $a_x^{\mathrm{e}} \leq 0$. This model is described by the following equations of motion:

$$x(t) = \begin{cases} x_0 + v_{x,0} t + \frac{1}{2} a_x^{\mathrm{o}} t^2 & t < t_{\mathrm{B}} \\ x_0 + v_{x,0} t + \frac{1}{2} \left(a_x^{\mathrm{o}} - a_x^{\mathrm{e}}\right) t^2 + a_x^{\mathrm{e}} t_{\mathrm{B}} t - \frac{1}{2} a_x^{\mathrm{e}} t_{\mathrm{B}}^2 & t \geq t_{\mathrm{B}} \end{cases} \tag{7.12a}$$

$$v_x(t) = \begin{cases} v_{x,0} + a_x^{\mathrm{o}} t & t < t_{\mathrm{B}} \\ v_{x,0} + a_x^{\mathrm{o}} t_{\mathrm{B}} + \left(a_x^{\mathrm{o}} - a_x^{\mathrm{e}}\right)\left(t - t_{\mathrm{B}}\right) & t \geq t_{\mathrm{B}} \end{cases} \tag{7.12b}$$

$$a_x(t) = \begin{cases} a_x^{\mathrm{o}} & t < t_{\mathrm{B}} \\ a_x^{\mathrm{o}} - a_x^{\mathrm{e}} & t \geq t_{\mathrm{B}} \end{cases} . \tag{7.12c}$$

A potential inaccuracy of this model occurs if a vehicle has decelerated until standstill and remains fully stopped thereafter. When solely the relative velocity and acceleration are included in the model, this effect is concealed since it depends on the absolute



(a) $a_x^{\mathrm{o}} = 0\,\mathrm{m/s^2}$ **(b)** $a_x^{\mathrm{o}} = -2\,\mathrm{m/s^2}$ **(c)** $a_x^{\mathrm{o}} = -3\,\mathrm{m/s^2}$

Figure 7.7 Trajectories and region of avoidable collisions for different object accelerations a_x^{o} in terms of distance x and relative velocity v_x. Solid lines show the trajectory prior to an emergency brake intervention and dotted lines with the emergency brake ($a_x^{\mathrm{e}} = -6\,\mathrm{m/s^2}$) applied. Green areas in the phase plane denote states where a collision is still avoidable whereas its effect is only reducible in the red areas. The boundary between the two areas is described by (7.13).

velocities. In principle, all derivations can be performed in a similar manner as presented in the following but taking the additional special case of a standing vehicle into account.

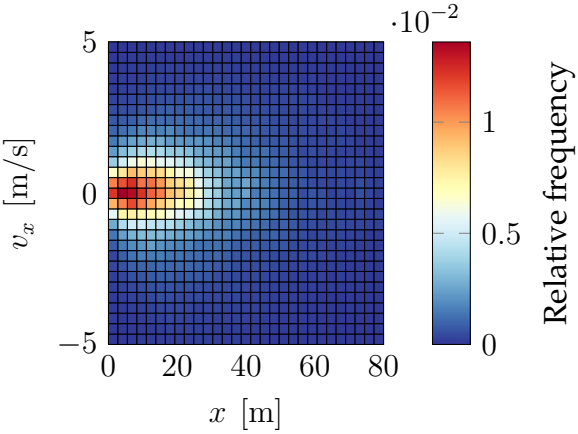
Multiple trajectories can be visualised at once in a phase portrait as shown in Fig. 7.7. Three cases that correspond to different values of the deceleration a_x^{o} are depicted. This includes the special case of an initially unaccelerated motion ($a_x^{\mathrm{o}} = 0$). An emergency brake intervention changes the direction of the trajectories. If a trajectory does not cross the v_x-axis, a collision is avoided. This applies to all trajectories which start in the green area of the phase plane. The subsets of initial states that lead to avoidable or non-avoidable collisions are separated by the boundary

$$2x_0\,(a_x^{\mathrm{o}} - a_x^{\mathrm{e}}) - v_{x,0}^2 = 0\ . \tag{7.13}$$

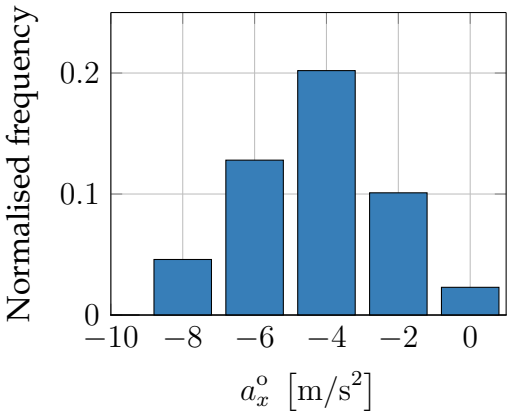
Furthermore, it can be seen that trajectories may initially start at different points in the phase plane in Fig. 7.7 but eventually lie on the same trajectory. In order to find a more compact scenario description, it is proposed in [144] to form subsets of all trajectories with a common end state. However, such a simplified representation is not feasible if effects that depend on the state have to be taken into account. The state dependence of the uncertainty models will be discussed in Sec. 7.2.3.2. Therefore, trajectories are considered individually for each triple of initial values $\left(x_0, v_{x,0}, a_x^{\mathrm{o}}\right)$ in this work.

As is seen from the previous considerations, the capability of an AEB system to prevent an accident is linked to the initial motion parameters. In order to obtain aggregate results, the occurrence of initial values may be weighted by their relevance to real traffic situations. To find representative weights, the following data sources[3] are used here:

[3]Note that it is assumed that the car-following behaviour during traffic without accidents is the same as in the accident case *before* the preceding vehicle brakes unexpectedly and causes a collision. Thus, a much larger data base can be used which yields a finer discretisation of the distribution of $\left(x_0, v_{x,0}\right)$.



(a) Distance x and relative velocity v_x to leading vehicle in the same lane.



(b) Mean deceleration of leading vehicle in rear-end collisions according to [137].

Figure 7.8 Empirical distributions of scenario parameters: (a) Initial relative motion state $\mathbf{x}_0$, (b) deceleration a_x^{o} of preceding vehicle.

- At the beginning of each scenario, the relative motion belongs to normal car-following behaviour. Therefore, the dataset that was previously presented in Sec. 4.2.5.2 can be analysed for the frequency in which combinations of distance and relative velocity $\left(x_0, v_{x,0}\right)$ occur. This distribution is visualised in Fig. 7.8(a).

- In order to model the acceleration a_x^{o} of the preceding vehicle, which eventually causes a collision, real accidents have to be analysed. In-depth analyses are for example provided by the German In-Depth Accident Study (GIDAS). It is not in the scope of this work to perform a detailed simulation of each individual recorded accident, since they differ by a multitude of situation-specific factors. Instead, only the mean deceleration of the preceding vehicle prior to the collision is considered. Values for 111 relevant cases have been identified in [137] and are used in the following.[4] The distribution is shown in Fig. 7.8(b).

In conclusion, traffic scenarios with an impending rear-end collision are concisely parametrised by the initial distance and relative velocity as well as a constant deceleration of the preceding vehicle. Distributions of the state combinations are obtained from empirical data.

7.2.3.2 Modelling of uncertainty in an AEB system

As is indicated in Fig. 7.6, the origin and propagation of uncertainties in the AEB system can be divided into four major aspects which will be outlined in the following.

[4] It needs to be remarked that only accidents with injuries are contained in the dataset. Collisions with minor damage, which might not be reported to authorities in general, are thus under-represented.

Stereo vision distance measurements Measurements of the distance to a preceding vehicle are modelled in disparity coordinates and transformed to Cartesian coordinates at the state estimation layer. Uncertainty in the measured disparity is modelled as additive Gaussian noise with variance σ_d^2.

The variance σ_d^2 depends on the accuracy of the entire disparity image, the size of the vehicle in this image and the algorithm that is used to aggregate the information. In the following, a simplified algorithm from Sec. 3.1 which aggregates the row-wise disparity values $\bar{d}$ over all image rows is assumed. Errors in the individual rows are supposed to be independently distributed with a variance of $\sigma_{e,\bar{d}}^2$.

The overall measurement variance σ_d^2 is inversely proportional to the number of image rows $n_{\Delta v,\mathrm{obj}}$ and thus increases with distance. Depending on the distance x and the object height h_{obj}, this number can be calculated according to (3.13b). The influence of an object's contour on a particular algorithm for disparity aggregation is not explicitly modelled but the lower bound that is derived in (3.23) is employed. Thus, the following expression for the disparity error variance of measurements of a vehicle is obtained:

$$\sigma_d^2 = \frac{\sigma_{e,\bar{d}}^2}{n_{\Delta v,\mathrm{obj}}} = \frac{x}{c_{\mathrm{k}} h_{\mathrm{obj}}} \sigma_{e,\bar{d}}^2 \,. \tag{7.14}$$

Motion state estimation The motion state is written as $\mathbf{x} \coloneqq \begin{bmatrix} x & v_x & a_x \end{bmatrix}^{\mathrm{T}}$. A state estimate $\hat{\mathbf{x}}_k$ can be calculated from disparity measurements that arrive at discrete points in time t_k. The covariance matrix of the state estimation error $\boldsymbol{\Sigma}_{\mathbf{x}_k} \coloneqq \mathrm{cov}\left(\hat{\mathbf{x}}_k - \mathbf{x}_k\right)$ is bounded from below by the inverse information matrix $\boldsymbol{\mathcal{I}}_{\mathbf{x}_k}^{-1} \preceq \boldsymbol{\Sigma}_{\mathbf{x}_k}$ as detailed in Sec. 4.1.

Attention has to be paid to the discrete-time nature of the system. The sampling time T_{S} determines how frequently new sensor measurements arrive at the state estimator. This in turn influences the accuracy of a state estimate $\boldsymbol{\Sigma}_{\mathbf{x}_k}$, the certainty of the criticality and thus the time at which a brake intervention will be triggered.

However, it is out of the scope of this work to analyse the effects of unsynchronised sampling in general. Therefore, a simplified approach is pursued here: Firstly, the expected value of the state estimates are treated as variables in continuous time $\hat{\mathbf{x}}_k \to \hat{\mathbf{x}}\left(t\right)$ which corresponds to $T_{\mathrm{S}} \to 0$. Secondly, the corresponding covariance matrix is defined as $\boldsymbol{\Sigma}_{\mathbf{x}_k}\left(t\right) \coloneqq \boldsymbol{\mathcal{I}}_{\mathbf{x}_{k^*}}^{-1}$ calculated from the discrete time recursion (4.22) until $k^* = \lceil \frac{t}{T_{\mathrm{S}}} - 1 \rceil$.

In general, the time evolution of the Cramér-Rao bound is state-dependent and thus needs to be calculated individually for each trajectory. This prevents to analyse entire subsets of trajectories with common end states. As has been previously discussed in Sec. 4.1.5.2, neither a state-independent nor asymptotic limit of the CRB exists due to the non-linear measurement model of the stereo camera.

Going beyond the scope of this work, special cases in which the bound becomes independent of the state could be investigated. As has been seen in Sec. 4.1.4.4, this applies for example to linear systems with additive, time-invariant noise processes. One could then calculate the asymptotic limit of the bound and thus achieves a very compact model.

Motion state prediction Implicitly, a constant acceleration model is employed in state estimation and criticality assessment to predict the relative vehicle motion. A Gaussian process noise model as detailed in Sec. 4.2 describes the uncertainty in these trajectory predictions. Since the relative motion between two vehicles is of interest, one has to consider uncertainty in both future trajectories of the ego-vehicle and the other vehicle. In contrast to the general case as discussed in Sec. 4.2.7, the transformation from absolute to relative coordinates simplifies if solely the longitudinal motion is analysed. Under the assumption of statistical independence, the process noise power spectral density S_x is twice the value which was estimated for a single vehicle in Sec. 4.2.5.

Criticality measures The propagation of uncertainty in the state estimates and trajectory predictions to algorithms for criticality estimation has been discussed in Sec. 5.2. The specific criticality measure which is considered for the AEB activation, the brake-threat-number (BTN), has been formulated in example 5.4. This criticality measure quantifies the risk of a situation in terms of the instantaneous deceleration $a_{x,\mathrm{req}}^{\mathrm{e}}$ that is required to avoid a collision by braking of the ego-vehicle. Given the relative motion state $\mathbf{x}$, the criticality $\kappa(\mathbf{x})$ is defined by (5.50):

$$\kappa(\mathbf{x}) = a_x - \frac{v_x^2}{2x} \,. \tag{7.15}$$

An AEB activation is triggered if

$$\kappa(\mathbf{x}(t)) \overset{!}{\leq} \kappa_0 \tag{7.16}$$

where $\kappa_0 < 0$ denotes a threshold on the negative acceleration.

Uncertainty in this criticality estimate is modelled with a zero mean Gaussian distribution $\kappa(\mathbf{x}) \sim \mathcal{N}\left(\mu_\kappa, \sigma_\kappa^2\right)$. Given a state estimate $\mathbf{x} \sim \mathcal{N}\left(\hat{\mathbf{x}}, \mathbf{\Sigma}_{\mathbf{x}_k}\right)$ and the process noise spectral density S_x in the underlying CA motion model, the uncertainty is propagated to $\kappa(\mathbf{x})$ from (7.15) according to (5.62):

$$\mu_\kappa = \kappa(\hat{\mathbf{x}}) \,, \tag{7.17a}$$

$$\sigma_\kappa^2 = \left(\nabla_{\mathbf{x}}\kappa(\mathbf{x})|_{\hat{\mathbf{x}}}\right) \cdot \mathbf{\Sigma}_{\mathbf{x}_k} \cdot \left(\nabla_{\mathbf{x}}\kappa(\mathbf{x})|_{\hat{\mathbf{x}}}\right)^{\mathrm{T}} - \frac{2\hat{x}}{5\hat{v}_x} S_x \,, \tag{7.17b}$$

$$\text{with} \quad \nabla_{\mathbf{x}}\kappa(\mathbf{x})|_{\hat{\mathbf{x}}} = \begin{bmatrix} \frac{\hat{v}_x^2}{2\hat{x}^2} & -\frac{\hat{v}_x}{\hat{x}} & 1 \end{bmatrix} \,.$$

A probabilistic generalisation of the brake activation condition (7.16) to a system that is

subject to known uncertainties[5] reads

$$\mathrm{P}\left(\kappa\left(\mathbf{x}\left(t\right)\right) \leq \kappa_0\right) \overset{!}{\geq} 1-\alpha \iff \int_{-\infty}^{\kappa_0} \mathcal{N}\left(\kappa; \mu_\kappa\left(t\right), \sigma_\kappa^2\left(t\right)\right) \, \mathrm{d}\kappa \overset{!}{\geq} 1-\alpha \tag{7.18}$$

with a predefined confidence level $1-\alpha$ [173]. Given the previously detailed models, this probabilistic condition and the activation time for the scenarios from Sec. 7.2.3.1 will be analysed in Sec. 7.2.4.2.

7.2.3.3 Collision energy reduction metric

This section details a metric that is used to assess the influence of uncertainties on the AEB system. One could consider the time of activation and compare the nominal with the disturbed case, as in [144]. However, this metric does not measure the effectiveness of the collision avoidance system. Another approach is thus to evaluate the influence that the system would have on a hypothetical collision in terms of physical quantities, for example the impact velocity. More sophisticated approaches even consider the consequences of the collision on the safety of human passengers [137].

One relatively simple approach that is applied in [100] is to calculate the kinetic impact energy $E_\mathrm{coll} = \frac{1}{2} m v_\mathrm{coll}^2$ of an inelastic collision with the relative collision velocity v_coll and vehicle mass m. A measure of effectiveness is then obtained from the difference between the case with ($E_\mathrm{coll,B}$) and without brake intervention (E_coll). Normalising this difference by E_coll yields the dimensionless relative collision energy reduction $\Delta E \in [0,1]$:

$$\Delta E = \frac{E_\mathrm{coll} - E_\mathrm{coll,B}}{E_\mathrm{coll}} = \frac{\frac{1}{2} m v_\mathrm{coll}^2 - \frac{1}{2} m v_\mathrm{coll,B}^2}{\frac{1}{2} m v_\mathrm{coll}^2} = 1 - \left(1 - \frac{v_\mathrm{coll} - v_\mathrm{coll,B}}{v_\mathrm{coll}}\right)^2 . \tag{7.19}$$

The underlying assumptions – only the first contact between the vehicles is considered and modelled with an inelastic collision – limit the accuracy of this model. Nevertheless, the simplicity of (7.19) is appealing and allows the development of explicit expressions.

ΔE depends solely on the velocity at the time of collision which can be obtained in closed form for the parametric trajectory model from Sec. 7.2.3.1. For the case without brake intervention ($a_x^\mathrm{e} = 0$), v_coll follows immediately from (7.12b):

$$v_\mathrm{coll} = v_{x,0} + a_x^\mathrm{o} t_\mathrm{coll} \, , \tag{7.20a}$$

where the time of collision t_coll reads

$$t_\mathrm{coll} = \begin{cases} -\frac{x_0}{v_{x,0}} & a_x^\mathrm{o} = 0 \\ -\frac{v_{x,0}}{a_x^\mathrm{o}} + \sqrt{\frac{v_{x,0}^2 - 2 x_0 a_x^\mathrm{o}}{\left(a_x^\mathrm{o}\right)^2}} & a_x^\mathrm{o} < 0 \end{cases} . \tag{7.20b}$$

[5] Note that this criterion leads to a different approach than the analysis of the decision timing in Sec. 5.2.4.3. There, a deterministic criticality measure and decision making function are assumed. It is studied how random errors in this criticality affect the time instant at which the threshold is exceeded first. In this section, on the other hand, a fully probabilistic criticality measure is assumed where all uncertainty is taken into account. Therefore, the time of activation of a Bayesian approach to decision making is analysed.

If an emergency brake manoeuvre with a deceleration $a_x^{\mathrm{e}} < 0$ is initiated at the time $0 \leq t_{\mathrm{B}} \leq t_{\mathrm{coll}}$, one obtains $v_{\mathrm{coll,B}}$ as a function of t_{B}:

$$v_{\mathrm{coll,B}} = v_{x,0} + a_x^{\mathrm{o}} t_{\mathrm{coll,B}} - a_x^{\mathrm{e}} \left(t_{\mathrm{coll,B}} - t_{\mathrm{B}} \right) \ , \tag{7.21a}$$

where the time of collision under braking $t_{\mathrm{coll,B}}$ is obtained from (7.12a):

$$t_{\mathrm{coll,B}} = \begin{cases} -\frac{x_0 - \frac{1}{2} a_x^{\mathrm{o}} t_{\mathrm{B}}^2}{v_{x,0} + a_x^{\mathrm{o}} t_{\mathrm{B}}} & a_x^{\mathrm{o}} = a_x^{\mathrm{e}} \\ -\frac{v_{x,0} + a_x^{\mathrm{e}} t_{\mathrm{B}}}{a_x^{\mathrm{o}} - a_x^{\mathrm{e}}} + \sqrt{\frac{v_{x,0}^2 - 2x_0 \left(a_x^{\mathrm{o}} - a_x^{\mathrm{e}} \right) + 2 v_{x,0} a_x^{\mathrm{e}} t_{\mathrm{B}} + a_x^{\mathrm{e}} a_x^{\mathrm{o}} t_{\mathrm{B}}^2}{\left(a_x^{\mathrm{o}} - a_x^{\mathrm{e}} \right)^2}} & a_x^{\mathrm{o}} \neq a_x^{\mathrm{e}} \end{cases} . \tag{7.21b}$$

Note that a collision only occurs if $t_{\mathrm{coll,B}}$ is non-negative and real which requires:

$$\begin{cases} v_{x,0} + a_x^{\mathrm{o}} t_{\mathrm{B}} \leq 0 & a_x^{\mathrm{o}} = a_x^{\mathrm{e}} \\ v_{x,0}^2 - 2x_0 \left(a_x^{\mathrm{o}} - a_x^{\mathrm{e}} \right) + 2 v_{x,0} a_x^{\mathrm{e}} t_{\mathrm{B}} + a_x^{\mathrm{e}} a_x^{\mathrm{o}} t_{\mathrm{B}}^2 \geq 0 & a_x^{\mathrm{o}} \neq a_x^{\mathrm{e}} \end{cases} . \tag{7.22}$$

Otherwise, one has $E_{\mathrm{coll,B}} = 0$ and $\Delta E = 1$. Inserting the intermediate results (7.20) and (7.21) into the definition (7.19) yields the following expression for ΔE:

$$\Delta E = \begin{cases} a_x^{\mathrm{e}} \frac{t_{\mathrm{B}}^2 a_x^{\mathrm{o}} + 2 v_{x,0} t_{\mathrm{B}} + 2x_0}{2 a_x^{\mathrm{o}} x_0 - v_{x,0}^2} & \frac{2x_0 a_x^{\mathrm{o}} - v_{x,0}^2}{a_x^{\mathrm{o}} t_{\mathrm{B}}^2 + 2 v_{x,0} t_{\mathrm{B}} + 2x_0} \leq a_x^{\mathrm{e}} \leq 0 \\ 1 & a_x^{\mathrm{e}} < \frac{2x_0 a_x^{\mathrm{o}} - v_{x,0}^2}{a_x^{\mathrm{o}} t_{\mathrm{B}}^2 + 2 v_{x,0} t_{\mathrm{B}} + 2x_0} \end{cases} . \tag{7.23}$$

The relative collision energy reduction is directly proportional to the strength of the emergency brake manoeuvre a_x^{e}. However, its dependence on the time of braking t_{B} as well as the initial motion state $\mathbf{x}_0$ is non-linear. This will be further analysed in the following section, where t_{B} is derived for the BTN criticality measure.

7.2.4 Effect of uncertainties on AEB brake interventions

This section builds on the previously introduced models and assesses the influence of uncertainties on the effectiveness of an AEB system. The analysis consists of three parts: First, the ideal case without uncertainty is studied in Sec. 7.2.4.1. Second, uncertainties are taken into account in Sec. 7.2.4.2 and their impact is assessed in comparison to the baseline case. Third, empirically determined weights are introduced for the scenario parameters, over which all analyses are conducted, in Sec. 7.2.4.3. Thus, an aggregate analysis of the real-world impact is obtained.

Numerical results are derived for a specific exemplary system which is characterised by the parameter values given in Tab. 7.2. Nevertheless, the interchangeable nature of the models and explicit analytical results facilitate extending the analysis to other systems, for example with a different sensor.

Table 7.2 Model parameter values of numerical example.

Variable	Value
Measurement model (7.14):	$c_{\mathrm{k}} b_{\mathrm{w}} = 121\,\mathrm{m}\cdot\mathrm{pel}\,,\quad \sigma^2_{e,\bar{d}} = 0.01\,\mathrm{pel}^2\,,\qquad h_{\mathrm{obj}} = 1.5\,\mathrm{m}$
Initial state covariance:	$\mathbf{\Sigma}_{\mathbf{x}_0} = \mathrm{diag}\left(100\,\mathrm{m}^2, 25\,\mathrm{m}^2/\mathrm{s}^2, 4\,\mathrm{m}^2/\mathrm{s}^4\right)$
Sampling time:	$T_{\mathrm{S}} = 0.0675\,\mathrm{s}$
Process noise:	$S_x = 2\cdot 0.261\,\mathrm{m}^2/\mathrm{s}^6\mathrm{s}^{-1}$
Confidence level	$1-\alpha = 90\%$
Ego-vehicle brake deceleration	$a_x^{\mathrm{e}} = -6\,\mathrm{m/s}^2$
BTN activation threshold	$\kappa_0 = a_x^{\mathrm{e}} = -6\,\mathrm{m/s}^2$

7.2.4.1 Ideal time of activation and collision energy reduction

Firstly, the time of braking t_{B} is calculated for the BTN (7.15). Inserting the trajectory model (7.12) into the activation condition (7.16) and solving for t yields three different cases for the time of activation t_{B}:

$$t_{\mathrm{B}} = \begin{cases} 0 & \kappa\left(\mathbf{x}_0\right) \leq \kappa_0 \\ \begin{cases} -\frac{x_0}{v_{x,0}} - \frac{v_{x,0}}{2\kappa_0} & a_x^{\mathrm{o}} = 0 \\ -\frac{v_{x,0}}{a_x^{\mathrm{o}}} + \sqrt{\frac{v_{x,0}^2 - 2x_0 a_x^{\mathrm{o}}}{\left(a_x^{\mathrm{o}}\right)^2}\left(1 - \frac{a_x^{\mathrm{o}}}{\kappa_0}\right)} & a_x^{\mathrm{o}} < 0 \end{cases} & \kappa\left(\mathbf{x}_0\right) > \kappa_0 \end{cases}. \tag{7.24}$$

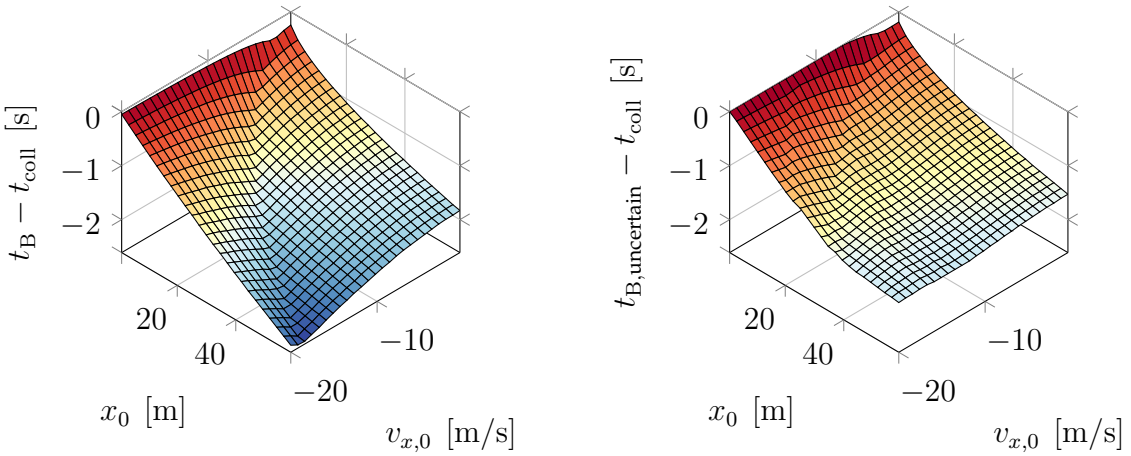
The first case refers to a situation which is already critical for the initial state $\mathbf{x}_0$. Exemplary values of t_{B} are visualised over $\left(x_0, v_{x,0}\right)$ in Fig. 7.9(a). It can be seen that a higher initial relative velocity $v_{x,0}$ causes an earlier activation.

Secondly, t_{B} from (7.24) is inserted into (7.23) which yields the collision energy reduction ΔE:

$$\Delta E = \begin{cases} \begin{cases} a_x^{\mathrm{e}} \frac{2x_0}{2a_x^{\mathrm{o}} x_0 - v_{x,0}^2} & a_x^{\mathrm{o}} - \frac{v_{x,0}^2}{2x_0} \leq a_x^{\mathrm{e}} \leq 0 \\ 1 & a_x^{\mathrm{e}} \leq a_x^{\mathrm{o}} - \frac{v_{x,0}^2}{2x_0} \end{cases} & \kappa\left(\mathbf{x}_0\right) \leq \kappa_0 \\ \begin{cases} \frac{a_x^{\mathrm{e}}}{\kappa_0} & \kappa_0 < a_x^{\mathrm{e}} < 0 \\ 1 & a_x^{\mathrm{e}} \leq \kappa_0 \end{cases} & \kappa\left(\mathbf{x}_0\right) > \kappa_0 \end{cases}. \tag{7.25}$$

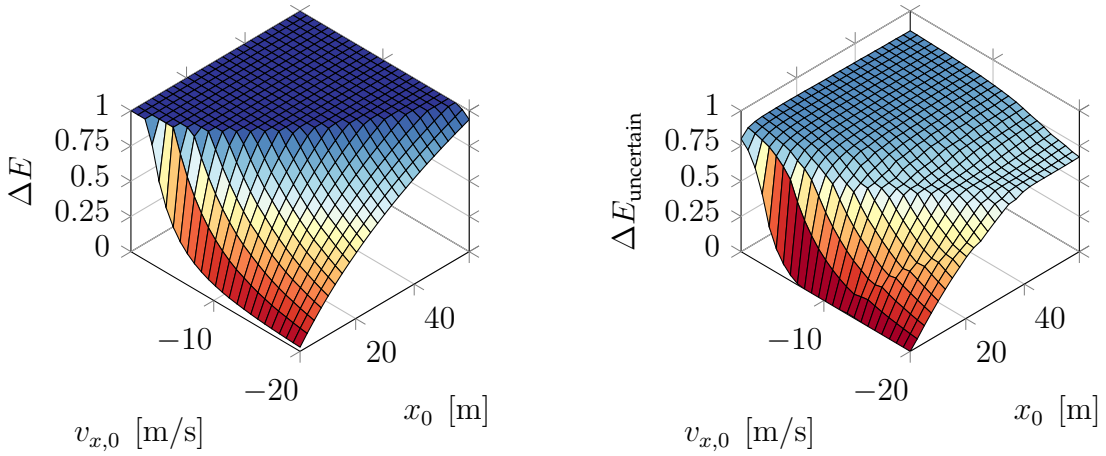
The first two cases refer to an already critical situation ($t_{\mathrm{B}} = 0$). A collision is only fully avoided in the second case which requires that a_x^{e} is sufficiently low. In practice, the braking capability of a vehicle is limited however and therefore, it is not possible to entirely avoid a collision for arbitrary initial states.

If the initial state is not already critical (third and fourth case in (7.25)), the activation timing of the BTN guarantees that the collision will be avoided ($\Delta E = 1$) for a threshold



(a) Deterministic baseline case. **(b)** Under uncertainty.

Figure 7.9 Activation timing of an AEB brake intervention over initial states $(x_0, v_{x,0})$ for $a_x^\mathrm{o} = -3\,\mathrm{m/s}^2$. The baseline case of a system with ideal knowledge of the state is shown in (a). Under the influence of uncertainties, activations are delayed as visualised in (b).



(a) Deterministic baseline case. **(b)** Under uncertainty.

Figure 7.10 Relative reduction of the collision energy $\Delta E_\mathrm{uncertain}$ by AEB brake intervention over initial states $(x_0, v_{x,0})$ for $a_x^\mathrm{o} = -3\,\mathrm{m/s}^2$. The baseline case of a system with ideal knowledge of the state is shown is shown in (a). Results under the influence of uncertainties are visualised in (b).

selection of $\kappa_0 = a_x^\mathrm{e}$. Furthermore, the result for the third case shows that a linear relationship between ΔE and a_x^e and an inversely proportional relation to κ_0 exist. Thus, if the activation threshold κ_0 is decreased to lower negative accelerations, the emergency brake intervention will be delayed and its effectiveness will decrease proportionally.

These results are visualised in Fig. 7.10(a) over combinations of the initial state and $\kappa_0 = a_x^\mathrm{e}$. The boundary of the plateau indicates the region where a collision is already unavoidable for the initial state. This boundary is described by (7.13) and has been previously seen in the phase portrait in Fig. 7.7.

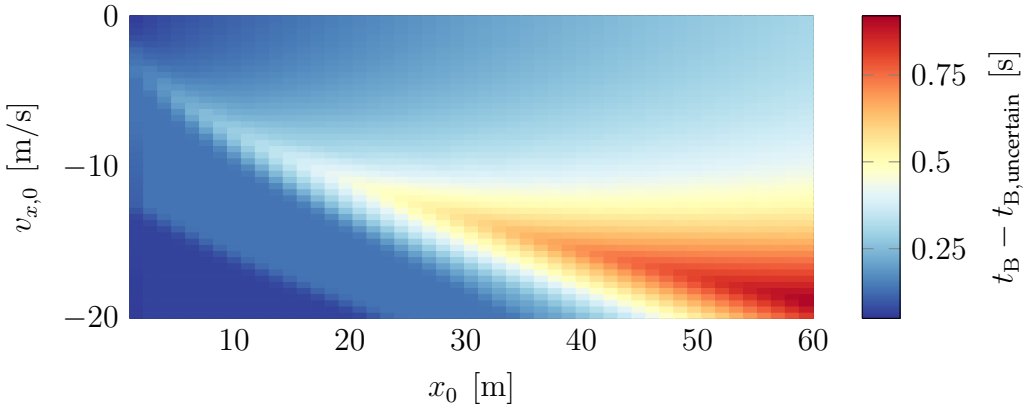


Figure 7.11 Difference in braking time $\Delta t_\mathrm{B} = t_\mathrm{B} - t_\mathrm{B,uncertain}$ over initial state $(x_0, v_{x,0})$ for $a_x^\mathrm{o} = -3\,\mathrm{m/s^2}$. Positive values indicate that triggering the emergency brake intervention is delayed. The worst delays are obtained for trajectories starting close to the parabola which defines the area of unavoidable collision states (see Fig. 7.7). In these cases, only very limited time is available for plausibilisation prior to the designated time of activation. Moreover, the results deteriorate for higher distances where the stereo vision measurements become less accurate.

7.2.4.2 Time of activation and collision energy reduction under uncertainty

In contrast to the deterministic case, the Bayesian activation criterion (7.18) cannot be analytically solved for t in general. It is necessary to resort to numerical methods for calculating the integral and solving for the smallest time $t_\mathrm{B,uncertain}$ for which the criterion is fulfilled.

The results are shown in Fig. 7.9(b). In comparison to Fig. 7.9(a), the activation is in general triggered later than in the deterministic case. Furthermore, the difference $\Delta t_\mathrm{B} := t_\mathrm{B} - t_\mathrm{B,uncertain}$ is visualised in Fig. 7.11 and three distinct effects can be observed:

1. The worst delays occur at the boundary (7.13) between the critical and uncritical region. In these situations, very limited time is available to increase the confidence before the brake intervention would have been already initiated under ideal conditions.

2. Along this boundary, the results deteriorate for increasing distances. This is due to the assumed stereo vision measurement principle whose accuracy decreases with distance.

3. Trajectories starting in the area of unavoidable collisions or sufficiently before are less affected by uncertainties. In the first case, the mean criticality $\mu_\kappa(t)$ is sufficiently high right from the beginning and the activation criterion (7.18) is fulfilled almost immediately. In the second case, sufficient time and sensor measurements are available before an intervention is required. Thus, the uncertainty in the state estimate is minimised and small values of the variance $\sigma_\kappa^2(t)$ are ensured.

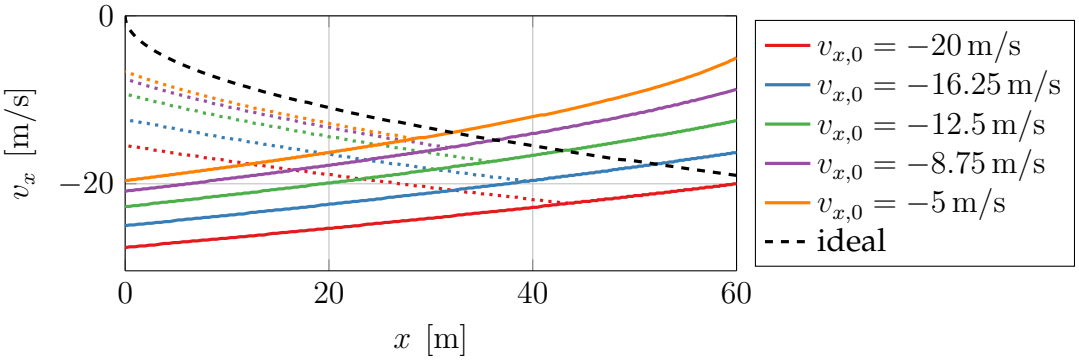


Figure 7.12 Trajectories starting at $x_0 = 60\,\mathrm{m}$ with different relative velocity $v_{x,0}$. Due to uncertainties, the emergency brake intervention (dotted trajectory) is initiated later than in the ideal deterministic case (dashed). The collision energy is proportional to the square of the velocity when crossing the $x = 0$ axis. The braking manoeuvre of the preceding vehicle is defined by $a_x^{\mathrm{o}} = -3\,\mathrm{m/s}^2$.

The practical implications of a delayed activation become visible for exemplary trajectories shown in Fig. 7.12. All trajectories start at the same initial distance $x_0 = 60\,\mathrm{m}$ but with different initial velocity. Dotted lines show the state evolution after a brake manoeuvre is initiated. These activations are delayed due to uncertainties and therefore collisions are mitigated but not entirely avoided.

Although a closed-form result for the difference Δt_{B} between the ideal and uncertain case is not available in general, one can derive the collision energy reduction in terms of Δt_{B}. To this end, (7.19) is calculated for an impact velocity of $v_{\mathrm{coll,B}} + a_x^{\mathrm{e}} \Delta t_{\mathrm{B}}$:

$$\begin{aligned} \Delta E_{\mathrm{uncertain}} &= \Delta E - 2\frac{v_{\mathrm{coll,B}}}{v_{\mathrm{coll}}^2}\left(a_x^{\mathrm{e}}\Delta t_{\mathrm{B}} + \frac{1}{2v_{\mathrm{coll}}}\left(a_x^{\mathrm{e}}\Delta t_{\mathrm{B}}\right)^2\right) \\ &= \Delta E - a_x^{\mathrm{e}}\frac{\Delta t_{\mathrm{B}}^2 a_x^{\mathrm{o}} + 2\Delta t_{\mathrm{B}}\left(t_{\mathrm{B}} a_x^{\mathrm{o}} + v_{x,0}\right)}{v_{x,0}^2 - 2a_x^{\mathrm{o}} x_0}\,. \end{aligned} \tag{7.26}$$

Delaying an AEB activation by Δt_{B} reduces the relative collision energy reduction quadratically. Absolute values of $\Delta E_{\mathrm{uncertain}}$ are visualised in Fig. 7.10(b).

7.2.4.3 Results weighted over scenario parameter distributions

The previously obtained results show that the effectiveness of an AEB system depends on the scenario parameters such as the initial distance between the two vehicles. In order to make a meaningful overall assessment of a system, one has to take into account that not every scenario occurs with the same frequency in real-world traffic. Therefore, the empirically obtained distributions shown in Fig. 7.8 are now used to calculate a weighted average value of the collision energy reduction ΔE.

Furthermore, it has been assumed so far that the other vehicle's deceleration starts at $t = 0$ and that sensor measurements are available from this time on as well. However,

in practice, a sensor's field of view is limited and therefore, the first time when an AEB becomes aware of a possibly threatening situation depends on the initial state $\left(x_0, v_{x,0}\right)$. Since the scenario description only covers the longitudinal movement, solely the longitudinal range of detection $x_{\text{sens,max}}$ of the stereo camera's field of view is taken into account. This is a best case assumption since effects such as limitations of the angular field of view or dynamic occlusions are not considered.

Precisely, a limited field of view negatively affects the AEB in two ways:

1. As seen in Fig. 7.7 there is a certain distance x for each relative velocity v_x below which a collision is not entirely avoidable any more. All situations in which the sensor's field of view does not extend beyond this critical boundary are thus not fully avoidable by the AEB.

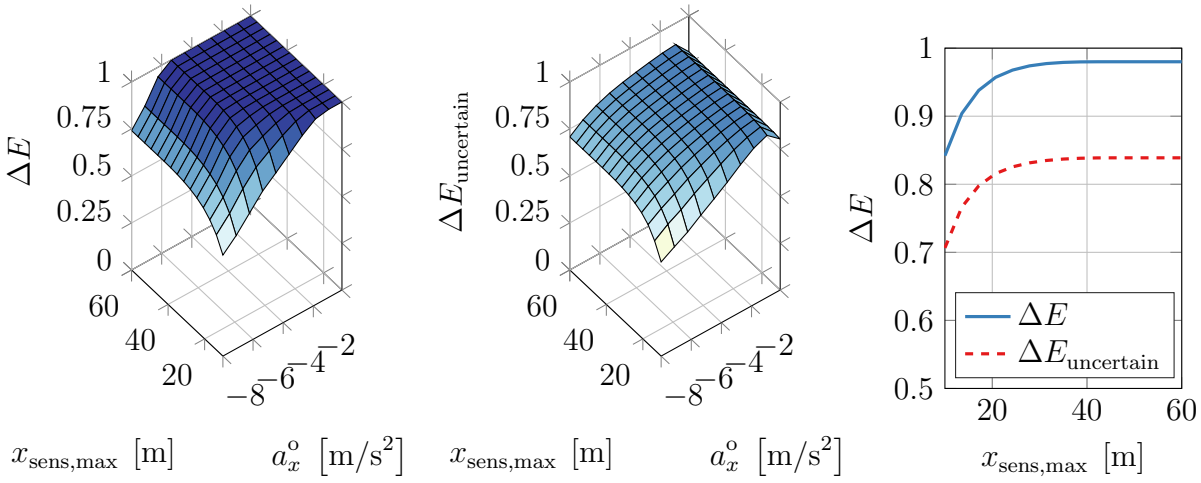
2. As has been discussed in Sec. 7.2.4.2, the time between the first observation and before reaching the critical region in the phase portrait determines the certainty of the criticality estimate. In general, the achievable certainty decreases for fewer measurements, that is a shorter sensor measurement range. Thus, the probabilistic activation criterion (7.18) is reached at a later time which has a negative effect on the collision energy reduction.

When taking all these effects into account, one obtains the results shown in Fig. 7.13. It is first visualised in Fig. 7.13(a)-(b) how the relative reduction of the collision energy ΔE behaves for different combinations of object deceleration a_x^{o} and sensor range $x_{\text{sens,max}}$. Each point is the weighted average of ΔE as previously seen in Fig. 7.10(a)-(b) over the distribution of $\left(x_0, v_{x,0}\right)$ from Fig. 7.8(a). As is expected, increasing the sensor range has a positive effect. Note that even in the baseline case without uncertainties, perfect results cannot be achieved, especially for high absolute values of a_x^{o}. As has been seen in Fig. 7.10(a), a collision is never avoidable for certain combinations of low initial distance and high relative velocity, due to the limited braking capabilities. Nevertheless, the weighted average in Fig. 7.13(a) can reach values close to one even for $a_x^{\text{o}} = -6\,\text{m/s}^2$. This indicates, as is expected, that human drivers maintain a sensible distance to a preceding vehicle most of the time. Situations with a combination of distance and relative velocity that are very sensitive to an unexpected braking of other traffic participants are proactively avoided and only occur with low weights.

If these results are furthermore weighted with the empirical distribution of a_x^{o} from Fig. 7.8(b), one obtains the results shown in Fig. 7.13(c). Under ideal conditions, a sensor range of $x_{\text{sens,max}} = 40\,\text{m}$ suffices to achieve an average collision energy reduction of approximately 95%. This high theoretical effectiveness is likely because the distribution of relative velocities predominantly concentrates around small values $|v_{x,0}| \leq 3\,\text{m/s}$ as seen in Fig. 7.8(a). Certain situations, for example if the ego-vehicle approaches an already stationary vehicle with a high velocity, are not included with high weights. In these cases, increased foresight is necessary to react sufficiently early.

With uncertainties taken into account, the average effectiveness drops by approximately 20%. Thus, although an optimistic model with lower bounds on the state

estimation uncertainty is employed, the effect on the eventual performance of the AEB system is not negligible.



(a) Weighted mean of ΔE over sensor range $x_{\text{sens,max}}$ and object deceleration a_x^{o} in deterministic baseline case.

(b) Weighted mean of $\Delta E_{\text{uncertain}}$ over sensor range $x_{\text{sens,max}}$ and object deceleration a_x^{o} under uncertainty.

(c) Weighted means of ΔE and $\Delta E_{\text{uncertain}}$ over sensor range $x_{\text{sens,max}}$.

Figure 7.13 Relative reduction of the collision energy ΔE weighted by scenario parameter distributions from Fig. 7.8. Weighting with the distribution of $(x_0, v_{x,0})$ from Fig. 7.8(a) yields the results shown in (a)-(b). The weighted means are shown over combinations of $x_{\text{sens,max}}$ and a_x^{o} in the baseline and uncertain case. Furthermore, the weighted mean of these results over the empirical distribution of a_x^{o} from Fig. 7.8(b) is shown in (c). This graph depicts the relation between the effectiveness of an AEB and the sensor detection range $x_{\text{sens,max}}$.

7.2.5 Summary and outlook

This section has analysed the influence of uncertainties on the collision prevention capabilities of an AEB system. The analysis is based on analytical models of uncertainty in algorithms for perception, state estimation and criticality assessment. Results from a numerical example demonstrate the concrete application of the method.

It is found that uncertainties affect the system's ability to perform a timely emergency brake activation depending on the driving scenario. The worst delays are obtained in scenarios which start very close to an unavoidable collision and thus leave only little time for plausibilisation. Fortunately, such critical scenarios are not predominant in real-world traffic, as has been seen from empirical scenario distributions. Therefore, the weighted average results for the exemplary system show a high collision avoidance benefit.

In order to derive the probabilistic model, the signal processing chain of an AEB system has been analysed block-wise. Models of uncertainty in sensor measurements have been propagated to the state estimates and a criticality measure. Finally, the time at which a probabilistic activation criterion is reached, has been calculated. Future works could attempt to derive this timing in a more abstract way without relying on a specific criticality algorithm. This could be achieved in the framework of hypothesis tests as introduced in Sec. 5.1. Given a series of uncertain sensor measurements over time, one has to test the hypothesis that a brake intervention is necessary and make a decision with a pre-defined level of confidence.

8 Conclusion and outlook

The workspace of an intelligent vehicle is in general highly dynamic, is characterised by other traffic participants with often unforeseeable behaviour and can be observed only partially through noisy sensor measurements. The scope of this thesis has been to formulate probabilistic models for these unknown and uncertain aspects. Such models benefit the development of reliable and robust driver assistance systems.

8.1 Conclusion

This work has addressed the question of how uncertainties affect algorithms in an ADAS signal processing chain. To this end, methods for finding parametric models of uncertainties in common processing steps have been developed and evaluated. The developed models of uncertainties can be propagated through a processing chain with the help of analytical approximations. In contrast to previous works – which mainly rely on numerical approaches – analytical solutions are less computationally demanding, open up a path to derive constraints on system parameters and can lead to generic conclusions. These advantages have been demonstrated for the examples of finding sensor accuracy requirements in a feature-based localisation task and the derivation of upper performance bounds on an autonomous emergency brake system.

Finding constraints on a system parameter, for example the accuracy of an environment sensor, answers the question of how the properties of an algorithm's input data relate to quality requirements on the processed output data. To this end, models of the input signals and the propagation to the algorithm's output are required. In this thesis, error models of distance measurements given by a stereo vision sensor have been developed in Sec. 3.1. The propagation of such measurement errors to algorithms for matching of corresponding environment features has been subsequently derived in Sec. 3.2. In combination, these models enable the model-based derivation of minimum sensor accuracy requirements for a map matching based localisation approach which has been detailed in Sec. 7.1.

Although the optimal solution of an estimation problem may be intractable, knowing the theoretical performance bound is of great value. This has been studied for state estimation problems in Sec. 4.1 and the recognition of driving manoeuvres in Sec. 5.1. These analytical bounds can be used to objectively evaluate the performance of a sub-optimal solution. This helps to select appropriate algorithms, determine minimum requirements on a component and to identify fundamental limitations in a system. In this thesis, such limitations have been investigated for an exemplary AEB system in

Sec. 7.2. The influence of uncertain distance measurements (Sec. 3.1), motion state estimates (Sec. 4.1) and state predictions (Sec. 4.2) on the timely activation of an AEB brake intervention has been studied.

The two aforementioned exemplary applications are founded on probabilistic models in different processing steps. The main results covered by these models will be summarised in the following.

Environment perception Two environment perception tasks build the foundations of the applications of emergency brake systems and feature-based localisation.

First, errors in distance measurements that are derived from stereo images have been modelled in Sec. 3.1. The results comprise closed-form expressions for the propagation of disparity measurement errors in an exemplary obstacle detection algorithm. Additionally, the transformation from disparity coordinates to Cartesian distances has been analysed in terms of statistical consistency. A novel transformation with improved consistency has been proposed. In order to model the relevant statistical properties of stereo images, a deductive top-down approach has been followed. The model's validity is shown for a known parametrisation, which is challenging to establish in diverse and dynamic traffic scenes. Inductive modelling of the errors from empirical data is thus a viable alternative. To this end, reference data has to be obtained, which has been studied in Ch. 6. Novel post processing methods for object detection and tracking have been presented and evaluated for the example of a LIDAR sensor.

Second, a statistical analysis of the fundamental problem of matching two registered sets of features has been developed in Sec. 3.2. Solving this kind of problem is essential for localisation tasks. Novel closed-form expressions for the covariance of the estimated transformation have been derived and a de-biased estimator has been proposed. These models can be employed for example in probabilistic filtering and sensor data fusion or to derive sensor accuracy constraints.

State estimation and prediction Algorithms that estimate and predict the motion of traffic participants are a crucial prerequisite for realising many ADAS functions.

State estimators aggregate measurement information over time and estimate unobservable motion states. In order to model the errors in these state estimates, the foundations of the Cramér-Rao bound have been discussed and applied to the example of motion state estimation in Sec. 4.1. In many relevant cases, this lower bound on the estimation error covariance can be efficiently calculated. Thus, an analytical, time-dependent model of the state estimation uncertainty can be derived and propagated to subsequent processing steps from a model of measurement errors at individual time steps. Furthermore, the performance of a sub-optimal state estimator can be compared to the theoretical limit. Therefore, the potential benefit of more accurate but often computationally more expensive estimators can be quantified.

Kinematic predictions of a vehicle's motion are a crucial part of a state estimator and algorithms for situation interpretation, for example in AEB systems. Since predicting the

unforeseeable future driver behaviour is never exact, statistical error analyses have been conducted on large-scale datasets of recorded vehicle trajectories in Sec. 4.2. A Gaussian white process noise model has been used to model the prediction errors, which facilitates closed-form results. The model parameters have been estimated using the expectation maximisation principle. Novel explicit expressions of the estimator equations have been derived for the considered model structure. Evaluations of the probabilistic model confirm the estimated parameter values but also reveal some non-Gaussian effects. Thus, future works could relax the model assumptions, for example with multimodal densities, and extend the parameter estimation scheme accordingly.

Situation interpretation In order to enhance the reliability of long-term motion predictions, additional semantic driver intentions can be inferred. In Sec. 5.1, novel performance bounds on such a manoeuvre recognition task have been developed. To this end, it has been proposed to formulate the task as a change detection problem in a linear dynamical system. Thus, known results on the statistical properties of such a problem can be applied. A novel recursive form of the theoretical performance bound has been derived in this work. This recursion facilitates an efficient computation and yields further insight into the fundamental dependence between the duration of an observation and the achievable rate of correctly detecting a manoeuvre.

Based on state estimates and motion predictions, criticality measures are used for decision making in an AEB system. A generic framework for modelling the uncertainty in these risk metrics has been proposed in Sec. 5.2. The framework has been exemplified for two specific criticality measures for which closed-form models have been developed. Furthermore, the timing of a brake intervention has been studied. To this end, an approximation method for calculating level crossing probabilities of a Gauss-Markov process has been derived and evaluated. This result generalises previous works on the special case of independent Gaussians.

8.2 Outlook

The obtained models and the techniques used to derive them can be employed in a wide range of applications. Two possible directions for further research, from which the advancement of driver assistance technology could benefit, are the definition of performance metrics and probabilistic decision making.

Firstly, knowing the goodness of a theoretical optimum solution helps to objectively assess the quality of an algorithm. Current research activities on this topic focus on the usually intractable problem of multi-target tracking. Predicting driver intentions and the future trajectories of traffic participants is a similarly difficult problem with a high relevance to higher automated driving functions. However, objective metrics and insight on fundamental limitations in these tasks are still lacking. With the help of the framework and models developed in this work, such insight could be developed. The added value is that one can differentiate conditions under which current algorithms

already perform close to the theoretical optimum from those where there is still room for improvement.

Secondly, analytical models of uncertainty facilitate the self-assessment of reliability in an algorithm. Due to limited computational resources in embedded devices, numerical approaches are difficult to apply to this end. On the other hand, significantly less demanding implementations are enabled by closed-form models. For example, the developed uncertainty propagation in criticality measures could be included in decision making algorithms for AEB brake interventions.

The scope of this work has been to systematically analyse an entire signal processing chain instead of only a particular component. The usefulness of such an end-to-end modelling approach has been demonstrated for the examples of deriving sensor parameter constraints and the analysis of performance bounds. Therefore, contributions of this work are given by the overall framework in which previous results could be integrated but also the development of novel methods and models. Nevertheless, potential approaches to more detailed models have been identified which could not be covered within the scope of this thesis. These ideas could be further developed in dedicated analyses and subsequently integrated into the overall framework.

A Appendix

A.1 Derivatives and matrix identities

The following provides an overview of basic matrix identities that are used in this thesis. A thorough treatment and further relations are detailed for example in [75, 131].

Throughout this work, the numerator-layout is used for the gradient or Jacobian of scalar- or vector-valued functions, respectively. Let $\mathbf{f}(\mathbf{x}) \in \mathbb{R}^m$ denote a vector-valued differentiable function of $\mathbf{x} \in \mathbb{R}^n$:

$$\nabla_{\mathbf{x}} f(\mathbf{x}) \coloneqq \begin{bmatrix} \frac{\partial f(\mathbf{x})}{\partial x_1} & \dots & \frac{\partial f(\mathbf{x})}{\partial x_n} \end{bmatrix}, \qquad \nabla_{\mathbf{x}} \mathbf{f}(\mathbf{x}) \coloneqq \begin{bmatrix} \frac{\partial f_1(\mathbf{x})}{\partial x_1} & \dots & \frac{\partial f_1(\mathbf{x})}{\partial x_n} \\ \vdots & \ddots & \vdots \\ \frac{\partial f_m(\mathbf{x})}{\partial x_1} & \dots & \frac{\partial f_m(\mathbf{x})}{\partial x_n} \end{bmatrix}. \tag{A.1}$$

The derivative of a scalar-valued function $f(\mathbf{X})$ of a matrix $\mathbf{X} \in \mathbb{R}^{m \times n}$ is denoted by:

$$\frac{\partial}{\partial \mathbf{X}} f(\mathbf{X}) \coloneqq \begin{bmatrix} \frac{\partial f(\mathbf{X})}{\partial X_{1,1}} & \dots & \frac{\partial f(\mathbf{X})}{\partial X_{1,n}} \\ \vdots & \ddots & \vdots \\ \frac{\partial f(\mathbf{X})}{\partial X_{m,1}} & \dots & \frac{\partial f(\mathbf{X})}{\partial X_{m,n}} \end{bmatrix}. \tag{A.2}$$

Two types of matrix functions, the determinant and the trace operator, frequently occur in the context of maximum likelihood estimation, as in Sec. 4.2.4. Closed-form results for the derivatives are available in these cases.

Theorem A.1 (Matrix derivatives of determinant and trace [33])
Let $\mathbf{X}$ denote an invertible square matrix $\mathbf{X} \in \mathbb{R}^{n \times n}$ and $\mathbf{A} \in \mathbb{R}^{m \times n}$, $\mathbf{B} \in \mathbb{R}^{n \times p}$. It holds that:

$$\frac{\partial}{\partial \mathbf{X}} \det(\mathbf{X}) = \det(\mathbf{X}) \left(\mathbf{X}^{-1}\right)^{\mathrm{T}} \tag{A.3}$$

$$\frac{\partial}{\partial \mathbf{X}} \log|\det(\mathbf{X})| = \left(\mathbf{X}^{-1}\right)^{\mathrm{T}} \tag{A.4}$$

$$\frac{\partial}{\partial \mathbf{X}} \operatorname{tr}\left(\mathbf{A}\mathbf{X}^{-1}\mathbf{B}\right) = -\left(\mathbf{X}^{-1}\mathbf{B}\mathbf{A}\mathbf{X}^{-1}\right)^{\mathrm{T}}. \tag{A.5}$$

The determinant and trace of a Kronecker product between two square matrices can be reformulated. This relationship makes it straightforward to find the derivative with respect to one of the two factors.

Theorem A.2 (Determinant, trace and inverse of Kronecker products [33])
It holds for $\mathbf{A} \in \mathbb{R}^{n\times n}, \mathbf{B} \in \mathbb{R}^{m\times m}$ that:

$$\det(\mathbf{A}\otimes\mathbf{B}) = \det(\mathbf{A})^m \det(\mathbf{B})^n \tag{A.6}$$
$$\operatorname{tr}(\mathbf{A}\otimes\mathbf{B}) = \operatorname{tr}(\mathbf{A})\operatorname{tr}(\mathbf{B}) \ . \tag{A.7}$$

Furthermore, if $\mathbf{A}^{-1}$ and $\mathbf{B}^{-1}$ exist, one has the equality:

$$\mathbf{A}^{-1}\otimes\mathbf{B}^{-1} = (\mathbf{A}\otimes\mathbf{B})^{-1} \ . \tag{A.8}$$

Theorem A.3 (Woodbury identity [75])
The following identity, also known as *matrix inversion lemma*, holds for regular matrices $\mathbf{A} \in \mathbb{R}^{n\times n}, \mathbf{B} \in \mathbb{R}^{m\times m}$ and $\mathbf{C} \in \mathbb{R}^{n\times m}$:

$$\left(\mathbf{A}+\mathbf{C}\mathbf{B}\mathbf{C}^{\mathrm{T}}\right)^{-1} = \mathbf{A}^{-1} - \mathbf{A}^{-1}\mathbf{C}\left(\mathbf{B}^{-1}+\mathbf{C}^{\mathrm{T}}\mathbf{A}^{-1}\mathbf{C}\right)^{-1}\mathbf{C}^{\mathrm{T}}\mathbf{A}^{-1} \ . \tag{A.9}$$

Furthermore, it follows that:

$$\begin{aligned}\left(\mathbf{A}+\mathbf{C}\mathbf{B}\mathbf{C}^{\mathrm{T}}\right)^{-1}\mathbf{C}\mathbf{B} &= \left(\mathbf{A}^{-1} - \mathbf{A}^{-1}\mathbf{C}\left(\mathbf{B}^{-1}+\mathbf{C}^{\mathrm{T}}\mathbf{A}^{-1}\mathbf{C}\right)^{-1}\mathbf{C}^{\mathrm{T}}\mathbf{A}^{-1}\right)\mathbf{C}\mathbf{B} \\ &= \mathbf{A}^{-1}\mathbf{C}\left(\mathbf{I}_{m\times m} - \left(\mathbf{B}^{-1}+\mathbf{C}^{\mathrm{T}}\mathbf{A}^{-1}\mathbf{C}\right)^{-1}\mathbf{C}^{\mathrm{T}}\mathbf{A}^{-1}\mathbf{C}\right)\mathbf{B} \\ &= \mathbf{A}^{-1}\mathbf{C}\left(\mathbf{B}^{-1}+\mathbf{C}^{\mathrm{T}}\mathbf{A}^{-1}\mathbf{C}\right)^{-1} \ . \end{aligned} \tag{A.10}$$

In the more general case of a regular matrix $\mathbf{A}$ and a possibly singular $\mathbf{B}$, the following identity applies instead of (A.9):

$$\left(\mathbf{A}+\mathbf{C}\mathbf{B}\mathbf{C}^{\mathrm{T}}\right)^{-1} = \mathbf{A}^{-1} - \mathbf{A}^{-1}\mathbf{C}\mathbf{B}\left(\mathbf{I}_{m\times m}+\mathbf{C}^{\mathrm{T}}\mathbf{A}^{-1}\mathbf{C}\mathbf{B}\right)^{-1}\mathbf{C}^{\mathrm{T}}\mathbf{A}^{-1} \ . \tag{A.11}$$

Theorem A.4 (Inversion of partitioned matrices [75])
We assume an invertible matrix $\mathbf{A} \in \mathbb{R}^{n\times n}$ which is written in a partitioned form with blocks $\mathbf{A}_{11} \in \mathbb{R}^{m\times m}$, $\mathbf{A}_{12} \in \mathbb{R}^{m\times p}$, $\mathbf{A}_{21} \in \mathbb{R}^{p\times m}$ and $\mathbf{A}_{22} \in \mathbb{R}^{p\times p}$ where $n = m + p$. The inverse of $\mathbf{A}$ is obtained block-wise as follows:

$$\begin{bmatrix}\mathbf{A}_{11} & \mathbf{A}_{12}\\ \mathbf{A}_{21} & \mathbf{A}_{22}\end{bmatrix}^{-1} = \begin{bmatrix}\mathbf{B}_{11}^{-1} & \mathbf{B}_{12}\\ \mathbf{B}_{21} & \mathbf{B}_{22}^{-1}\end{bmatrix} \tag{A.12a}$$

where

$$\mathbf{B}_{11} \coloneqq \mathbf{A}_{11} - \mathbf{A}_{12}\mathbf{A}_{22}^{-1}\mathbf{A}_{21} \ , \tag{A.12b}$$

$$\mathbf{B}_{22} \coloneqq \mathbf{A}_{22} - \mathbf{A}_{21}\mathbf{A}_{11}^{-1}\mathbf{A}_{12} \,, \tag{A.12c}$$

$$\mathbf{B}_{12} \coloneqq -\mathbf{A}_{11}^{-1}\mathbf{A}_{12}\left(\mathbf{A}_{22} - \mathbf{A}_{21}\mathbf{A}_{11}^{-1}\mathbf{A}_{12}\right)^{-1} = -\mathbf{A}_{11}^{-1}\mathbf{A}_{12}\mathbf{B}_{22}^{-1} \,, \tag{A.12d}$$

$$\mathbf{B}_{21} \coloneqq -\left(\mathbf{A}_{22} - \mathbf{A}_{21}\mathbf{A}_{11}^{-1}\mathbf{A}_{12}\right)^{-1}\mathbf{A}_{21}\mathbf{A}_{11}^{-1} = -\mathbf{B}_{22}^{-1}\mathbf{A}_{21}\mathbf{A}_{11}^{-1} \,. \tag{A.12e}$$

A.2 Constant turn rate and acceleration model

First, the time evolution $\mathbf{x}\left(t_k + T\right)$ of the state in the CTRA model is calculated by solving the differential equation (4.48) with an initial value $\mathbf{x}_k \coloneqq \mathbf{x}\left(t_k\right)$:

$$\begin{aligned} x\left(t_k + T\right) = x_k &+ \cos\left(\theta_k\right)\left[v_k\frac{\sin\left(\omega_k T\right)}{\omega_k} + a_k\left(\frac{\sin\left(\omega_k T\right)}{\omega_k}T - \frac{1-\cos\left(\omega_k T\right)}{\omega_k^2}\right)\right] \\ &- \sin\left(\theta_k\right)\left[v_k\frac{1-\cos\left(\omega_k T\right)}{\omega_k} - a_k\left(\frac{\cos\left(\omega_k T\right)}{\omega_k}T - \frac{\sin\left(\omega_k T\right)}{\omega_k^2}\right)\right] \end{aligned} \tag{A.13a}$$

$$\begin{aligned} y\left(t_k + T\right) = y_k &+ \cos\left(\theta_k\right)\left[v_k\frac{1-\cos\left(\omega_k T\right)}{\omega_k} - a_k\left(\frac{\cos\left(\omega_k T\right)}{\omega_k}T - \frac{\sin\left(\omega_k T\right)}{\omega_k^2}\right)\right] \\ &+ \sin\left(\theta_k\right)\left[v_k\frac{\sin\left(\omega_k T\right)}{\omega_k} + a_k\left(\frac{\sin\left(\omega_k T\right)}{\omega_k}T - \frac{1-\cos\left(\omega_k T\right)}{\omega_k^2}\right)\right] \end{aligned} \tag{A.13b}$$

$$v\left(t_k + T\right) = v_k + a_k T \tag{A.13c}$$

$$\theta\left(t_k + T\right) = \theta_k + \omega_k T \tag{A.13d}$$

$$a\left(t_k + T\right) = a_k \tag{A.13e}$$

$$\omega\left(t_k + T\right) = \omega_k \,. \tag{A.13f}$$

For $T = T_\mathrm{S}$, this yields the discrete time state transition $\mathbf{x}_{k+1} = \mathbf{f}\left(\mathbf{x}_k\right)$. Note that the first two equations possess a singularity at $\omega_k = 0$ which can be resolved by rewriting the trigonometric functions as series expansions.

Second, the state transition matrix $\mathbf{\Phi}\left(t, t_0\right)$ is derived. To this end, the Jacobian of $\mathbf{f}\left(t\right)$ is obtained as:

$$\mathbf{F}\left(t\right) \coloneqq \nabla_{\mathbf{x}(t)}\mathbf{f}\left(\mathbf{x}\left(t\right)\right) = \begin{bmatrix} 0 & 0 & \cos\left(\theta\left(t\right)\right) & -v\left(t\right)\sin\left(\theta\left(t\right)\right) & 0 & 0 \\ 0 & 0 & \sin\left(\theta\left(t\right)\right) & v\left(t\right)\cos\left(\theta\left(t\right)\right) & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \end{bmatrix} . \tag{A.14}$$

A closed-form expression for $\mathbf{\Phi}(t, t_0)$ follows according to (4.73):

$$\mathbf{\Phi}(t, t_0) = \begin{bmatrix} 1 & 0 & \Phi_{1,3}(t, t_0) & \Phi_{1,4}(t, t_0) & 0 & 0 \\ 0 & 1 & \Phi_{2,3}(t, t_0) & \Phi_{2,4}(t, t_0) & 0 & 0 \\ 0 & 0 & 1 & 0 & t - t_0 & 0 \\ 0 & 0 & 0 & 1 & 0 & t - t_0 \\ 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 \end{bmatrix} \tag{A.15a}$$

$$\Phi_{1,3}(t, t_0) = \cos(\theta_0)\,\alpha(t, t_0) - \sin(\theta_0)\,\beta(t, t_0) \tag{A.15b}$$

$$\Phi_{2,3}(t, t_0) = \sin(\theta_0)\,\alpha(t, t_0) + \cos(\theta_0)\,\beta(t, t_0) \tag{A.15c}$$

$$\begin{aligned} \Phi_{1,4}(t, t_0) = &- \cos(\theta_0)\left[\alpha(t, t_0)\,\frac{a_0}{\omega_0} + \beta(t, t_0)\,v_0 - \gamma(t, t_0)\,a_0\,(t - t_0)\right] \\ &- \sin(\theta_0)\left[\alpha(t, t_0)\,(v_0 + a_0\,(t - t_0)) - \beta(t, t_0)\,\frac{a_0}{\omega_0}\right] \end{aligned} \tag{A.15d}$$

$$\begin{aligned} \Phi_{2,4}(t, t_0) = &- \sin(\theta_0)\left[\alpha(t, t_0)\,\frac{a_0}{\omega_0} + \beta(t, t_0)\,v_0 - \gamma(t, t_0)\,a_0\,(t - t_0)\right] \\ &+ \cos(\theta_0)\left[\alpha(t, t_0)\,(v_0 + a_0\,(t - t_0)) - \beta(t, t_0)\,\frac{a_0}{\omega_0}\right] \end{aligned} \tag{A.15e}$$

with the following abbreviations

$$\begin{aligned} \alpha(t, t_0) &:= \frac{\sin(\omega_0\,(t - t_0))}{\omega_0}\,, & \beta(t, t_0) &:= \frac{1 - \cos(\omega_0\,(t - t_0))}{\omega_0}\,, \\ \gamma(t, t_0) &:= \frac{\cos(\omega_0\,(t - t_0))}{\omega_0}\,, & \mathbf{x}_0 &:= \mathbf{x}(t_0)\ . \end{aligned} \tag{A.15f}$$

The process noise covariance matrix $\mathbf{Q}_k$ of the discrete-time system is obtained by inserting $\mathbf{\Phi}(t, t_0)$ from (A.15) into (4.45b):

$$\operatorname{cov}(\mathbf{w}_k) = \mathbf{Q}_k = \begin{bmatrix} 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & \frac{1}{3}{T_\mathrm{S}}^3 S_a & 0 & \frac{1}{2}{T_\mathrm{S}}^2 S_a & 0 \\ 0 & 0 & 0 & \frac{1}{3}{T_\mathrm{S}}^3 S_\omega & 0 & \frac{1}{2}{T_\mathrm{S}}^2 S_\omega \\ 0 & 0 & \frac{1}{2}{T_\mathrm{S}}^2 S_a & 0 & T_\mathrm{S} S_a & 0 \\ 0 & 0 & 0 & \frac{1}{2}{T_\mathrm{S}}^2 S_\omega & 0 & T_\mathrm{S} S_\omega \end{bmatrix}\ . \tag{A.16}$$

In order to enable the estimation of the parameters S_a and S_ω in Sec. 4.2.4 an equivalent formulation in terms of an auxiliary noise process $\mathbf{w}'_k$ with $\mathbf{w}_k = \mathbf{G}\mathbf{w}'_k$ is introduced:

$$\operatorname{cov}(\mathbf{w}'_k) = \mathbf{Q}'_k = \underbrace{\begin{bmatrix} \frac{1}{3}{T_\mathrm{S}}^3 & \frac{1}{2}{T_\mathrm{S}}^2 \\ \frac{1}{2}{T_\mathrm{S}}^2 & T_\mathrm{S} \end{bmatrix}}_{=:\mathbf{Q}_1} \otimes \underbrace{\begin{bmatrix} S_a & 0 \\ 0 & S_\omega \end{bmatrix}}_{=\mathbf{S}}\,, \qquad \mathbf{G} := \begin{bmatrix} 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}\ . \tag{A.17}$$

Therefore, $\mathbf{S}$ can be estimated using (4.60) in conjunction with the result (4.69) for covariance matrices with Kronecker product structure.

A concluding remark concerns the order in which the state transition matrix $\mathbf{\Phi}(t, t_0)$ has been derived: At first, the system differential equation is linearised (A.14) and subsequently integrated in (A.15) (linearisation and discretisation), as in [107].

Alternatively, one can first solve the time evolution $\mathbf{x}(t)$ starting from an initial state $\mathbf{x}_0$ as in (A.13) and differentiate this result with respect to $\mathbf{x}_0$. Due to the exchanged order of the operations, this approach is known as discretisation and linearisation. The approach is analytically shown to be more accurate in [70].

For the CTRA model, the difference between both approaches lies in the treatment of acceleration $a(t)$ and yaw rate $\omega(t)$. Since these state variables do not occur in the first two of the differential equations of the model (4.48), the Jacobian $\mathbf{F}(t)$ from (A.14) contains zeros at the respective positions. However, when the time evolution (A.13) is calculated first, such a dependence occurs and $\mathbf{\Phi}(t, t_0)$ features additional non-zero elements:

$$\begin{aligned}\Phi_{1,5}(t, t_0) = &\cos(\theta_0)\left[\alpha(t, t_0)(t - t_0) - \frac{\beta(t, t_0)}{\omega_0}\right] \\ &+ \sin(\theta_0)\left[\gamma(t, t_0)(t - t_0) - \frac{\alpha(t, t_0)}{\omega_0}\right]\end{aligned} \tag{A.18a}$$

$$\begin{aligned}\Phi_{2,5}(t, t_0) = &\sin(\theta_0)\left[\alpha(t, t_0)(t - t_0) - \frac{\beta(t, t_0)}{\omega_0}\right] \\ &- \cos(\theta_0)\left[\gamma(t, t_0)(t - t_0) - \frac{\alpha(t, t_0)}{\omega_0}\right]\end{aligned} \tag{A.18b}$$

$$\begin{aligned}\Phi_{1,6}(t, t_0) = &\cos(\theta_0)\left[-\alpha(t, t_0)\left(\frac{2a_0}{\omega_0}(t - t_0) + \frac{v_0}{\omega_0}\right)\right. \\ &\left.+ \beta(t, t_0)\frac{2a_0}{\omega_0^2} + \gamma(t, t_0)\left(v_0(t - t_0) + a_0(t - t_0)^2\right)\right] \\ &- \sin(\theta_0)\left[\alpha(t, t_0)\left(v_0(t - t_0) + a_0(t - t_0)^2 - \frac{2a_0}{\omega_0^2}\right)\right. \\ &\left.- \beta(t, t_0)\frac{v_0}{\omega_0} + \gamma(t, t_0)\frac{2a_0}{\omega_0}(t - t_0)\right]\end{aligned} \tag{A.18c}$$

$$\begin{aligned}\Phi_{2,6}(t, t_0) = &\sin(\theta_0)\left[-\alpha(t, t_0)\left(\frac{2a_0}{\omega_0}(t - t_0) + \frac{v_0}{\omega_0}\right)\right. \\ &\left.+ \beta(t, t_0)\frac{2a_0}{\omega_0^2} + \gamma(t, t_0)\left(v_0(t - t_0) + a_0(t - t_0)^2\right)\right] \\ &+ \cos(\theta_0)\left[\alpha(t, t_0)\left(v_0(t - t_0) + a_0(t - t_0)^2 - \frac{2a_0}{\omega_0^2}\right)\right. \\ &\left.- \beta(t, t_0)\frac{v_0}{\omega_0} + \gamma(t, t_0)\frac{2a_0}{\omega_0}(t - t_0)\right]\end{aligned} . \tag{A.18d}$$

All other entries are identical to (A.15).

Since the approach is more accurate than the general method (4.72c), the extended transition matrix $\mathbf{\Phi}(t, t_0)$ is used in this work to propagate the initial state covariance according to (4.72a). Note that the propagation of the process noise $\mathbf{Q}(t_k + T, t_k)$ according to (4.72b) can be analytically solved in both cases. However, in contrast to (A.16), the result becomes much lengthier if the additional entries from (A.18) are taken into account.

A.3 Reformulation of mixed Kronecker matrix product

The goal is to show that the trace of a mixed Kronecker and matrix product can be expressed as a sum of traces of matrix products:

$$\operatorname{tr}\left((\mathbf{S} \otimes \mathbf{Q}_1)^{-1}\mathbf{M}\right) = \operatorname{tr}\left(\left(\mathbf{S}^{-1} \otimes \mathbf{Q}_1^{-1}\right)\mathbf{M}\right) = \sum_{i=1}^{n_Q}\sum_{j=1}^{n_Q} \tilde{Q}_{ij} \operatorname{tr}\left(\mathbf{S}^{-1}\tilde{\mathbf{M}}^{(j,i)}\right) . \tag{A.19}$$

The identity is helpful in order to find the derivative with respect to $\mathbf{S}$. At first, the abbreviation $\mathbf{U} \coloneqq \left(\mathbf{S}^{-1} \otimes \mathbf{Q}_1^{-1}\right)\mathbf{M}$ is introduced. The elements of $\mathbf{S}^{-1}$ are denoted by $\tilde{S}_{m,n},\ m = 1, \ldots, n_S,\ n = 1, \ldots, n_S$. A similar notation is chosen for the elements of the inverse $\mathbf{Q}_1^{-1}$, that is $\tilde{Q}_{i,j}$, $i = 1, \ldots, n_Q$, $j = 1, \ldots, n_Q$. Moreover, a block matrix notation is introduced for $\mathbf{M}$, where $\mathbf{M}_{i:i+n_Q,j:j+n_Q}$ denotes the $n_Q \times n_Q$ sub-matrix at position i, j. With these notations, the $n_S n_Q \times n_S n_Q$-dimensional $\mathbf{U}$ can be written explicitly:

$$\mathbf{U} = \begin{bmatrix} \tilde{S}_{1,1}\mathbf{Q}_1^{-1} & \cdots & \tilde{S}_{1,n_S}\mathbf{Q}_1^{-1} \\ \vdots & \ddots & \vdots \\ \tilde{S}_{n_S,1}\mathbf{Q}_1^{-1} & \cdots & \tilde{S}_{n_S,n_S}\mathbf{Q}_1^{-1} \end{bmatrix} \cdot \begin{bmatrix} \mathbf{M}_{1:n_Q,1:n_Q} & \cdots & \mathbf{M}_{1:n_Q,(n_S-1)n_Q:n_S n_Q} \\ \vdots & \ddots & \vdots \\ \mathbf{M}_{(n_S-1)n_Q:n_S n_Q,1:n_Q} & \cdots & \mathbf{M}_{(n_S-1)n_Q:n_S n_Q,(n_S-1)n_Q:n_S n_Q} \end{bmatrix} .$$

As one is interested in $\operatorname{tr}(\mathbf{U})$, only the diagonal elements of this product are studied:

$$\begin{aligned} U_{k,k} &= \sum_{n=1}^{n_S}\sum_{j=1}^{n_Q} \tilde{S}_{1,n}\tilde{Q}_{k,j}M_{j+(n-1)n_Q,k} , && k = 1, \ldots, n_Q \\ U_{k,k} &= \sum_{n=1}^{n_S}\sum_{j=1}^{n_Q} \tilde{S}_{2,n}\tilde{Q}_{k-n_Q,j}M_{j+(n-1)n_Q,k} , && k = n_Q + 1, \ldots, 2n_Q \\ &\vdots \\ U_{k,k} &= \sum_{n=1}^{n_S}\sum_{j=1}^{n_Q} \tilde{S}_{n_S,n}\tilde{Q}_{k-(n_S-1)n_Q,j}M_{j+(n-1)n_Q,k} , && k = (n_S-1)\,n_Q + 1, \ldots, n_S n_Q . \end{aligned}$$

Summing up the diagonal yields:

$$\begin{aligned}
\operatorname{tr}(\mathbf{U}) &= \sum_{k=1}^{n_Q}\sum_{n=1}^{n_S}\sum_{j=1}^{n_Q} \tilde{S}_{1,n}\tilde{Q}_{k,j}M_{j+(n-1)n_Q,k} \\
&\quad + \sum_{k=n_Q+1}^{2n_Q}\sum_{n=1}^{n_S}\sum_{j=1}^{n_Q} \tilde{S}_{2,n}\tilde{Q}_{k-n_Q,j}M_{j+(n-1)n_Q,k} \\
&\quad + \cdots + \sum_{k=(n_S-1)n_Q+1}^{n_Sn_Q}\sum_{n=1}^{n_S}\sum_{j=1}^{n_Q} \tilde{S}_{n_S,n}\tilde{Q}_{k-(n_S-1)n_Q,j}M_{j+(n-1)n_Q,k} \\
&= \sum_{k=1}^{n_Q}\sum_{n=1}^{n_S}\sum_{j=1}^{n_Q} \tilde{S}_{1,n}\tilde{Q}_{k,j}M_{j+(n-1)n_Q,k} + \sum_{k=1}^{n_Q}\sum_{n=1}^{n_S}\sum_{j=1}^{n_Q} \tilde{S}_{2,n}\tilde{Q}_{k,j}M_{j+(n-1)n_Q,k+n_Q} \\
&\quad + \cdots + \sum_{k=1}^{n_Q}\sum_{n=1}^{n_S}\sum_{j=1}^{n_Q} \tilde{S}_{n_S,n}\tilde{Q}_{k,j}M_{j+(n-1)n_Q,k+(n_S-1)n_Q} \\
&= \sum_{m=1}^{n_S}\sum_{k=1}^{n_Q}\sum_{n=1}^{n_S}\sum_{j=1}^{n_Q} \tilde{S}_{m,n}\tilde{Q}_{k,j}M_{j+(n-1)n_Q,k+(m-1)n_Q} \\
&= \sum_{k=1}^{n_Q}\sum_{j=1}^{n_Q}\tilde{Q}_{k,j}\sum_{m=1}^{n_S}\sum_{n=1}^{n_S} \tilde{S}_{m,n}M_{j+(n-1)n_Q,k+(m-1)n_Q} \,. \qquad \text{(A.21)}
\end{aligned}$$

Key to finding an expression in $\mathbf{S}$ again is to introduce the matrix $\tilde{\mathbf{M}}^{(j,k)}$ whose elements are taken from $\mathbf{M}$ according to a regular pattern:

$$\tilde{\mathbf{M}}^{(j,k)} = \Big[M_{j+(u-1)n_Q,\; k+(v-1)n_Q}\Big]_{\substack{u=1,\dots,n_S\\ v=1,\dots,n_S}} \,. \qquad \text{(A.22)}$$

With the definition of $\tilde{\mathbf{M}}^{(j,k)}$, one obtain the desired result from (A.21):

$$\operatorname{tr}(\mathbf{U}) = \sum_{k=1}^{n_Q}\sum_{j=1}^{n_Q}\tilde{Q}_{k,j}\sum_{m=1}^{n_S}\sum_{n=1}^{n_S}\tilde{S}_{m,n}\tilde{M}^{(j,k)}_{n,m} = \sum_{k=1}^{n_Q}\sum_{j=1}^{n_Q}\tilde{Q}_{k,j}\operatorname{tr}\left(\mathbf{S}^{-1}\tilde{\mathbf{M}}^{(j,k)}\right) \,. \qquad \text{(A.23)}$$

A.4 Recursive GLR test statistic as quadratic form

The following derivation details the simplification of λ_{L+1} from (5.37) to an equivalent expression but written as a quadratic form in (5.38). For brevity, the abbreviation $\tilde{\boldsymbol{\theta}} \coloneqq \boldsymbol{\theta}_1 - \boldsymbol{\theta}_0$ is used.

Firstly, the recursive expression for $\mathbf{M}_{L+1,11}$ from (5.31) is inserted into (5.37). The definition of λ_L from the previous iteration is then recovered. The remaining terms contain matrix products which can be calculated recursively as well. To this end, a recursively defined abbreviation $\boldsymbol{\Lambda}_L$, $L \geq 1$ with $\boldsymbol{\Lambda}_1 = \mathbf{0}_{n_\theta \times n}$ is introduced:

$$\underbrace{\boldsymbol{\Lambda}_{L+1}}_{n_\theta \times n} = \underbrace{\left(\boldsymbol{\Phi}_L\tilde{\boldsymbol{\theta}}\right)^{\mathrm{T}}}_{n_\theta \times L\cdot m} \cdot \underbrace{\boldsymbol{\mathcal{I}}_{\mathbf{V}_L}}_{L\cdot m \times L\cdot m} \cdot \underbrace{\mathbf{H}_L}_{L\cdot m \times L\cdot n} \cdot \underbrace{\mathbf{M}^{\mathrm{T}}_{L+1,21}}_{L\cdot n \times n}$$

$$
\begin{aligned}
&= \sum_{l=1}^{L} \left(\boldsymbol{\varphi}_l^{\mathrm{T}}\tilde{\boldsymbol{\theta}}\right)^{\mathrm{T}} \boldsymbol{\mathcal{I}}_{\mathbf{v}}\mathbf{C}\mathbf{M}_{L+1,21}^{(l)\ \mathrm{T}} \\
&\overset{(5.35)}{=} \sum_{l=1}^{L-1} \left(\boldsymbol{\varphi}_l^{\mathrm{T}}\tilde{\boldsymbol{\theta}}\right)^{\mathrm{T}} \boldsymbol{\mathcal{I}}_{\mathbf{v}}\mathbf{C} \left(\prod_{j=l}^{L-1} \boldsymbol{\Gamma}_{j+1}\mathbf{M}_{l,22}\right)^{\mathrm{T}} \boldsymbol{\Gamma}_{L+1}^{\mathrm{T}} + \left(\boldsymbol{\varphi}_L^{\mathrm{T}}\tilde{\boldsymbol{\theta}}\right)^{\mathrm{T}} \boldsymbol{\mathcal{I}}_{\mathbf{v}}\mathbf{C} \left(\boldsymbol{\Gamma}_{L+1}\mathbf{M}_{L,22}\right)^{\mathrm{T}} \\
&= -\left(\boldsymbol{\Lambda}_L\mathbf{M}_{L,22}^{-1} + \left(\boldsymbol{\varphi}_L^{\mathrm{T}}\tilde{\boldsymbol{\theta}}\right)^{\mathrm{T}} \boldsymbol{\mathcal{I}}_{\mathbf{v}}\mathbf{C}\right) \left(\boldsymbol{\mathcal{S}} + \mathbf{M}_{L,22}^{-1}\right)^{-1} \mathbf{D}_{12}\mathbf{D}_{22}^{-1} \,. \qquad \text{(A.24)}
\end{aligned}
$$

It has been used here that $\boldsymbol{\Gamma}_{L+1}$ from (5.34) can be written solely in terms of $\mathbf{M}_{L,22}$:

$$
\begin{aligned}
\boldsymbol{\Gamma}_{L+1} &= -\mathbf{M}_{L+1,22}\mathbf{D}_{21} \left(\mathbf{D}_{11} + \mathbf{M}_{L,22}^{-1}\right)^{-1} \mathbf{M}_{L,22}^{-1} \qquad \text{(A.25)} \\
&\overset{(5.33)}{=} -\left(\mathbf{D}_{22} - \mathbf{D}_{21}\left(\mathbf{D}_{11} + \mathbf{M}_{L,22}^{-1}\right)^{-1}\mathbf{D}_{12}\right)^{-1} \mathbf{D}_{21} \left(\mathbf{D}_{11} + \mathbf{M}_{L,22}^{-1}\right)^{-1} \\
&\overset{(A.10)}{=} -\mathbf{D}_{22}^{-1}\mathbf{D}_{21} \left(\boldsymbol{\mathcal{S}} + \mathbf{M}_{L,22}^{-1}\right)^{-1} \mathbf{M}_{L,22}^{-1} \,. \qquad \text{(A.26)}
\end{aligned}
$$

Thus, one obtains λ_{L+1} from (5.37) as $\lambda_{L+1} = \lambda_L + \Delta\lambda_{L+1}$ where the additive term $\Delta\lambda_{L+1}$ is given by:

$$
\begin{aligned}
\Delta\lambda_{L+1} = &\underbrace{\left(\boldsymbol{\Lambda}_L + \left(\boldsymbol{\varphi}_L^{\mathrm{T}}\tilde{\boldsymbol{\theta}}\right)^{\mathrm{T}} \boldsymbol{\mathcal{I}}_{\mathbf{v}}\mathbf{C}\mathbf{M}_{L,22}\right)}_{=:\boldsymbol{\alpha}} \cdot \underbrace{\boldsymbol{\mathcal{S}}\left(\boldsymbol{\mathcal{S}} + \mathbf{M}_{L,22}^{-1}\right)^{-1}\mathbf{M}_{L,22}^{-1}}_{=:\boldsymbol{\beta}} \\
&\cdot \underbrace{\left(\boldsymbol{\Lambda}_L + \left(\boldsymbol{\varphi}_L^{\mathrm{T}}\tilde{\boldsymbol{\theta}}\right)^{\mathrm{T}} \boldsymbol{\mathcal{I}}_{\mathbf{v}}\mathbf{C}\mathbf{M}_{L,22}\right)^{\mathrm{T}}}_{=:\boldsymbol{\alpha}^{\mathrm{T}}} - 2\boldsymbol{\Lambda}_{L+1}\mathbf{C}^{\mathrm{T}}\boldsymbol{\mathcal{I}}_{\mathbf{v}} \left(\boldsymbol{\varphi}_{L+1}^{\mathrm{T}}\tilde{\boldsymbol{\theta}}\right) \qquad \text{(A.27)} \\
&+ \left(\boldsymbol{\varphi}_{L+1}^{\mathrm{T}}\tilde{\boldsymbol{\theta}}\right)^{\mathrm{T}} \left(\boldsymbol{\mathcal{I}}_{\mathbf{v}} - \boldsymbol{\mathcal{I}}_{\mathbf{v}}\mathbf{C}\mathbf{M}_{L+1,22}\mathbf{C}^{\mathrm{T}}\boldsymbol{\mathcal{I}}_{\mathbf{v}}\right) \left(\boldsymbol{\varphi}_{L+1}^{\mathrm{T}}\tilde{\boldsymbol{\theta}}\right) \,.
\end{aligned}
$$

While the second and third term contain $\boldsymbol{\Lambda}_{L+1}$ and $\mathbf{M}_{L+1,22}$, the first one involves the respective expressions from iteration L.

To simplify (A.27) the factor $\boldsymbol{\alpha}$ is rewritten:

$$
\begin{aligned}
\boldsymbol{\alpha} &= \left[\boldsymbol{\Lambda}_L\mathbf{M}_{L,22}^{-1} + \left(\boldsymbol{\varphi}_L^{\mathrm{T}}\tilde{\boldsymbol{\theta}}\right)^{\mathrm{T}} \boldsymbol{\mathcal{I}}_{\mathbf{v}}\mathbf{C}\right] \cdot \left[-\left(\boldsymbol{\mathcal{S}} + \mathbf{M}_{L,22}^{-1}\right)^{-1}\mathbf{D}_{12}\mathbf{D}_{22}^{-1}\right] \\
&\quad \cdot \left[-\left(\boldsymbol{\mathcal{S}} + \mathbf{M}_{L,22}^{-1}\right)^{-1}\mathbf{D}_{12}\mathbf{D}_{22}^{-1}\right]^{-1} \mathbf{M}_{L,22} \\
&\overset{(A.24)}{=} -\boldsymbol{\Lambda}_{L+1}\left(\mathbf{D}_{12}\mathbf{D}_{22}^{-1}\right)^{-1} \left(\boldsymbol{\mathcal{S}} + \mathbf{M}_{L,22}^{-1}\right)\mathbf{M}_{L,22} \,. \qquad \text{(A.28)}
\end{aligned}
$$

Inserting this expression into (A.27) yields:

$$
\begin{aligned}
\boldsymbol{\alpha}\boldsymbol{\beta}\boldsymbol{\alpha}^{\mathrm{T}} &= \boldsymbol{\Lambda}_{L+1}\left(\mathbf{D}_{12}\mathbf{D}_{22}^{-1}\right)^{-1} \left(\boldsymbol{\mathcal{S}} + \mathbf{M}_{L,22}^{-1}\right)\mathbf{M}_{L,22}\boldsymbol{\mathcal{S}} \left(\mathbf{D}_{22}^{-1}\mathbf{D}_{21}\right)^{-1} \boldsymbol{\Lambda}_{L+1}^{\mathrm{T}} \\
&= \boldsymbol{\Lambda}_{L+1}\left(\mathbf{D}_{12}\mathbf{D}_{22}^{-1}\right)^{-1} \left(\boldsymbol{\mathcal{S}}\mathbf{M}_{L,22}\boldsymbol{\mathcal{S}} + \boldsymbol{\mathcal{S}}\right) \left(\mathbf{D}_{22}^{-1}\mathbf{D}_{21}\right)^{-1} \boldsymbol{\Lambda}_{L+1}^{\mathrm{T}} \,. \qquad \text{(A.29)}
\end{aligned}
$$

In order to proceed further, $\boldsymbol{\mathcal{S}}$ as defined in (5.32) is reformulated using (5.27):

$$
\boldsymbol{\mathcal{S}} = \boldsymbol{\mathcal{S}}^{\mathrm{T}} = \mathbf{D}_{11} - \mathbf{D}_{12}\mathbf{D}_{22}^{-1}\mathbf{D}_{21}
$$

$$
\begin{aligned}
&= \mathbf{A}^{\mathrm{T}}\boldsymbol{\mathcal{I}}_{\mathbf{w}}\mathbf{A} - \mathbf{A}^{\mathrm{T}}\boldsymbol{\mathcal{I}}_{\mathbf{w}}\left(\boldsymbol{\mathcal{I}}_{\mathbf{w}} + \mathbf{C}^{\mathrm{T}}\boldsymbol{\mathcal{I}}_{\mathbf{v}}\mathbf{C}\right)^{-1}\boldsymbol{\mathcal{I}}_{\mathbf{w}}\mathbf{A} \\
&= \mathbf{A}^{\mathrm{T}}\boldsymbol{\mathcal{I}}_{\mathbf{w}}\left(\boldsymbol{\mathcal{I}}_{\mathbf{w}} + \mathbf{C}^{\mathrm{T}}\boldsymbol{\mathcal{I}}_{\mathbf{v}}\mathbf{C}\right)^{-1}\left(\mathbf{C}^{\mathrm{T}}\boldsymbol{\mathcal{I}}_{\mathbf{v}}\mathbf{C}\right)\mathbf{A} \\
&= -\left(\mathbf{D}_{12}\mathbf{D}_{22}^{-1}\right)\left(\mathbf{C}^{\mathrm{T}}\boldsymbol{\mathcal{I}}_{\mathbf{v}}\mathbf{C}\right)\mathbf{A}\,. \qquad \text{(A.30)}
\end{aligned}
$$

Inserting this into (A.29) yields:

$$
\begin{aligned}
&\boldsymbol{\alpha}\boldsymbol{\beta}\boldsymbol{\alpha}^{\mathrm{T}} \\
&= \boldsymbol{\Lambda}_{L+1}\left[\left(\mathbf{C}^{\mathrm{T}}\boldsymbol{\mathcal{I}}_{\mathbf{v}}\mathbf{C}\right)\mathbf{A}\mathbf{M}_{L,22}\mathbf{A}^{\mathrm{T}}\left(\mathbf{C}^{\mathrm{T}}\boldsymbol{\mathcal{I}}_{\mathbf{v}}\mathbf{C}\right) + \left(\mathbf{C}^{\mathrm{T}}\boldsymbol{\mathcal{I}}_{\mathbf{v}}\mathbf{C}\right)\boldsymbol{\mathcal{I}}_{\mathbf{w}}^{-1}\left(\boldsymbol{\mathcal{I}}_{\mathbf{w}} + \mathbf{C}^{\mathrm{T}}\boldsymbol{\mathcal{I}}_{\mathbf{v}}\mathbf{C}\right)\right]\boldsymbol{\Lambda}_{L+1}^{\mathrm{T}} \\
&= \boldsymbol{\Lambda}_{L+1}\mathbf{C}^{\mathrm{T}}\left[\boldsymbol{\mathcal{I}}_{\mathbf{v}} + \boldsymbol{\mathcal{I}}_{\mathbf{v}}\mathbf{C}\left(\boldsymbol{\mathcal{I}}_{\mathbf{w}}^{-1} + \mathbf{A}\mathbf{M}_{L,22}\mathbf{A}^{\mathrm{T}}\right)\mathbf{C}^{\mathrm{T}}\boldsymbol{\mathcal{I}}_{\mathbf{v}}\right]\mathbf{C}\boldsymbol{\Lambda}_{L+1}^{\mathrm{T}} \\
&\overset{(A.9)}{=} \boldsymbol{\Lambda}_{L+1}\mathbf{C}^{\mathrm{T}}\left[\boldsymbol{\mathcal{I}}_{\mathbf{v}} + \boldsymbol{\mathcal{I}}_{\mathbf{v}}\mathbf{C}\left(\boldsymbol{\mathcal{I}}_{\mathbf{w}} - \boldsymbol{\mathcal{I}}_{\mathbf{w}}\mathbf{A}\left(\mathbf{M}_{L,22}^{-1} + \mathbf{A}^{\mathrm{T}}\boldsymbol{\mathcal{I}}_{\mathbf{w}}^{-1}\mathbf{A}\right)^{-1}\mathbf{A}^{\mathrm{T}}\boldsymbol{\mathcal{I}}_{\mathbf{w}}\right)^{-1}\mathbf{C}^{\mathrm{T}}\boldsymbol{\mathcal{I}}_{\mathbf{v}}\right]\mathbf{C}\boldsymbol{\Lambda}_{L+1}^{\mathrm{T}} \\
&\overset{(5.33)}{=} \boldsymbol{\Lambda}_{L+1}\mathbf{C}^{\mathrm{T}}\left[\boldsymbol{\mathcal{I}}_{\mathbf{v}} - \boldsymbol{\mathcal{I}}_{\mathbf{v}}\mathbf{C}\left(\mathbf{C}^{\mathrm{T}}\boldsymbol{\mathcal{I}}_{\mathbf{v}}\mathbf{C} - \mathbf{M}_{L+1,22}^{-1}\right)^{-1}\mathbf{C}^{\mathrm{T}}\boldsymbol{\mathcal{I}}_{\mathbf{v}}\right]\mathbf{C}\boldsymbol{\Lambda}_{L+1}^{\mathrm{T}} \\
&\overset{(A.9)}{=} \boldsymbol{\Lambda}_{L+1}\mathbf{C}^{\mathrm{T}}\boldsymbol{\mathcal{I}}_{\mathbf{v}}\left(\boldsymbol{\mathcal{I}}_{\mathbf{v}} - \boldsymbol{\mathcal{I}}_{\mathbf{v}}\mathbf{C}\mathbf{M}_{L+1,22}\mathbf{C}^{\mathrm{T}}\boldsymbol{\mathcal{I}}_{\mathbf{v}}\right)^{-1}\boldsymbol{\mathcal{I}}_{\mathbf{v}}\mathbf{C}\boldsymbol{\Lambda}_{L+1}^{\mathrm{T}}\,.
\end{aligned}
\qquad \text{(A.31)}
$$

With this identity, the sum in (A.27) can be finally combined:

$$
\begin{aligned}
\Delta\lambda_{L+1} = &\left(\boldsymbol{\mathcal{I}}_{\mathbf{v}}\mathbf{C}\boldsymbol{\Lambda}_{L+1}^{\mathrm{T}}\right)^{\mathrm{T}}\left(\boldsymbol{\mathcal{I}}_{\mathbf{v}} - \boldsymbol{\mathcal{I}}_{\mathbf{v}}\mathbf{C}\mathbf{M}_{L+1,22}\mathbf{C}^{\mathrm{T}}\boldsymbol{\mathcal{I}}_{\mathbf{v}}\right)^{-1}\left(\boldsymbol{\mathcal{I}}_{\mathbf{v}}\mathbf{C}\boldsymbol{\Lambda}_{L+1}^{\mathrm{T}}\right) \\
&- 2\left(\boldsymbol{\mathcal{I}}_{\mathbf{v}}\mathbf{C}\boldsymbol{\Lambda}_{L+1}^{\mathrm{T}}\right)^{\mathrm{T}}\left(\boldsymbol{\varphi}_{L+1}^{\mathrm{T}}\tilde{\boldsymbol{\theta}}\right) \\
&+ \left(\boldsymbol{\varphi}_{L+1}^{\mathrm{T}}\tilde{\boldsymbol{\theta}}\right)^{\mathrm{T}}\left(\boldsymbol{\mathcal{I}}_{\mathbf{v}} - \boldsymbol{\mathcal{I}}_{\mathbf{v}}\mathbf{C}\mathbf{M}_{L+1,22}\mathbf{C}^{\mathrm{T}}\boldsymbol{\mathcal{I}}_{\mathbf{v}}\right)\left(\boldsymbol{\varphi}_{L+1}^{\mathrm{T}}\tilde{\boldsymbol{\theta}}\right) \\
= &\,\mathbf{a}_{L+1}^{\mathrm{T}}\cdot\left(\boldsymbol{\mathcal{I}}_{\mathbf{v}}^{-1} - \mathbf{C}\mathbf{M}_{L+1,22}\mathbf{C}^{\mathrm{T}}\right)\cdot\mathbf{a}_{L+1} \qquad \text{(A.32)}
\end{aligned}
$$

with

$$
\mathbf{a}_{L+1} \coloneqq \boldsymbol{\mathcal{I}}_{\mathbf{v}}\boldsymbol{\varphi}_{L+1}^{\mathrm{T}}\left(\boldsymbol{\theta}_1 - \boldsymbol{\theta}_0\right) - \left(\boldsymbol{\mathcal{I}}_{\mathbf{v}}^{-1} - \mathbf{C}\mathbf{M}_{L+1,22}\mathbf{C}^{\mathrm{T}}\right)^{-1}\mathbf{C}\boldsymbol{\Lambda}_{L+1}^{\mathrm{T}}\,. \qquad \text{(A.33)}
$$

Bibliography

[1] **Agamennoni, G., Nieto, J., and Nebot, E.** *Estimation of Multivehicle Dynamics by Considering Contextual Information*. In: *Robotics, IEEE Transactions on* 28.4 (2012), pp. 855–870.

[2] **Akaike, H.** *Information Theory and an Extension of the Maximum Likelihood Principle*. In: *Selected Papers of Hirotugu Akaike*. 1998, pp. 199–213.

[3] **Altendorfer, R.** *Observable dynamics and coordinate systems for automotive target tracking*. In: *Intelligent Vehicles Symposium (IV), IEEE*. 2009, pp. 741–746.

[4] **Althoff, M., Stursberg, O., and Buss, M.** *Model-Based Probabilistic Collision Detection in Autonomous Driving*. In: *Intelligent Transportation Systems, IEEE Transactions on* 10.2 (2009), pp. 299–310.

[5] **Arulampalam, M., Maskell, S., Gordon, N., and Clapp, T.** *A tutorial on particle filters for online nonlinear/non-Gaussian Bayesian tracking*. In: *Signal Processing, IEEE Transactions on* 50.2 (2002), pp. 174–188.

[6] **Arun, K., Huang, T. S., and Blostein, S. D.** *Least-Squares Fitting of Two 3-D Point Sets*. In: *Pattern Analysis and Machine Intelligence, IEEE Transactions on* PAMI-9.5 (1987), pp. 698–700.

[7] **Axelsson, P., Orguner, U., Gustafsson, F., and Norrlöf, M.** *ML estimation of process noise variance in dynamic systems*. In: *Proceedings of the 18th IFAC World Congress*. 2011, pp. 5609–5614.

[8] **Badino, H.** *Binocular Ego-Motion Estimation for Automotive Applications*. PhD thesis. Goethe Universität Frankfurt am Main, 2008.

[9] **Bansal, A., Badino, H., and Huber, D.** *Understanding how camera configuration and environmental conditions affect appearance-based localization*. In: *Intelligent Vehicles Symposium (IV), IEEE*. 2014, pp. 800–807.

[10] **Bar-Shalom, Y., Li, X. R., and Kirubarajan, T.** *Estimation with applications to tracking and navigation*. Wiley, 2001.

[11] **Bar-Shalom, Y. and Willett, P.** *A Survey of Some Recent Results on the CRLB for Parameter Estimation and Its Extension*. In: *Information Fusion, 19th International Conference on*. 2016.

[12] **Barth, A.** *Vehicle Tracking and Motion Estimation Based on Stereo Vision Sequences*. PhD thesis. Friedrich-Wilhelms-Universität Bonn, 2010.

[13] **Basseville, M. and Nikiforov, I. V.** *Detection of Abrupt Changes: Theory and Application*. Prentice-Hall, Inc., 1993.

[14] **Beinhofer, M.** *Landmark Placement for Mobile Robot Navigation*. PhD thesis. University of Freiburg, Department of Computer Science, 2014.

[15] **Bengler, K., Dietmayer, K., Farber, B., Maurer, M., Stiller, C., and Winner, H.** *Three Decades of Driver Assistance Systems: Review and Future Perspectives*. In: *Intelligent Transportation Systems Magazine, IEEE* 6.4 (2014), pp. 6–22.

[16] **Bergman, N.** *Recursive Bayesian Estimation: Navigation and Tracking Applications*. PhD thesis. Linköping University, Department of Electrical Engineering, 1999.

[17] **Bernini, N., Bertozzi, M., Castangia, L., Patander, M., and Sabbatelli, M.** *Real-time obstacle detection using stereo vision for autonomous ground vehicles: A survey*. In: *Intelligent Transportation Systems (ITSC), 17th IEEE International Conference on*. 2014, pp. 873–878.

[18] **Berthelot, A., Tamke, A., Dang, T., and Breuel, G.** *A novel approach for the probabilistic computation of Time-To-Collision*. In: *Intelligent Vehicles Symposium (IV), IEEE*. 2012, pp. 1173–1178.

[19] **Berthelot, A., Tamke, A., Dang, T., and Breuel, G.** *Stochastic situation assessment in advanced driver assistance system for complex multi-objects traffic situations*. In: *Intelligent Robots and Systems (IROS), IEEE/RSJ International Conference on*. 2012, pp. 1180–1185.

[20] **Bevermeier, M., Peschke, S., and Haeb-Umbach, R.** *Joint Parameter Estimation and Tracking in a Multi-Stage Kalman Filter for Vehicle Positioning*. In: *69th Vehicular Technology Conference (VTC), IEEE*. 2009, pp. 1–5.

[21] **Bier, A. and Luchowski, L.** *Error Analysis of Stereo Calibration and Reconstruction*. In: *Computer Vision/Computer Graphics CollaborationTechniques*. Vol. 5496. 2009, pp. 230–241.

[22] **Birnbaum, Z. W.** *Effect of linear truncation on a multinormal population*. In: *The Annals of Mathematical Statistics* (1950), pp. 272–279.

[23] **Blackman, S. and Popoli, R.** *Design and analysis of modern tracking systems*. Artech House, 1999.

[24] **Blanc, C., Checchin, P., Gidel, S., and Trassoudaine, L.** *Data fusion performance evaluation for range measurements combined with cartesian ones for road obstacle tracking*. In: *Vehicular Electronics and Safety (ICVES), IEEE International Conference on*. 2007, pp. 1–6.

[25] **Blostein, S. and Huang, T.** *Error Analysis in Stereo Determination of 3-D Point Positions*. In: *Pattern Analysis and Machine Intelligence, IEEE Transactions on* PAMI-9.6 (1987), pp. 752–765.

[26] **Box, M. J.** *Bias in Nonlinear Estimation*. In: *Journal of the Royal Statistical Society. Series B (Methodological)* 33.2 (1971), pp. 171–201.

[27] **Braeuchle, C., Flehmig, F., Rosenstiel, W., and Kropf, T.** *Maneuver decision for active pedestrian protection under uncertainty*. In: *Intelligent Transportation Systems (ITSC), 16th International IEEE Conference on*. 2013, pp. 646–651.

[28] **Braeuchle, C., Ruenz, J., Flehmig, F., Rosenstiel, W., and Kropf, T.** *Situation analysis and decision making for active pedestrian protection using Bayesian networks*. In: *6. Tagung Fahrerassistenzsysteme*. 2013.

[29] **Brahmi, M.** *Reference Systems for Environmental Perception*. In: *Automotive Systems Engineering*. 2013, pp. 205–221.

[30] **Brännström, M., Coelingh, E., and Sjoberg, J.** *Model-Based Threat Assessment for Avoiding Arbitrary Vehicle Collisions*. In: *Intelligent Transportation Systems, IEEE Transactions on* 11.3 (2010), pp. 658–669.

[31] **Brännström, M., Sandblom, F., and Hammarstrand, L.** *A Probabilistic Framework for Decision-Making in Collision Avoidance Systems*. In: *Intelligent Transportation Systems, IEEE Transactions on* 14.2 (2013), pp. 637–648.

[32] **Broadhurst, A., Baker, S., and Kanade, T.** *Monte Carlo road safety reasoning*. In: *Intelligent Vehicles Symposium, 2005. Proceedings. IEEE*. 2005, pp. 319–324.

[33] **Brookes, M.** *The Matrix Reference Manual*. Imperial College. 2011. URL: http://www.ee.imperial.ac.uk/hp/staff/dmb/matrix/intro.html.

[34] **Campos, G. de, Runarsson, A., Granum, F., Falcone, P., and Alenljung, K.** *Collision avoidance at intersections: A probabilistic threat-assessment and decision-making system for safety interventions*. In: *Intelligent Transportation Systems (ITSC), 17th IEEE International Conference on*. 2014, pp. 649–654.

[35] **Cao, P., Wachenfeld, W., and Winner, H.** *Perception sensor modeling for virtual validation of automated driving*. In: *it-Information Technology* 57.4 (2015), pp. 243–251.

[36] **Censi, A.** *On achievable accuracy for range-finder localization*. In: *Proceedings of the IEEE International Conference on Robotics and Automation (ICRA)*. 2007, pp. 4170–4175.

[37] **Chang, P., Hirvonen, D., Camus, T., and Southall, B.** *Stereo-Based Object Detection, Classification, and Quantitative Evaluation with Automotive Applications*. In: *Computer Vision and Pattern Recognition (CVPR) - Workshops, IEEE Computer Society Conference on*. 2005, pp. 62–62.

[38] **Cheng, Y., Maimone, M., and Matthies, L.** *Visual odometry on the Mars exploration rovers - a tool to ensure accurate driving and science imaging*. In: *Robotics Automation Magazine, IEEE* 13.2 (2006), pp. 54–62.

[39] **Clark, D. and Ivekovic, S.** *The Cramer-Rao Lower Bound for 3-D state estimation from rectified stereo cameras*. In: *Information Fusion, 13th International Conference on*. 2010.

[40] **Clarke, J. C.** *Modelling uncertainty: A primer*. In: *Tutorial of Department of Eng. Science* (1998).

[41] **Cox, D. R. and Donnelly, C. A.** *Principles of Applied Statistics*. Cambridge Books Online. Cambridge University Press, 2011.

[42] **Dagli, I., Breuel, G., Schittenhelm, H., and Schanz, A.** *Cutting-in vehicle recognition for ACC systems- towards feasible situation analysis methodologies*. In: *Intelligent Vehicles Symposium (IV), IEEE*. 2004, pp. 925–930.

[43] **DasGupta, A.** *Asymptotic Theory of Statistics and Probability*. Springer. 2008.

[44] **Daum, F. E.** *Bounds on performance for multiple target tracking*. In: *Automatic Control, IEEE Transactions on* 35.4 (1990), pp. 443–446.

[45] **Dempster, A. P., Laird, N. M., and Rubin, D. B.** *Maximum likelihood from incomplete data via the EM algorithm*. In: *Journal Of The Royal Statistical Society, Series B.* 39.1 (1977), pp. 1–38.

[46] **Dirndorfer, T., Botsch, M., and Knoll, A.** *Model-Based Analysis of Sensor-Noise in Predictive Passive Safety Algorithms*. In: *Proceedings of the 22nd Enhanced Safety of Vehicles Conference*. 2011.

[47] **Donges, E.** *Fahrerverhaltensmodelle*. In: *Handbuch Fahrerassistenzsysteme*. 2nd ed. 2012, pp. 15–23.

[48] **Dorst, L.** *First order error propagation of the Procrustes method for 3D attitude estimation*. In: *Pattern Analysis and Machine Intelligence, IEEE Transactions on* 27.2 (2005), pp. 221–229.

[49] **Doucet, A., De Freitas, N., and Gordon, N.** *Sequential Monte Carlo methods in practice*. Ed. by **Doucet, A.** Springer, 2001.

[50] **Du Toit, N. and Burdick, J.** *Probabilistic Collision Checking With Chance Constraints*. In: *Robotics, IEEE Transactions on* 27.4 (2011), pp. 809–815.

[51] **Dubbelman, G., Hansen, P., and Browning, B.** *Bias compensation in visual odometry*. In: *Intelligent Robots and Systems (IROS), IEEE/RSJ International Conference on*. 2012, pp. 2828–2835.

[52] **Eggert, D., Lorusso, A., and Fisher, R.** *Estimating 3-D rigid body transformations: a comparison of four major algorithms*. In: *Machine Vision and Applications* 9.5-6 (1997), pp. 272–290.

[53] **Eidehall, A. and Petersson, L.** *Statistical Threat Assessment for General Road Scenes Using Monte Carlo Sampling*. In: *Intelligent Transportation Systems, IEEE Transactions on* 9.1 (2008), pp. 137–147.

[54] **Farboud-Sheshdeh, S., Barfoot, T., and Kwong, R.** *Towards Estimating Bias in Stereo Visual Odometry*. In: *Computer and Robot Vision (CRV), Canadian Conference on*. 2014, pp. 8–15.

[55] **Fessler, J.** *Mean and variance of implicitly defined biased estimators (such as penalized maximum likelihood): applications to tomography*. In: *Image Processing, IEEE Transactions on* 5.3 (1996), pp. 493–506.

[56] **Firl, J.** *Probabilistic Maneuver Recognition in Traffic Scenarios*. PhD thesis. Karlsruhe Institute of Technology, 2014.

[57] **Fischler, M. A. and Bolles, R. C.** *Random Sample Consensus: A Paradigm for Model Fitting with Applications to Image Analysis and Automated Cartography*. In: *Commun. ACM* 24.6 (1981), pp. 381–395.

[58] **Fooladgar, F., Samavi, S., Soroushmehr, S., and Shirani, S.** *Geometrical Analysis of Localization Error in Stereo Vision Systems*. In: *Sensors Journal, IEEE* 13.11 (2013), pp. 4236–4246.

[59] **Franke, U. and Kutzbach, I.** *Fast stereo based object detection for stop & go traffic*. In: *Intelligent Vehicles Symposium (IV), IEEE*. 1996, pp. 339–344.

[60] **Fritsche, C., Orguner, U., Svensson, L., and Gustafsson, F.** *A Survey of Some Recent Results on the CRLB for Parameter Estimation and Its Extension*. In: *Information Fusion, 19th International Conference on*. 2016.

[61] **Gehrig, S., Barth, A., Schneider, N., and Siegemund, J.** *A multi-cue approach for stereo-based object confidence estimation*. In: *Intelligent Robots and Systems (IROS), IEEE/RSJ International Conference on*. 2012, pp. 3055–3060.

[62] **Geiger, A., Lenz, P., and Urtasun, R.** *Are we ready for autonomous driving? The KITTI vision benchmark suite*. In: *Computer Vision and Pattern Recognition (CVPR), IEEE Conference on*. 2012, pp. 3354–3361.

[63] **Gelb, A.**, ed. *Applied optimal estimation*. M.I.T. Press, 1974.

[64] **Gibson, S. and Ninness, B.** *Robust maximum-likelihood estimation of multivariable dynamic systems*. In: *Automatica* 41.10 (2005), pp. 1667–1682.

[65] **Gietelink, O. J.** *Design and validation of advanced driver assistance systems*. PhD thesis. Delft University of Technology, 2007.

[66] **Gindele, T.** *Learning Behavior Models for Interpreting and Predicting Traffic Situations*. PhD thesis. Karlsruhe Institute of Technology, 2014.

[67] **Goodwin, G. and Aguero, J.** *Approximate EM Algorithms for Parameter and State Estimation in Nonlinear Stochastic Models*. In: *Decision and Control and European Control Conference (CDC-ECC), 44th IEEE Conference on*. 2005, pp. 368–373.

[68] **Gruyer, D., Belaroussi, R., and Revilloud, M.** *Accurate lateral positioning from map data and road marking detection*. In: *Expert Systems with Applications* (2015).

[69] **Gustafsson, F. and Hendeby, G.** *Some Relations Between Extended and Unscented Kalman Filters*. In: *Signal Processing, IEEE Transactions on* 60.2 (2012), pp. 545–555.

[70] **Gustafsson, F. and Isaksson, A.** *Best choice of coordinate system for tracking coordinated turns*. In: *Decision and Control, 35th IEEE Conference on*. Vol. 3. 1996, 3145–3150 vol.3.

[71] **Havlak, F. and Campbell, M.** *Discrete and Continuous, Probabilistic Anticipation for Autonomous Robots in Urban Environments*. In: *Robotics, IEEE Transactions on* 30.2 (2014), pp. 461–474.

[72] **Heigele, C., Mielenz, H., Heckel, J., and Schramm, D.** *Accurate and fast localization in unstructured environment based on shape context keypoints*. In: *Information Fusion, 17th International Conference on*. 2014.

[73] **Helmer, T.** *Development of a methodology for the evaluation of active safety using the example of preventive pedestrian protection*. PhD thesis. Technische Universität Berlin, 2014.

[74] **Hendeby, G.** *Performance and Implementation Aspects of Nonlinear Filtering*. PhD thesis. Linköping University, Automatic Control, 2008.

[75] **Henderson, H. V. and Searle, S. R.** *On Deriving the Inverse of a Sum of Matrices*. In: *SIAM Review* 23.1 (1981), pp. 53–60.

[76] **Hermes, C.** *Aktionserkennung und -prädiktion mittels Trajektorienklassifikation*. PhD thesis. Bielefeld University, Applied Informatics, 2012.

[77] **Hernandez, M., Farina, A., and Ristic, B.** *PCRLB for tracking in cluttered environments: measurement sequence conditioning approach*. In: *Aerospace and Electronic Systems, IEEE Transactions on* 42.2 (2006), pp. 680–704.

[78] **Hernandez, M., Kirubarajan, T., and Bar-Shalom, Y.** *Multisensor resource deployment using posterior Cramer-Rao bounds*. In: *Aerospace and Electronic Systems, IEEE Transactions on* 40.2 (2004), pp. 399–416.

[79] **Hernandez, M., Marrs, A., Gordon, N., Maskell, S., and Reed, C.** *Cramer-Rao bounds for nonlinear filtering with measurement origin uncertainty*. In: *Information Fusion, 5th International Conference on*. Vol. 1. 2002, pp. 18–25.

[80] **Herpel, T., Lauer, C., German, R., and Salzberger, J.** *Trade-off between coverage and robustness of automotive environment sensor systems*. In: *Intelligent Sensors, Sensor Networks and Information Processing (ISSNIP), International Conference on*. 2008, pp. 551–556.

[81] **Hillenbrand, J., Kroschel, K., and Schmid, V.** *Situation assessment algorithm for a collision prevention assistant*. In: *Intelligent Vehicles Symposium (IV), IEEE*. 2005, pp. 459–465.

[82] **Hillenbrand, J., Spieker, A., and Kroschel, K.** *A Multilevel Collision Mitigation Approach - Its Situation Assessment, Decision Making, and Performance Tradeoffs*. In: *Intelligent Transportation Systems, IEEE Transactions on* 7.4 (2006), pp. 528–540.

[83] **Hillenbrand, J.** *Fahrerassistenz zur Kollisionsvermeidung*. PhD thesis. Universität Karlsruhe (TH), 2008.

[84] **Hirschmuller, H.** *Accurate and efficient stereo processing by semi-global matching and mutual information*. In: *Computer Vision and Pattern Recognition (CVPR), IEEE Computer Society Conference on*. Vol. 2. 2005, pp. 807–814.

[85] **Hoffmann, J.** *Das Darmstädter Verfahren (EVITA) zum Testen und Bewerten von Frontalkollisionsgegenmaßnahmen*. PhD thesis. Technische Universität Darmstadt, 2008.

[86] **Horn, B. K. P.** *Closed-form solution of absolute orientation using unit quaternions*. In: *Journal of the Optical Society of America A* 4.4 (1987), pp. 629–642.

[87] **Horn, B. K. P., Hilden, H., and Negahdaripour, S.** *Closed-Form Solution of Absolute Orientation using Orthonormal Matrices*. In: *Journal of the Optical Society of America* 5.7 (1988), pp. 1127–1135.

[88] **Horridge, P. and Hernandez, M.** *Performance bounds for angle-only filtering with application to sensor network management*. In: *Information Fusion, 6^{th} Conference on*. Vol. 1. 2003, pp. 695–703.

[89] **Hospach, D., Mueller, S., Bringmann, O., Gerlach, J., and Rosenstiel, W.** *Simulation and evaluation of sensor characteristics in vision based advanced driver assistance systems*. In: *Intelligent Transportation Systems (ITSC), 17^{th} IEEE International Conference on*. 2014, pp. 2610–2615.

[90] **Houenou, A., Bonnifait, P., and Cherfaoui, V.** *Risk Assessment for Collision Avoidance Systems*. In: *Intelligent Transportation Systems (ITSC), 17^{th} IEEE International Conference on*. 2014, pp. 386–391.

[91] **Houenou, A., Bonnifait, P., and Cherfaoui, V. W. Y.** *Vehicle trajectory prediction based on motion model and maneuver recognition*. In: *Intelligent Robots and Systems (IROS), IEEE/RSJ International Conference on*. 2013, pp. 4363–4369.

[92] **Hu, Z. and Uchimura, K.** *U-V-disparity: an efficient algorithm for stereovision based scene analysis*. In: *Intelligent Vehicles Symposium (IV), IEEE*. 2005, pp. 48–54.

[93] **Hue, C., Le Cadre, J.-P., and Perez, P.** *Posterior Cramer-Rao bounds for multi-target tracking*. In: *Aerospace and Electronic Systems, IEEE Transactions on* 42.1 (2006), pp. 37–49.

[94] **Ike, T.** *Evidenzbasierte Analyse interner Konflikte bei der Fusion komplementärer Fahrzeugumfeldbeschreibungen*. PhD thesis. Universität Hannover, 2011.

[95] **Jansson, J.** *Collision Avoidance Theory : with Application to Automotive Collision Mitigation*. PhD thesis. Linköping University, Department of Electrical Engineering, 2005.

[96] **Jaynes, E. T. and Kempthorne, O.** *Foundations of Probability Theory, Statistical Inference, and Statistical Theories of Science*. In: vol. II. 1976. Chap. Confidence Intervals vs Bayesian Intervals, pp. 175–257.

[97] **Julier, S. and Uhlmann, J. K.** *A General Method for Approximating Nonlinear Transformations of Probability Distributions*. Tech. rep. Department of Engineering Science, University of Oxford, 1996.

[98] **Käfer, E., Hermes, C., Wöhler, C., Ritter, H., and Kummert, F.** *Recognition of situation classes at road intersections*. In: *Robotics and Automation (ICRA), IEEE International Conference on*. 2010, pp. 3960–3965.

[99] **Kalman, R. E.** *A New Approach to Linear Filtering and Prediction Problems*. In: *Transactions of the ASME–Journal of Basic Engineering* 82.Series D (1960), pp. 35–45.

[100] **Kämpchen, N., Schiele, B., and Dietmayer, K.** *Situation Assessment of an Autonomous Emergency Brake for Arbitrary Vehicle-to-Vehicle Collision Scenarios*. In: *Intelligent Transportation Systems, IEEE Transactions on* 10.4 (2009), pp. 678–687.

[101] **Kämpchen, N.** *Feature-Level Fusion of Laser Scanner and Video Data for Advanced Driver Assistance Systems*. PhD thesis. Ulm University, 2007.

[102] **Kapp, A.** *Quality Measures for Lidar Signal Procoessing*. In: *Intelligent Vehicles Symposium, 2006 IEEE*. 2006, pp. 163–168.

[103] **Kapp, A.** *Ein Beitrag zur Verbesserung und Erweiterung der Lidar-Signalverarbeitung für Fahrzeuge*. PhD thesis. Universität Karlsruhe (TH), 2007.

[104] **Karlsson, R., Jansson, J., and Gustafsson, F.** *Model-based statistical tracking and decision making for collision avoidance application*. In: *American Control Conference*. Vol. 4. 2004, pp. 3435–3440.

[105] **Kasper, D., Weidl, G., Dang, T., Breuel, G., Tamke, A., Wedel, A., and Rosenstiel, W.** *Object-Oriented Bayesian Networks for Detection of Lane Change Maneuvers*. In: *Intelligent Transportation Systems Magazine, IEEE* 4.3 (2012), pp. 19–31.

[106] **Kay, S. M.** *Fundamentals of Statistical Signal Processing, Volume 2: Detection Theory*. Prentice-Hall Inc, 1993.

[107] **Kelly, A.** *Linearized Error Propagation in Odometry*. In: *The International Journal of Robotics Research* 23.2 (2004), pp. 179–218.

[108] **Kiencke, U., Schwarz, M., and Weickert, T.** *Signalverarbeitung: Zeit-Frequenz-Analyse und Schätzverfahren*. Oldenbourg Wissenschaftsverlag, 2008.

[109] **Kim, T. and Jeong, H.-Y.** *A Novel Algorithm for Crash Detection Under General Road Scenes Using Crash Probabilities and an Interactive Multiple Model Particle Filter*. In: *Intelligent Transportation Systems, IEEE Transactions on* 15.6 (2014), pp. 2480–2490.

[110] **Knuth, J. and Barooah, P.** *Error growth in position estimation from noisy relative pose measurements*. In: *Robotics and Autonomous Systems* 61.3 (2013), pp. 229–244.

[111] **Kutzera, C.** *Eine Methode zur Bewertung von umfelderfassenden Sensoren und Fahrerassistenzsystemen*. PhD thesis. Universität Erlangen-Nürnberg, 2011.

[112] **Labayrade, R., Aubert, D., and Tarel, J.-P.** *Real time obstacle detection in stereovision on non flat road geometry through "v-disparity" representation*. In: *Intelligent Vehicles Symposium (IV), IEEE*. Vol. 2. 2002, pp. 646–651.

[113] **Lages, U., Spencer, M., and Katz, R.** *Automatic Scenario Generation Based on Laserscanner Reference Data and Advanced Offline Processing*. In: *Intelligent Vehicles Symposium (IV), IEEE*. 2013, pp. 153–155.

[114] **Lambert, A., Gruyer, D., Pierre, G., and Ndjeng, A.** *Collision Probability Assessment for Speed Control*. In: *Intelligent Transportation Systems (ITSC), 11th IEEE International Conference on*. 2008, pp. 1043–1048.

[115] **Lategahn, H. and Stiller, C.** *Vision-Only Localization*. In: *Intelligent Transportation Systems, IEEE Transactions on* 15.3 (2014), pp. 1246–1257.

[116] **Le Marchand, O., Bonnifait, P., Ibanez-Guzmán, J., Betaille, D., and Peyret, F.** *Characterization of GPS multipath for passenger vehicles across urban environments*. In: *ATTI dell'Istituto Italiano di Navigazione* 189 (2009), pp. 77–88.

[117] **Lefèvre, S., Laugier, C., and Ibanez-Guzman, J.** *Risk assessment at road intersections: Comparing intention and expectation*. In: *Intelligent Vehicles Symposium (IV), IEEE*. 2012, pp. 165–171.

[118] **Lefèvre, S., Vasquez, D., and Laugier, C.** *A survey on motion prediction and risk assessment for intelligent vehicles*. In: *ROBOMECH Journal* 1.1, 1 (2014).

[119] **Lei, M., Han, C., and Liu, P.** *Expectation Maximization (EM) algorithm-based nonlinear target tracking with adaptive state transition matrix and noise covariance*. In: *Information Fusion, 10th International Conference on*. 2007.

[120] **Lerro, D. and Bar-Shalom, Y.** *Tracking with debiased consistent converted measurements versus EKF*. In: *Aerospace and Electronic Systems, IEEE Transactions on* 29.3 (1993), pp. 1015–1022.

[121] **Leven, W. F.** *Approximate Cramer-Rao Bounds for Multiple Target Tracking*. PhD thesis. Georgia Institute of Technology, 2006.

[122] **Levy, B. C.** *Principles of Signal Detection and Parameter Estimation*. 1st ed. Springer Publishing Company, Inc., 2008.

[123] **Li, X. and Jilkov, V.** *A Survey of Maneuvering Target Tracking - Part III: Measurement Models*. In: *SPIE Conf. on Signal and Data Processing of Small Targets*. 2001, pp. 423–446.

[124] **Li, X. and Jilkov, V.** *Survey of maneuvering target tracking. Part I. Dynamic models*. In: *Aerospace and Electronic Systems, IEEE Transactions on* 39.4 (2003), pp. 1333–1364.

[125] **Liebner, M., Klanner, F., Baumann, M., Ruhhammer, C., and Stiller, C.** *Velocity-Based Driver Intent Inference at Urban Intersections in the Presence of Preceding Vehicles*. In: *Intelligent Transportation Systems Magazine, IEEE* 5.2 (2013), pp. 10–21.

[126] **Lin, C.-Y. and Tomizuka, M.** *Probabilistic Approach to Modeling and Parameter Learning of Indirect Drive Robots From Incomplete Data*. In: *Mechatronics, IEEE/ASME Transactions on* PP.99 (2014), pp. 1–10.

[127] **Lu, F. and Milios, E.** *Robot pose estimation in unknown environments by matching 2D range scans*. In: *Computer Vision and Pattern Recognition (CVPR), IEEE Computer Society Conference on*. 1994, pp. 935–938.

[128] **Lu, N. and Zimmermann, D. L.** *On Likelihood-based Inference for a separable covariance matrix*. Tech. rep. University of Iowa, 2004.

[129] **Ma, X. and Andreasson, I.** *Behavior measurement, analysis, and regime classification in car following*. In: *Intelligent Transportation Systems, IEEE Transactions on* 8.1 (2007), pp. 144–156.

[130] **Mader, W., Linke, Y., Mader, M., Sommerlade, L., Timmer, J., and Schelterom, B.** *A numerically efficient implementation of the expectation maximization algorithm for state space models*. In: *Applied Mathematics and Computation* 241 (2014), pp. 222–232.

[131] **Magnus, J. R.** *Matrix differential calculus with applications in statistics and econometrics*. Ed. by **Neudecker, H.** Rev. ed., repr. John Wiley, 2002.

[132] **Manjunath, B. and Wilhelm, S.** *Moments Calculation for the Double Truncated Multivariate Normal Density*. In: *SSRN Working Paper Series* (2010).

[133] **Maskell, S.** *Sequentially Structured Bayesian Solutions*. PhD thesis. Cambridge University, Engineering Department, 2004.

[134] **Mathai, A. and Provost, S.** *Quadratic Forms in Random Variables*. Taylor & Francis, 1992.

[135] **Matthies, L. and Grandjean, P.** *Stochastic performance, modeling and evaluation of obstacle detectability with imaging range sensors*. In: *Robotics and Automation, IEEE Transactions on* 10.6 (1994), pp. 783–792.

[136] **Matthies, L. and Shafer, S.** *Error modeling in stereo navigation*. In: *Robotics and Automation, IEEE Journal of* 3.3 (1987), pp. 239–248.

[137] **Maurer, T.** *Bewertung von Mess- und Prädiktionsunsicherheiten in der zeitlichen Eingriffsentscheidung für automatische Notbrems- und Ausweichsysteme*. PhD thesis. Universität Duisburg-Essen, 2013.

[138] **Meng, H., Hernandez, M., Liu, Y., and Wang, X.** *Computationally efficient PCRLB for tracking in cluttered environments: measurement existence conditioning approach*. In: *Signal Processing, IET* 3.2 (2009), pp. 133–149.

[139] **Meyer-Delius, D., Plagemann, C., and Burgard, W.** *Probabilistic situation recognition for vehicular traffic scenarios*. In: *Robotics and Automation (ICRA), IEEE International Conference on*. 2009, pp. 459–464.

[140] **Mirzaei, F., Mourikis, A., and Roumeliotis, S.** *On the Performance of Multi-robot Target Tracking*. In: *Robotics and Automation (ICRA), IEEE International Conference on*. 2007, pp. 3482–3489.

[141] **Muntzinger, M. M.** *Zustandsschätzung mit chronologisch ungeordneten Sensordaten für die Fahrzeugumfelderfassung*. PhD thesis. Universität Ulm, 2011.

[142] **Murphy, K. P.** *Dynamic Bayesian Networks: Representation, Inference and Learning*. PhD thesis. University of California, Berkeley, 2002.

[143] **Neumann-Cosel, K. von**. *Virtual Test Drive: Simulation umfeldbasierter Fahrzeugfunktionen*. Dissertation. Technische Universität München, 2014.

[144] **Nilsson, J., Ödblom, A., and Fredriksson, J.** *Worst Case Analysis of Automotive Collision Avoidance Systems*. In: *Vehicular Technology, IEEE Transactions on* 65.4 (2016), pp. 1899–1911.

[145] **Nordlund, P.-J. and Gustafsson, F.** *Probabilistic Conflict Detection for Piecewise Straight Paths*. Tech. rep. 2871. Linköping University, Automatic Control, 2008.

[146] **Ohta, N. and Kanatani, K.** *Optimal estimation of three-dimensional rotation and reliability evaluation*. In: *Computer Vision – ECCV*. Vol. 1406. 1998, pp. 175–187.

[147] **Olson, C. F., Matthies, L. H., Schoppers, M., and Maimone, M. W.** *Rover navigation using stereo ego-motion*. In: *Robotics and Autonomous Systems* 43.4 (2003), pp. 215–229.

[148] **Otto, C.** *Fusion of Data from Heterogeneous Sensors with Distributed Fields of View and Situation Evaluation for Advanced Driver Assistance Systems*. PhD thesis. Karlsruhe Institute of Technology, 2013.

[149] **Pandey, M.** *An effective approximation to evaluate multinormal integrals*. In: *Structural Safety* 20.1 (1998), pp. 51–67.

[150] **Patil, S., Berg, J. van den, and Alterovitz, R.** *Estimating probability of collision for safe motion planning under Gaussian motion and sensing uncertainty*. In: *Robotics and Automation (ICRA), IEEE International Conference on*. 2012, pp. 3238–3244.

[151] **Petrich, D., Dang, T., Kasper, D., Breuel, G., and Stiller, C.** *Map-based long term motion prediction for vehicles in traffic environments*. In: *Intelligent Transportation Systems (ITSC), 16th IEEE International Conference on*. 2013, pp. 2166–2172.

[152] **Pfeiffer, D.** *The Stixel World*. PhD thesis. Humboldt-Universität zu Berlin, 2012.

[153] **Pietzsch, S.** *Modellgestützte Sensordatenfusion von Laserscanner und Radar zur Erfassung komplexer Fahrzeugumgebungen*. PhD thesis. Technische Universität München, 2015.

[154] **Pinggera, P., Pfeiffer, D., Franke, U., and Mester, R.** *Know Your Limits: Accuracy of Long Range Stereoscopic Object Measurements in Practice*. In: *Computer Vision – ECCV*. Vol. 8690. 2014, pp. 96–111.

[155] **Pink, O.** *Bildbasierte Selbstlokalisierung von Straßenfahrzeugen*. PhD thesis. Karlsruher Institut für Technologie, 2011.

[156] **Ponte Müller, F. de, Navajas, L. M., and Strang, T.** *Characterization of a Laser Scanner Sensor for the Use as a Reference System in Vehicular Relative Positioning*. In: *Communication Technologies for Vehicles*. Vol. 7865. 2013, pp. 146–158.

[157] **Puente León, F. and Kiencke, U.** *Messtechnik : Systemtheorie für Ingenieure und Informatiker*. Springer Vieweg, 2012.

[158] **Qu, X., Soheilian, B., and Paparoditis, N.** *Vehicle localization using mono-camera and geo-referenced traffic signs*. In: *Intelligent Vehicles Symposium (IV), IEEE*. 2015, pp. 605–610.

[159] **Ristic, B. and Morelande, M.** *Comments on 'Cramér-Rao lower bound for tracking multiple targets'*. In: *IET Radar, Sonar Navigation* 1.1 (2007), pp. 74–76.

[160] **Robert, C. P.** *The Bayesian Choice: From Decision-Theoretic Foundations to Computational Implementation*. 2nd. Springer, 2007.

[161] **Roth, M., Hendeby, G., and Gustafsson, F.** *EKF/UKF maneuvering target tracking using coordinated turn models with polar/Cartesian velocity*. In: *Information Fusion (FUSION), 17th International Conference on*. 2014.

[162] **Roth, M.** *Kalman filters for nonlinear systems and heavy-tailed noise*. In: *Licentiate thesis, Linköping University* (2013).

[163] **Rydstrom, M., Strom, E., Svensson, A., and Urruela, A.** *Practical automotive applications of Cramer-Rao bound analysis*. In: *Intelligent Vehicles Symposium (IV), IEEE*. 2005, pp. 61–66.

[164] **Sabater, N., Morel, J.-M., and Almansa, A.** *How Accurate Can Block Matches Be in Stereo Vision?* In: *SIAM Journal on Imaging Sciences* 4.1 (2011), pp. 472–500.

[165] **Sahai, H. and Ojeda, M. M.** *A comparison of approximations to percentiles of the noncentral χ^2-distribution*. In: *Revista de Matemática: Teoría y Aplicaciones* 10.1-2 (2003), pp. 57–76.

[166] **Särkkä, S.** *Bayesian filtering and smoothing*. Cambridge University Press, 2013.

[167] **Schlechtriemen, J., Wedel, A., Breuel, G., and Kuhnert, K.-D.** *A probabilistic long term prediction approach for highway scenarios*. In: *Intelligent Transportation Systems (ITSC), 17th IEEE International Conference on*. 2014, pp. 732–738.

[168] **Schlosser, M. and Kroschel, K.** *Limits in tracking with extended Kalman filters*. In: *Aerospace and Electronic Systems, IEEE Transactions on* 40.4 (2004), pp. 1351–1359.

[169] **Schmidt, C.** *Fahrstrategien zur Unfallvermeidung im Straßenverkehr für Einzel- und Mehrobjektszenarien*. PhD thesis. Karlsruher Institut für Technologie, 2013.

[170] **Schmidt, F.** *Funktionale Absicherung kamerabasierter Aktiver Fahrerassistenzsysteme durch Hardware-in-the-Loop-Tests*. PhD thesis. Universität Kaiserslautern, 2012.

[171] **Schneider, J., Wilde, A., and Naab, K.** *Probabilistic approach for modeling and identifying driving situations*. In: *Intelligent Vehicles Symposium (IV), IEEE*. 2008, pp. 343–348.

[172] **Schön, T. B., Wills, A., and Ninness, B.** *System identification of nonlinear state-space models*. In: *Automatica* 47.1 (2011), pp. 39–49.

[173] **Schreier, M., Willert, V., and Adamy, J.** *An Integrated Approach to Maneuver-Based Trajectory Prediction and Criticality Assessment in Arbitrary Road Environments*. In: *Intelligent Transportation Systems, IEEE Transactions on* (2016).

[174] **Schubert, R., Klöden, H., Wanielik, G., and Kälberer, S.** *Performance evaluation of Multiple Target Tracking in the absence of reference data*. In: *Information Fusion, 13th International Conference on*. 2010.

[175] **Schubert, R., Adam, C., Obst, M., Mattern, N., Leonhardt, V., and Wanielik, G.** *Empirical evaluation of vehicular models for ego motion estimation*. In: *Intelligent Vehicles Symposium (IV), IEEE*. 2011, pp. 534–539.

[176] **Schwarz, G.** *Estimating the Dimension of a Model*. In: *Ann. Statist.* 6.2 (1978), pp. 461–464.

[177] **Se, S. and Brady, M.** *Ground plane estimation, error analysis and applications*. In: *Robotics and Autonomous Systems* 39.2 (2002), pp. 59–71.

[178] **Sengupta, D. and Kay, S.** *Parameter estimation and GLRT detection in colored non-Gaussian autoregressive processes*. In: *Acoustics, Speech and Signal Processing, IEEE Transactions on* 38.10 (1990), pp. 1661–1676.

[179] **Sibley, G., Matthies, L., and Sukhatme, G.** *Bias Reduction and Filter Convergence for Long Range Stereo*. In: *Robotics Research*. Vol. 28. 2007, pp. 285–294.

[180] **Sörstedt, J., Svensson, L., Sandblom, F., and Hammarstrand, L.** *A New Vehicle Motion Model for Improved Predictions and Situation Assessment*. In: *Intelligent Transportation Systems, IEEE Transactions on* 12.4 (2011), pp. 1209–1219.

[181] **Taylor, J.** *The Cramer-Rao estimation error lower bound computation for deterministic nonlinear systems*. In: *Decision and Control including the 17th Symposium on Adaptive Processes, IEEE Conference on*. 1978, pp. 1178–1181.

[182] **Taylor, R., Flanagan, B., and Uber, J.** *Computing the recursive posterior Cramer-Rao bound for a nonlinear nonstationary system*. In: *Acoustics, Speech, and Signal Processing (ICASSP), IEEE International Conference on*. Vol. 6. 2003, pp. 673–676.

[183] **Tellmann, D.** *Hardware-in-the-loop gestützte Entwicklungsplattform für Fahrassistenzsysteme - Modelle der Umfeldsensorik und angepasste Fahrermodelle*. PhD thesis. Universität Kassel, 2011.

[184] **Tempo, R., Calafiore, G., and Dabbene, F.** *Randomized Algorithms for Analysis and Control of Uncertain Systems: With Applications*. 2nd ed. Springer Publishing Company, Inc., 2012.

[185] **Terejanu, G., Singla, P., Singh, T., and Scott, P. D.** *Uncertainty Propagation for Nonlinear Dynamical Systems using Gaussian Mixture Models*. In: *Journal of Guidance, Control, and Dynamics* 31.6 (2008), pp. 1622–1633.

[186] **Themann, P., Kotte, J., Raudszus, D., and Eckstein, L.** *Impact of positioning uncertainty of vulnerable road users on risk minimization in collision avoidance systems*. In: *Intelligent Vehicles Symposium (IV), IEEE*. 2015, pp. 1201–1206.

[187] **Thrun, S., Burgard, W., and Fox, D.** *Probabilistic robotics*. MIT Press, 2005.

[188] **Tichavsky, P., Muravchik, C., and Nehorai, A.** *Posterior Cramer-Rao bounds for discrete-time nonlinear filtering*. In: *Signal Processing, IEEE Transactions on* 46.5 (1998), pp. 1386–1396.

[189] **Tiemann, N.** *Ein Beitrag zur Situationsanalyse im vorausschauenden Fußgängerschutz*. PhD thesis. Universität Duisburg-Essen, 2012.

[190] **Van Trees, H. L.** *Detection, estimation, and modulation theory*. Vol. 1: Detection, estimation, and linear modulation theory. Wiley, 1968.

[191] **Van Zanten, A. T.** *Bosch ESP systems: 5 years of experience*. Tech. rep. 2000-01-1633. SAE Technical Paper, 2000.

[192] **Vasquez, D. and Fraichard, T.** *Motion prediction for moving objects: a statistical approach*. In: *Robotics and Automation (ICRA), IEEE International Conference on*. Vol. 4. 2004, 3931–3936 Vol.4.

[193] **Vignat, C. and Bercher, J.-F.** *On Fisher information inequalities and score functions in non-invertible linear systems*. In: *Journal of Inequalities in Pure and Applied Mathematics* 4.4 (2003), pp. 1–9.

[194] **Wang, D. Z., Posner, I., and Newman, P.** *Model-Free Detection and Tracking of Dynamic Objects with 2D Lidar*. In: *The International Journal of Robotics Research (IJRR)* 34.7 (2015), pp. 1039–1063.

[195] **Welzel, A., Reisdorf, P., and Wanielik, G.** *Improving Urban Vehicle Localization with Traffic Sign Recognition*. In: *Intelligent Transportation Systems (ITSC), 18th IEEE International Conference on*. 2015, pp. 2728–2732.

[196] **Wendel, J.** *Integrierte Navigationssysteme : Sensordatenfusion, GPS und inertiale Navigation*. 2nd ed. Oldenbourg, 2011.

[197] **Weng, J., Cohen, P., and Rebibo, N.** *Motion and structure estimation from stereo image sequences*. In: *Robotics and Automation, IEEE Transactions on* 8.3 (1992), pp. 362–382.

[198] **Werner, K., Jansson, M., and Stoica, P.** *On Estimation of Covariance Matrices With Kronecker Product Structure*. In: *Signal Processing, IEEE Transactions on* 56.2 (2008), pp. 478–491.

[199] **Wiest, J., Karg, M., Kunz, F., Reuter, S., Kressel, U., and Dietmayer, K.** *A Probabilistic Maneuver Prediction Framework for Self-Learning Vehicles with Application to Intersections*. In: *Intelligent Vehicles Symposium (IV), IEEE*. 2015, pp. 349–355.

[200] **Wiest, J., Hoffken, M., Kresel, U., and Dietmayer, K.** *Probabilistic trajectory prediction with Gaussian mixture models*. In: *Intelligent Vehicles Symposium (IV), IEEE*. 2012, pp. 141–146.

[201] **Willsky, A. and Jones, H.** *A generalized likelihood ratio approach to the detection and estimation of jumps in linear systems*. In: *Automatic Control, IEEE Transactions on* 21.1 (1976), pp. 108–112.

[202] **Yuan, X., Lian, F., and Han, C.** *Models and Algorithms for Tracking Target with Coordinated Turn Motion*. In: *Mathematical Problems in Engineering* (2014), pp. 1–10.

[203] **Yuan, X.-X. and Pandey, M.** *Analysis of approximations for multinormal integration in system reliability computation*. In: *Structural Safety* 28.4 (2006), pp. 361–377.

[204] **Zhang, F., Simon, C., Chen, G., Buckl, C., and Knoll, A.** *Cumulative Error Estimation from Noisy Relative Measurements*. In: *Intelligent Transportation Systems (ITSC), 16th IEEE International Conference on*. 2013.

[205] **Zhang, P., Gu, J., and Milios, E.** *Registration uncertainty for robot self-localization in 3D*. In: *Computer and Robot Vision, 2nd Canadian Conference on*. 2005, pp. 490–497.

[206] **Zhang, T. and Boult, T.** *Realistic stereo error models and finite optimal stereo baselines*. In: *Applications of Computer Vision (WACV), IEEE Workshop on*. 2011, pp. 426–433.

[207] **Zheng, P. and McDonald, M.** *The effect of sensor errors on the performance of collision warning systems*. In: *Intelligent Transportation Systems (ITSC), IEEE International Conference on*. 2003, pp. 469–474.

[208] **Ziegler, J., Lategahn, H., Schreiber, M., Keller, C., Knoppel, C., Hipp, J., Haueis, M., and Stiller, C.** *Video based localization for Bertha*. In: *Intelligent Vehicles Symposium (IV), IEEE*. 2014, pp. 1231–1238.

[209] **Ziehn, J., Ruf, M., Rosenhahn, B., Willersinn, D., Beyerer, J., and Gotzig, H.** *Correspondence between Variational Methods and Hidden Markov Models*. In: *Intelligent Vehicles Symposium (IV), IEEE*. 2015, pp. 380–385.

[210] **Zofka, M., Kuhnt, F., Kohlhaas, R., Rist, C., Schamm, T., and Zollner, J.** *Data-driven simulation and parametrization of traffic scenarios for the development of advanced driver assistance systems*. In: *Information Fusion, 18th International Conference on*. 2015, pp. 1422–1428.

List of publications

[211] **Rohde, J., Stellet, J. E., Mielenz, H., and Zöllner, J. M.** *Model-Based Derivation of Perception Accuracy Requirements for Vehicle Localization in Urban Environments*. In: *Intelligent Transportation Systems (ITSC), 18th IEEE International Conference on*. 2015, pp. 712–718.

[212] **Rohde, J., Stellet, J. E., Mielenz, H., and Zöllner, J. M.** *Localization Accuracy Estimation with Application to Perception Design*. In: *Robotics and Automation (ICRA), IEEE International Conference on*. 2016, pp. 4777–4783.

[213] **Stellet, J. E., Kutzera, C., and Weigel, R.** *Optimale Geradenschätzung für Time-of-Flight Umfeldsensoren*. In: *tm-Technisches Messen* 78.6 (2011), pp. 286–291.

[214] **Stellet, J. E. and Rogg, T.** *On linear observers and application to fault detection in synchronous generators*. In: *Control Theory and Technology* 12.4 (2014), pp. 345–356.

[215] **Stellet, J. E., Walkling, L., and Zöllner, J. M.** *Post Processing of Laser Scanner Measurements for Testing Advanced Driver Assistance Systems*. In: *Information Fusion, 19th International Conference on*. 2016.

[216] **Stellet, J. E., Suchaneck, A., Gießler, M., León, F. P., and Gauterin, F.** *Fahrbahnreibwertschätzung mit optimaler linearer Parametrierung*. In: *at - Automatisierungstechnik* 62.8 (2014), pp. 570–581.

[217] **Stellet, J. E., Heigele, C., Kuhnt, F., Zöllner, J. M., and Schramm, D.** *Performance evaluation and statistical analysis of algorithms for ego-motion estimation*. In: *Intelligent Transportation Systems (ITSC), 17th IEEE International Conference on*. 2014, pp. 2125–2131.

[218] **Stellet, J. E., Straub, F., Schumacher, J., Branz, W., and Zöllner, J. M.** *Estimating the Process Noise Variance for Vehicle Motion Models*. In: *Intelligent Transportation Systems (ITSC), 18th IEEE International Conference on*. 2015, pp. 1512–1519.

[219] **Stellet, J. E., Schumacher, J., Branz, W., and Zöllner, J. M.** *Performance Bounds on Change Detection with Application to Manoeuvre Recognition for Advanced Driver Assistance Systems*. In: *Intelligent Vehicles Symposium (IV), IEEE*. 2015, pp. 1112–1119.

[220] **Stellet, J. E., Zofka, M. R., Schumacher, J., Schamm, T., and Zöllner, J. M.** *Testing of Advanced Driver Assistance Towards Automated Driving: A Survey and Taxonomy on Existing Approaches and Open Questions*. In: *Intelligent Transportation Systems (ITSC), 18th IEEE International Conference on*. 2015, pp. 1455–1462.

[221] **Stellet, J. E., Schumacher, J., Branz, W., and Zöllner, J. M.** *Uncertainty propagation in criticality measures for driver assistance*. In: *Intelligent Vehicles Symposium (IV), IEEE*. 2015, pp. 1187–1194.

[222] **Stellet, J. E., Vogt, P., Schumacher, J., Branz, W., and Zöllner, J. M.** *Analytical derivation of performance bounds of autonomous emergency brake systems*. In: *Intelligent Vehicles Symposium (IV), IEEE*. 2016, pp. 220–226.

[223] **Stellet, J. E., Schumacher, J., Lange, O., Branz, W., Niewels, F., and Zöllner, J. M.** *Statistical Modelling of Object Detection in Stereo Vision-Based Driver Assistance*. In: *Intelligent Autonomous Systems 13*. Vol. 302. 2016, pp. 749–761.

List of supervised theses

[224] **Hoch, M.** *Mehrmodellbasierte Bewegungsschätzung mit Laserscannern*. Bachelor thesis. Hochschule Karlsruhe – Technik und Wirtschaft, 2015.

[225] **Straub, F.** *Bestimmung von Prädiktionsunsicherheiten von dynamischen Bewegungsmodellen*. Bachelor thesis. Hochschule Karlsruhe – Technik und Wirtschaft, 2014.

[226] **Walkling, R. L.** *Offline-Signalverarbeitung für Laserscanner zur Objektverfolgung und Bewegungsschätzung*. Master thesis. Karlsruher Institut für Technologie, 2015.

List of Figures

List of Tables