



TECHNISCHE
UNIVERSITÄT
WIEN

VIENNA
UNIVERSITY OF
TECHNOLOGY

DISSERTATION

Time-Triggered Object Tracking for Advanced Driver Assistance Systems

ausgeführt zum Zwecke der Erlangung des akademischen Grades
eines Doktors der technischen Wissenschaften

unter der Leitung von

Privatdoz. Dipl.-Ing. Dr.techn. Wilfried Elmenreich

eingereicht an der Fakultät für Informatik
der Technischen Universität Wien

von

Moritz Koplín (geb. Mauthner)
Matrikelnummer: 0427797
Max-Lebsche-Platz 41
D-81377 München

Wien, im Mai 2009

Abstract

The development of advanced driver assistance systems based on multi-sensor object tracking is an increasingly significant topic in the automotive industry. An important factor in achieving the required level of system dependability is the system's capability to maintain real-time images of the states of all relevant objects within a well-defined range of the surrounding environment.

Most state-of-the-art systems cannot guarantee that the real-time images of the object states can be updated within a well-defined accuracy interval, as the object state observations are sampled by uncontrolled sensors and transmitted over an indeterministic controller area network bus system.

To overcome this shortcoming this thesis proposes a paradigm shift toward time-triggered advanced driver assistance systems based on multi-sensor object tracking, in which a time-triggered deterministic bus system establishes a global time-base and synchronizes the clocks of all nodes.

In order to prove the feasibility of this paradigm shift, models of a state-of-the-art and a time-triggered advanced driver assistance system based on multi-sensor object tracking are being developed and compared with regard to their performance.

The models consist of two sensors, an object tracking subsystem and a feature service subsystem which are interconnected via a bus system. The sensors sample object state observations of known accuracy which are transmitted over the bus system to the object tracking subsystem. Thereon, the object state observations are processed by a Kalman filter based algorithm and provided to the feature service subsystem, which requires real-time images of all relevant object states at predefined points in time.

The results show that the state-of-the-art model is generally advantageous in scenarios with low process noise but is outmatched by the time-triggered model for increasing process noise. Thus for linearized state-space models and for potentially dangerous scenarios with high dynamics in state space parameter derivatives which are not modeled, the time-triggered model becomes advantageous which promotes the paradigm shift toward time-triggered advanced driver assistance systems based on multi-sensor object tracking.

In order to provide a case study for the time-triggered model, an advanced driver assistance system based on multi-sensor object tracking using FlexRay as the time-triggered backbone of an automotive real-time architecture has been developed. This approach has been evaluated by using realistic data from field tests where vehicle trajectories were provided by a differential GPS system.

Zusammenfassung

Die Entwicklung zeitgesteuerter Fahrerassistenzsysteme der neuen Generation zur Umfelderkennung mittels mehrerer Sensoren gewinnt in der Automobilindustrie zunehmend an Bedeutung. Ein wichtiger Faktor bei der Erreichung der erforderlichen Zuverlässigkeit des Systems ist die Fähigkeit des Systems Echtzeit-Bilder der Zustände aller relevanten Objekte innerhalb eines wohldefinierten Bereiches in der Umgebung aufrecht zu erhalten.

Jedoch können die meisten dem Stand der Technik entsprechenden Systeme nicht garantieren, dass die Echtzeit-Bilder der Objektzustände innerhalb eines genau definierten Intervalls aktualisiert werden, da die Beobachtung eines Objektzustandes durch unsynchronisierte Sensoren vorgenommen wird und über ein nichtdeterministisches Controller Area Network Bus-System übertragen wird.

Um diesen Mangel zu beheben, wird in dieser Arbeit ein Paradigmenwechsel in Richtung zeitgesteuerter Fahrerassistenzsysteme vorgeschlagen, in denen ein zeitgesteuertes deterministisches Bus-System eine globale Zeit bereitstellt und die Uhren aller Knoten synchronisiert.

Um die Durchführbarkeit dieses Paradigmenwechsels zu beweisen, werden Modelle eines dem Stand der Technik entsprechenden und eines zeitgesteuerten Fahrerassistenzsystems entwickelt und bezüglich ihrer Leistung verglichen.

Die Modelle bestehen aus zwei Sensoren, einem Objektverfolgungs-Teilsystem und einem Diensterbringungs-Teilsystem, die über ein Bus-System verbunden sind. Die Sensoren liefern Objektbeobachtungen bekannter Genauigkeit, die über das Bus-System zum Objektverfolgungs-Teilsystem gesendet werden. Dort wird die Beobachtung des Objektzustandes durch einen Kalman-Filter Algorithmus verarbeitet.

Die Ergebnisse zeigen, dass das aktuell verwendete Modell in Szenarien mit geringem Prozessrauschen vorteilhaft ist, aber bei höherem Prozessrauschen von dem zeitgesteuerten Modell übertroffen wird. Daher ist für linearisierte Zustands-Räume und für potenziell gefährliche Szenarien mit hoher Dynamik in den nicht modellierten Zustands-Ableitungen, das zeitgesteuerte Modell von Vorteil, was den Paradigmenwechsel in Richtung zeitgesteuerter Fahrerassistenzsysteme nahelegt.

Für die Evaluierung des Ansatzes wurde ein zeitgesteuertes Fahrerassistenzsystem zur Objektverfolgung entwickelt, welches mehrere Sensoren beinhaltet, die durch das Echtzeit-Protokoll Flexray vernetzt sind. Zur Analyse wurden Fahrdaten, welche durch ein Differential-GPS-System gewonnen wurden, verwendet.

Acknowledgments

Diese Arbeit entstand während meiner dreijährigen Doktorandenzeit in der Elektronikforschung der Volkswagen AG.

Mein besonderer Dank gilt meinem Betreuer PD Dr. Wilfried Elmenreich für die vielen Diskussionen und Hilfestellungen während der Anfertigung der Doktorarbeit. Ferner gilt mein Dank Prof. Dr. Hermann Rohling von der TU Hamburg-Harburg für die Übernahme des Koreferats.

Danken möchte ich auch Dr. Dirk Stüker, Jens Ende, Thomas Engel, Dr. Holger Philipps und Dr. Kristian Weiss für ihre hilfreiche Unterstützung während der Entstehung der Doktorarbeit.

Contents

1. Introduction	1
1.1. Background	1
1.2. Problem Statement	1
1.3. Outline	4
2. Object Tracking	5
2.1. Sensor Fusion	6
2.1.1. Objective of Sensor Fusion	6
2.1.2. Concept of Sensor Fusion	6
2.1.3. Time-Triggered Sensor Fusion Model	7
2.2. Object State Observations	7
2.2.1. Observation Quantities	8
2.2.2. Observation Time-Stamp	8
2.2.3. Observation Preprocessing Time	9
2.2.4. Sensor Cycle-Time	9
2.2.5. Sensor Phase	10
2.3. Object State Prediction	11
2.3.1. Linear Differential Equations	11
2.3.2. Non-Linear Differential Equations	12
2.4. Object State Association	13
2.4.1. Gating	15
2.4.2. Nearest Neighbor Algorithm	15
2.4.3. Probabilistic Data Association	16
2.4.4. Multiple Hypotheses Tracking	16
2.5. Object State Update	17
2.5.1. Kalman Filter	17
2.5.2. Extended Kalman Filter	18
2.5.3. Unscented Kalman Filter	19
2.5.4. Particle Filter	20
2.5.5. Interacting Multiple Model Approach	20
2.5.6. Out-of-Sequence Measurement Problem	20
2.6. Performance Evaluation	21
2.6.1. Statistical Measures	22
2.6.2. (Monte Carlo) Simulations	23

2.6.3. Experimental Data	23
2.7. Limitations of Object Tracking Algorithms	23
2.8. Chapter Summary	24
3. Related Work	25
3.1. Sensors	25
3.1.1. Sensor Activation	25
3.1.2. Sensor Allocation	25
3.1.3. Sensor Parametrization	26
3.2. Bus Systems	26
3.2.1. Controller Area Network	27
3.2.2. TTP/C	27
3.2.3. FlexRay	28
3.3. Object Tracking	28
3.3.1. Processing of In-Sequence Measurements	28
3.3.2. Processing of Out-of-Sequence Measurements	28
3.3.3. Provision of the Feature Service Subsystems with Real-Time Images	31
3.3.4. Performance Evaluation	31
3.4. Feature Services	32
3.4.1. Longitudinal Control	32
3.4.2. Lateral Control	32
3.4.3. Integrated Longitudinal and Lateral Control	33
3.5. Chapter Summary	33
4. Numerical Simulation	35
4.1. Model of a State-of-the-Art Multi-Sensor Advanced Driver Assistance System	35
4.1.1. Sensors	35
4.1.2. Bus System	37
4.1.3. Object Tracking Subsystem	38
4.1.4. State-of-the-Art Model Schedule	39
4.2. Proposed Paradigm Shift	41
4.2.1. Sensors	41
4.2.2. Bus System	42
4.2.3. Object Tracking Subsystem	42
4.2.4. Time-Triggered Model Schedule	42
4.3. Model of the Environment	45
4.3.1. Variance of the Complexity of the Environment	45
4.3.2. Process Noise	46
4.4. Performance Measure	47

4.5.	Simulation Results	47
4.5.1.	Optimal Configurations	47
4.5.2.	Best State-of-the-Art and Time-Triggered Configurations	49
4.6.	Analysis of Simulation Results	55
4.6.1.	Relationship Between Mean Performance and Parameters	57
4.6.2.	Upper Bound for Fusion Process Time	59
4.6.3.	Process Noise Power Spectral Density	70
4.6.4.	Interrelations	70
4.7.	Discussion	72
4.8.	Chapter Summary	73
5.	Case Study	75
5.1.	Sensors	75
5.1.1.	Scabor	75
5.1.2.	Laser Scanner	77
5.2.	Bus System	78
5.3.	Object Tracking Subsystem	78
5.4.	Scenarios	79
5.5.	Differential Global Positioning System	81
5.6.	Experimental Results	81
5.7.	Discussion of Experimental Results	86
5.7.1.	Scenario 1	88
5.7.2.	Scenario 2	92
5.8.	Discussion	92
5.9.	Chapter Summary	94
6.	Discussion	95
6.1.	State-of-the-Art vs. Time-Triggered Configurations	95
6.2.	Outlook	97
A.	Observation Preprocessing Variance Modeling	99
B.	Process Noise of Linearized State Evolution	103

List of Figures

1.1. Components of an advanced driver assistance system based on object tracking	2
2.1. Basic steps of recursive object tracking algorithms	5
2.2. Observation time-stamps and observation PTs for a single-processor sensor m	9
2.3. Observation time-stamps, observation PTs, and sensor CTs for a single-processor sensor m	10
2.4. Observation time-stamps, observation PTs, sensor CTs, and sensor PH for a single-processor sensor m	11
2.5. Association and update of predicted images of the object states with object state observations	14
2.6. Prediction of sigma points versus mean of the prediction	19
2.7. Out-of-sequence measurement problem	21
4.1. Model of a multi-sensor ADASOT	35
4.2. State-of-the-art model schedule	40
4.3. Unsynchronized time-triggered model schedule	43
4.4. Synchronized time-triggered model schedule	44
4.5. Optimal state-of-the-art or time-triggered configurations for different points in parameter space	48
4.6. Comparison of best state-of-the-art configurations and best time-triggered configurations for $q = 0.01 \frac{m^2}{s^5}$	50
4.7. Comparison of best state-of-the-art configurations and best time-triggered configurations for $q = 0.1 \frac{m^2}{s^5}$	51
4.8. Comparison of best state-of-the-art configurations and best time-triggered configurations for $q = 1 \frac{m^2}{s^5}$	52
4.9. Comparison of best state-of-the-art configurations and best time-triggered configurations for $q = 10 \frac{m^2}{s^5}$	53
4.10. Comparison of best state-of-the-art configurations and best time-triggered configurations for $q = 100 \frac{m^2}{s^5}$	54
4.11. Temporal evolution of an object state ST image in the state-of-the-art BUFF configuration	60
4.12. Temporal evolution of an object state ST image in the state-of-the-art ADVA configuration	61

4.13. Temporal evolution of an object state ST image in the time-triggered unsynchronized BUFF configuration	62
4.14. Temporal evolution of an object state ST image in the time-triggered unsynchronized ADVA configuration	63
4.15. Temporal evolution of an object state ST image in the time-triggered synchronized configuration	64
5.1. Volkswagen Touran test vehicle	76
5.2. Test equipment	76
5.3. Case study set-up	77
5.4. Scenario 1	80
5.5. Scenario 2	80
5.6. Time-triggered unsynchronized BUFF configuration scenario 1, $x - y$ plot	82
5.7. Time-triggered unsynchronized BUFF configuration scenario 1, $RT - x$ plot, $RT - y$ plot	83
5.8. Time-triggered unsynchronized ADVA configuration scenario 1, $x - y$ plot	84
5.9. Time-triggered unsynchronized ADVA configuration scenario 1, $RT - x$ plot, $RT - y$ plot	85
5.10. Time-triggered synchronized configuration scenario 1, $x - y$ plot	86
5.11. Time-triggered synchronized configuration scenario 1, $RT - x$ plot, $RT - y$ plot	87
5.12. Time-triggered unsynchronized BUFF configuration scenario 2, $x - y$ plot	88
5.13. Time-triggered unsynchronized BUFF configuration scenario 2, $RT - x$ plot, $RT - y$ plot	89
5.14. Time-triggered unsynchronized ADVA configuration scenario 2, $x - y$ plot	90
5.15. Time-triggered unsynchronized ADVA configuration scenario 2, $RT - x$ plot, $RT - y$ plot	91
5.16. Time-triggered synchronized configuration scenario 2, $x - y$ plot	92
5.17. Time-triggered synchronized configuration scenario 2, $RT - x$ plot, $RT - y$ plot	93

List of Tables

4.1. ISM fusion PT, Δt_{PT}^{fusISM} , categories 66

Glossary

ABS	anti-lock brake system
ACC	adaptive cruise control
ADASOT	advanced driver assistance systems based on object tracking
ADVA	advanced algorithm approach
AEB	automatic emergency brake
ALK	automatic lane keeping
BUFF	buffering approach
CAN	controller area network
CT	cycle-time
CU	covariance union
DAS	driver assistance systems
DGPS	differential global positioning system
ECM	error covariance matrix
EKF	extended Kalman filter
ESP	electronic stability program
FTDMA	flexible time division multiple access
IMM	interacting multiple model
IPC	industrial PC
ISM	in-sequence measurement
JPDA	joint probabilistic data association
JT	junction tree
KF	Kalman filter
LIN	local interconnect network
MHT	multiple hypotheses tracking
MOST	media-oriented system transport
MP	mean performance
MSE	mean-square error
OEM	original equipment manufacturers
OOS	out-of-sequence
OOSM	out-of-sequence measurement
OSET	overall system execution time
PA	parking assistants
PDA	probabilistic data association
PDF	probability density function
PF	particle filter

PH	phase
PT	preprocessing time
RMSE	root-mean-square error
RT	real-time
ST	state-time
TDMA	time division multiple access
TTP	Time Triggered Protocol
UB	upper bound
UKF	unscented Kalman filter
WCET	worst case execution time

Symbols

$\hat{\vec{x}}$	(reasonable) estimate of an object state vector
\vec{v}	innovation
\vec{j}	object recognition Markov state vector
\vec{m}	complexity of environment Markov chain
\vec{v}	zero mean Gaussian noise of an object state observation
\vec{x}	object state vector
\vec{z}	object state observation
c	complexity of environment factor
F	temporal transition matrix of an object state vector
H	spatial transition matrix from object state space to object state observation space
I	identity matrix
J	object recognition Markov state vector transition probability matrix
K	Kalman gain matrix
M	complexity of environment Markov chain transition probability matrix
P	error covariance matrix of an object state vector
Q	error covariance matrix of a temporal transition
q	process noise power spectral density
R	error covariance matrix of an object state observation

1. Introduction

1.1. Background

In 2007, 431419 people were injured and 4949 people were killed in road accidents in Germany. Most of the fatalities were caused by situations in which a driver did not react properly or quickly enough to an unexpected event [21].

To make roads safer, many automotive original equipment manufacturers (OEM) and suppliers work on the development of advanced driver assistance systems based on object tracking (ADASOT) [64]. ADASOTs consist of one or multiple sensor(s), an object tracking subsystem and one or multiple feature service subsystem(s) interconnected via a bus system as schematically depicted in Figure 1.1. In contrast to the actual generation of active driver assistance systems (DAS), which avoid accidents by preventing the driver from losing control over the vehicle in dangerous situations (e.g., anti-lock brake system (ABS), electronic stability program (ESP) [52]), the new generation of ADASOT assists the driver with the longitudinal and/or lateral vehicle control in order to improve the driver's reaction to unexpected events [91] (e.g., automatic emergency brake (AEB), automatic lane keeping (ALK)).

1.2. Problem Statement

As the number and potential of ADASOT features grow, the question of how to guarantee the correctness of their services becomes more and more important [122, 123]. Although ADASOT feature service(s) “only” assist while the driver remains in full control, an incorrect ADASOT feature service can undoubtedly cause dangerous situations, as the capability of human beings to adapt quickly to unexpected events is restricted [44, 138].

The basis for achieving a correct ADASOT feature service is an exact assessment of the surrounding environment. This requires tracking all relevant objects within a feature service specific range and maintaining real-time (RT) images of the object states whose deviations from reality do not exceed a feature specific upper bound (feature specific accuracy demand) [134]. As RT images of evolving object states are invalidated by the progression of time, they have to be updated within a well-defined time interval (accuracy interval) with object state observations that satisfy a well-defined accuracy level [76]. As a result, the lowest possible accuracy level of object state observations, the maximum object state evolution and the maximum

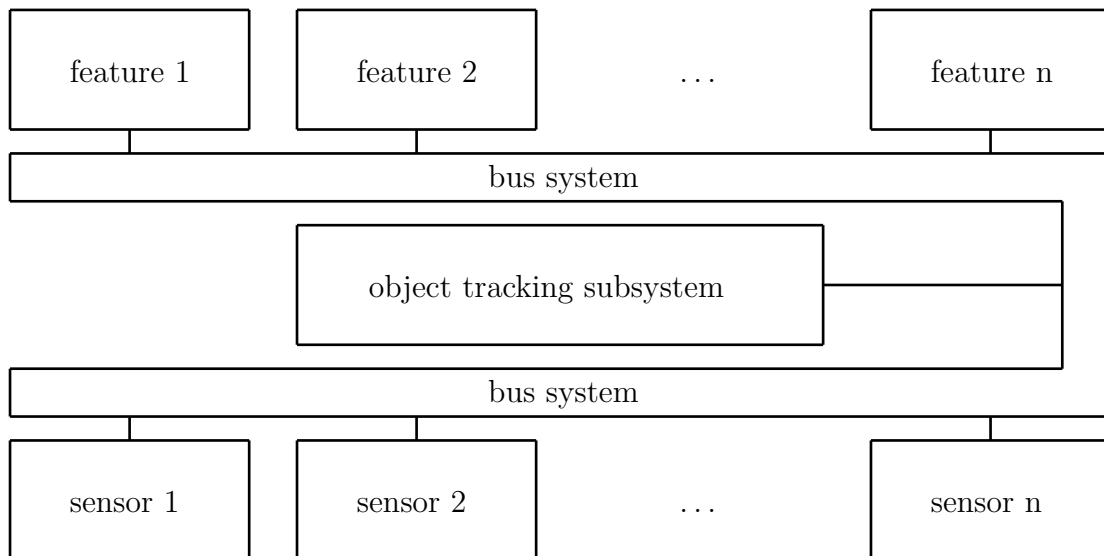


Figure 1.1.: Components of an advanced driver assistance system based on object tracking

accuracy interval that can occur in an ADASOT have to be taken into account when determining which feature specific accuracy demand can be satisfied [77].

As the accuracy level of an object state observation from a single-sensor may be subject to fluctuations due to sporadic object state observations being faulty or cluttered (radars cannot differentiate between reflections from vehicles and reflections from cans, cameras are sensitive to light changes or fog, etc.) [116, 15, 65], single-sensor ADASOTs are often limited to low feature service specific accuracy demands.

One approach to deal with this problem comprises updating the RT images of the object states with redundant object state observations derived from heterogeneous sensors, as sporadically occurring deficiencies in a sensor’s object state observations can usually be compensated by redundant object state observations from one or more other sensors (multi-sensor ADASOT approach [38]).

In contrast to single-sensor ADASOTs, where it is common to use point-to-point connections between sensor and object tracking subsystem, the use of multiple heterogeneous sensors in multi-sensor ADASOTs leads to the use of a bus system that interconnects the sensors and the object tracking subsystem [108].

In most state-of-the-art multi-sensor ADASOTs, the object state observations are sampled by uncontrolled sensors and transmitted over a controller area network (CAN) bus system [132, 142], which is the dominant bus system in the automobile industry. Despite its undoubted advantages, such as high flexibility and low costs, the usage of a CAN bus system or related indeterministic transmission protocols in

multi-sensor ADASOTs seems disputable as it is impossible to guarantee an update of the RT images of the object states with object state observations of all sensors within a well-defined accuracy interval. This is because the transmission of object state observations from a sensor to the object tracking subsystem may be delayed by the transmission of object state observations from further sensors or messages from other nodes that communicate over the bus system, thus leading to unpredictable transmission delays [91].

To overcome this shortcoming, a paradigm shift toward time-triggered multi-sensor ADASOTs based on the principles of the time-triggered architecture as presented by Kopetz et al. [79], wherein a time-triggered deterministic bus system establishes a global time-base and synchronizes the clocks of all nodes, seems feasible.

However, this paradigm shift is expected to affect the mean ADASOT performance, as the gained temporal determinism may introduce additional delays and demand supplementary hardware resources [75, 108].

It therefore exists the need to study whether the paradigm shift toward time-triggered multi-sensor ADASOT is feasible, or whether such a paradigm shift would lead to a drastic mean performance loss, which would outweigh the benefits of a guaranteed update of the RT images of the object states with object state observations of all sensors within a well-defined accuracy interval.

Accordingly it is the object of this thesis to study how the ADASOT mean performance is affected by the paradigm shift toward time-triggered multi-sensor ADASOTs.

To answer this research question for a class of multi-sensor ADASOTs there are two commonly used approaches:

- To define generalized models of state-of-the-art and time-triggered multi-sensor ADASOTs and to determine the mean ADASOT performance for both models through simulation of the model behavior in a generalized environment;¹ and
- To design a case study of a state-of-the-art multi-sensor ADASOT and a time-triggered multi-sensor ADASOT and to generalize the results gained from field tests.

Due to the difficulty to accomplish reproducible conditions for the high number of test drives that would be necessary to produce statistically meaningful results for a set of ADASOT scenarios in field tests [53], it seems feasible to tackle the posed question by a simulation and to show the feasibility of the paradigm shift toward time-triggered multi-sensor ADASOTs by a case study.

¹An analytical solution cannot be expected due to the use of heterogeneous sensors which prevents the system from reaching a steady state [90].

1.3. Outline

The thesis is structured as follows:

Chapter 2 introduces basic terms and concepts of object tracking such as sensor fusion, object state observations, object state prediction, object state association, object state update, performance evaluation and limitations of object tracking algorithms in order to provide the reader who is unfamiliar with the subject the necessary background and to provide the reader with the notation used throughout the thesis.

Chapter 3 provides a survey on related research in the fields of sensors used in military, automotive and robotics, event-triggered and time triggered bus-systems, object tracking with special focus on out-of-sequence measurements and automotive feature services.

Chapter 4 presents the comparison of state-of-the-art and time-triggered multi-sensor ADASOTs using results Monte Carlo simulations. Therein a model of a state-of-the-art multi-sensor ADASOT is presented and it is shown how the paradigm shift toward a time-triggered multi-sensor ADASOT can be mapped onto a time-triggered model. Furthermore, chapter 4 presents a model of the environment and defines a system performance measure. After presenting the results of the Monte Carlo simulations on the state-of-the-art model and the time-triggered model, the simulation results are analyzed and a conclusion on the observed model behaviors is drawn.

Chapter 5 presents a case study of a time-triggered multi-sensor ADASOT, consisting of two sensors, a bus system, and an object tracking subsystem. After presenting the experimental scenarios which are used for test drives and the differential global positioning system (DGPS) which is used to validate the estimated trajectories of the tracked object, chapter 5 presents the results of the experimental test drives. The experimental results are then analyzed and based thereon, a conclusion is drawn.

Chapter 6 concludes the thesis and provides an outlook to future research objectives.

2. Object Tracking

Object tracking is the continuous estimation of RT images of presumably evolving object states. This estimation is based on information gained from the most recent object state observations available and can be enriched by information from less recent object state observations when at least some knowledge about the evolution of the object states exists. Said knowledge may be based on of measured kinematic quantities (acceleration, speed, etc.) as well as kinematic constraints that a certain object type may underly (cars seldom fly, etc.) and makes it possible to predict images of future object states within certain accuracy bounds.

The aspired estimation of RT images of object states using object state observations from multiple sensors leads to a class of recursive object tracking algorithms which fall in the generic concept of sensor fusion and perform three basic steps, as depicted in Figure 2.1 (see also [125]):

1. Predicting images of the object states that relate to the the same time-instant as the most recent object state observations available;
2. Associating the predicted images of the object states and the most recent object state observations; and
3. Updating the predicted images of the object states with the associated object state observations.

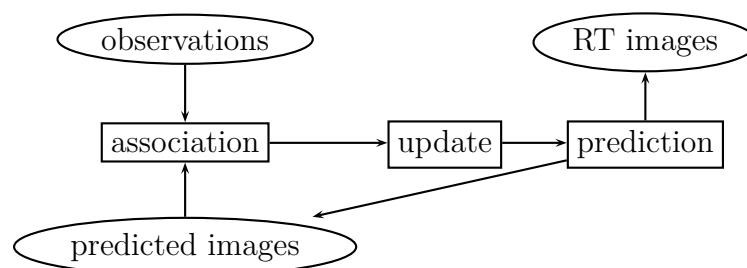


Figure 2.1.: Basic steps of recursive object tracking algorithms

In this chapter, the concept of sensor fusion and the time-triggered sensor fusion model (section 2.1), basic terms related to object state observations (section 2.2),

prediction of images of the object states (section 2.3), object state association (section 2.4), object state update (section 2.5), performance evaluation (section 2.6) and limitations of the object tracking algorithms (section 2.7) are briefly discussed.

2.1. Sensor Fusion

“Sensor fusion is the combining of sensory data or data derived from sensory data from disparate sources such that the resulting information is in some sense better than would be possible when these sources were used individually. The term better in that case can mean more accurate, more complete, or more dependable, or refer to the result of an emerging view, such as stereoscopic vision (calculation of depth information by combining two-dimensional images from two cameras at slightly different viewpoints).” [144, 38]

The relationship between sensor fusion and object tracking is very close, due to typical concepts of sensor fusion being deployed in object tracking. In this section, the objective of sensor fusion (subsection 2.1.1), the concept of sensor fusion (subsection 2.1.2), and the time-triggered sensor fusion model proposed by Elmenreich and Pitzek [40] (subsection 2.1.3) are discussed.

2.1.1. Objective of Sensor Fusion

According to Varshney [139], the ultimate objective of multi-sensor fusion is to provide an accurate situation assessment. The many factors which contribute to system performance and which are affected and typically improved by fusing information of several sensors, are:

- System reliability and robustness;
- Coverage;
- Confidence;
- Response time; and
- Resolution.

2.1.2. Concept of Sensor Fusion

One good approach to understand the concept of sensor fusion is to focus on its analogy to the human capability to perceive the surrounding environment. While crossing a road for example, a person will try to detect all surrounding moving objects and obstacles that could be dangerous or force this person to choose an alternative to the desired way.

At first glance, one would expect the human vision to be sufficient for this task. But hearing the sound of a car may yield additional information. For example, the speed or acceleration of an approaching car are difficult to assess from sight, if the car drives along a course which is close to the line-of-sight. In such a case the sound of an engine may provide additional information. When fusing the vision and sound information this kind of fusion is called competitive, as both senses deliver information on the same quantities but of different quality [37].

If some cars could not be seen due to obstacles but could be heard and others could be seen but not heard, the fusion of vision and sound information would be called complementary fusion, which is very easy to handle since the information can be simply appended to each other [20].

In contrast to that, cooperative fusion is the most difficult to design as it generates enhanced information which is not derivable from each of the single sensors. Such enhanced information is based on knowledge of interdependencies between the pieces of information provided by the sensors. One example for cooperative fusion would be that a person wishing to cross a road sees an approaching car and notices that the weather is rainy and cold. As it is known that bad weather may result in difficult driving conditions, the person will adapt the deemed safe distance to the approaching car accordingly.

2.1.3. Time-Triggered Sensor Fusion Model

Elmenreich and Pitzek [41, 39] distinguish between three levels of a time-triggered sensor fusion system, namely a smart transducer level, a fusion level, and an application level. The smart transducer level consists of sensors and actuators which are interconnected via a communication system. The fusion level provides sensor fusion and fault tolerance (e.g. object tracking subsystem). The application level provides decision making and a human machine interface (e.g. feature service subsystem).

The communication between these levels is accomplished over well-defined interfaces which enable system composability. The sensors send their observations to the fusion level in a unified manner and hide their interior. The fusion level processes the sensor observations and provides a fault-tolerant image of the environment to the application level.

It is argued, that this approach leads to a less complex system, the possibility of software reuse, as well as benefits in the area of configuration, diagnosis, and maintenance.

2.2. Object State Observations

The most common sources of state observations of moving objects are radar sensors, cameras and laser sensors. In order to simplify the following analysis and discus-

sion, this section defines basic terms related to sensor usage, such as observation quantities (subsection 2.2.1), observation time-stamp (subsection 2.2.2), observation preprocessing time (subsection 2.2.3), sensor cycle-time (subsection 2.2.4) and sensor phase (subsection 2.2.5).

2.2.1. Observation Quantities

Object state observations of a moving object “ n ” typically include dynamic observation quantities such as coordinates (x, y) , speed (v_x, v_y) , and acceleration (a_x, a_y) , as well as static observation quantities such as dimensions and shape.

Gaussian distributed observation quantities, for example, can be fully described by an object state observation vector, \vec{z}_n , representing their means

$$\vec{z}_n = \begin{pmatrix} x \\ y \\ v_x \\ v_y \\ \vdots \end{pmatrix} \quad (2.1)$$

and an assigned error covariance matrix (ECM), R_n .

$$R_n = \begin{pmatrix} Cov(x, x) & Cov(x, y) & Cov(x, v_x) & Cov(x, v_y) & & \\ Cov(x, y) & Cov(y, y) & Cov(y, v_x) & Cov(y, v_y) & & \\ Cov(x, v_x) & Cov(y, v_x) & Cov(v_x, v_x) & Cov(v_x, v_y) & \dots & \\ Cov(x, v_y) & Cov(y, v_y) & Cov(v_x, v_y) & Cov(v_y, v_y) & & \\ & & \vdots & & & \end{pmatrix} \quad (2.2)$$

Whenever the accuracy of the object state observation is not static $R_n(t)$, up-to-date accuracy information is necessary for a meaningful interpretation of the object state observations.

2.2.2. Observation Time-Stamp

As far as evolving object states are considered, a meaningful interpretation of object state observations further depends on the precise knowledge of the time points, t_k , which they reflect. As time is continuous, the precision of a digital representation of a time point is always bound by the number of significant digits and the precision of the used clock. But even if the number of significant digits and the precision of the used clock are sufficient, difficulties may arise whenever an object state observation is collected over a period of time, which is too long to be neglected.

A laser scanner, for example, which rotates over 360° will detect objects at different points in time as the laser beam sweeps over the detection region. The assignment of one time-stamp to all objects that have been detected during one scan would be a constant source of failure. This can be diminished, if the time point at which the laser scanner points toward the geometric middle of an object, is assigned the time-stamp of the corresponding object state observation.

2.2.3. Observation Preprocessing Time

Most automobile sensors rely on measurement principles where the desired observation quantities are only implicitly included in a scan or snapshot and must be extracted using complex observation preprocessing algorithms. The time interval which elapses while a sensor “ m ”, $sens_m$, extracts the observation quantities is referred to as observation preprocessing time (PT), $\Delta t_{PT}^{sens_m}$, in the further.¹ Said interval correlates with the complexity of the surrounding environment as an increasing complexity of the surrounding environment leads to increasing observation preprocessing times [119].

Figure 2.2 depicts the activity of a single-processor sensor over RT, wherein observation time-stamps are indicated by circles and observation PTs are indicated by bars.

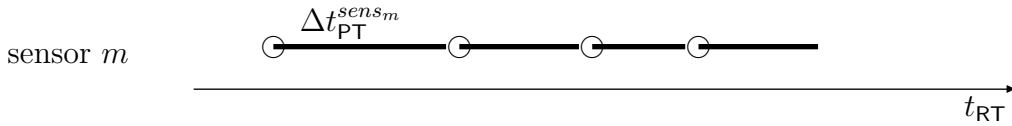


Figure 2.2.: Observation time-stamps and observation PTs for a single-processor sensor m

2.2.4. Sensor Cycle-Time

The sensor cycle-time (CT), $\Delta t_{CT}^{sens_m}$, is defined by the time interval between two consecutive observation time-stamps as depicted in Figure 2.3 wherein the CT is indicated by the thin line above the observation PTs.

In a single-processor sensor, the sensor CT is therefore equal to or higher than the sensor PT and delivers the highest number of measurements if the sensor CT is adapted to the sensor PT, i.e., a new sample is processed as soon as the processing of the last sample has been finished. In a multi-processor sensor, the sensor CT can

¹The term *preprocessing* refers to raw measurements being processed by a sensor *before* being processed by the object tracking subsystem.

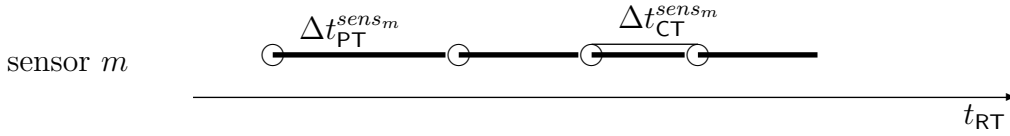


Figure 2.3.: Observation time-stamps, observation PTs, and sensor CTs for a single-processor sensor m

be smaller than the sensor PT if, e.g., scans or snapshots are routed to a processor with freely available computational resources [69].

In order to achieve a constant sensor CT, i.e., delivering object state observations whose time-stamps are equidistantly spaced over RT, a sensor must be provided with such processing resources that the worst case execution time (WCET) of the observation PTs equals the given sensor CT multiplied by the number of processors in use. However, this approach seems unfeasible if the potential changes of the complexity of the surrounding environment are very high and would lead to a sensor being equipped with enormous computing resources which would be unneeded in scenarios exhibiting normal complexity [119].

Two possibilities to overcome this problem are commonly treated in literature. The first is to use observation preprocessing algorithms with “anytime behavior”. The second is to define an upper bound for the considered complexity.

In the case of observation preprocessing algorithms with anytime behavior, the quality of the observation quantities is continuously improved as long as the observation PT is shorter than the sensor CT. When the observation preprocessing time equals the sensor CT, the observation preprocessing is stopped. As a result, the observation quantities can be used anytime during the improvement phase.

If defining an upper bound for the considered complexity, less relevant information has to be discarded. For example, a sensor takes a snapshot of the surrounding environment which implicitly contains observation quantities of 15 objects but only the object state observations of the 10 most relevant objects (objects within the region of interest) are further preprocessed.

2.2.5. Sensor Phase

The phase (PH) of a sensor m between the k th object state observation time-stamp and a l th benchmark is abbreviated by $\Delta t_{\text{PH},k,l}^{\text{sens}_m}$. In Figure 2.4 $\Delta t_{\text{PH},1,1}^{\text{sens}_m}$ and $\Delta t_{\text{CT}}^{\text{sens}_m}$ are depicted for a single-processor sensor m wherein the PH is indicated by the thin line above the observation PTs. The first benchmark is the origin of the RT axis.

If PTs, CTs and PH of a sensor are fixed, the sensor’s timely behavior is fully deterministic.

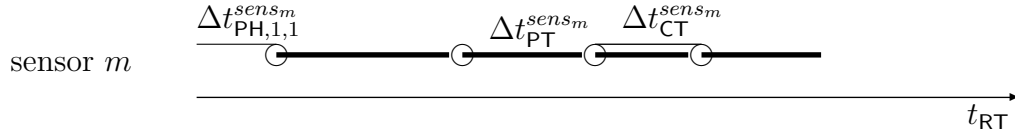


Figure 2.4.: Observation time-stamps, observation PTs, sensor CTs, and sensor PH for a single-processor sensor m

2.3. Object State Prediction

Whenever an object state vector comprises along with the object quantities all of their non-zero first, second and higher order derivatives with respect to time, the evolution of said object quantities over a time interval $\Delta t = (t_{k+1} - t_k)$ can be fully predicted. In the field of object tracking the evolution of object state vectors is typically described in the form of linear (subsection 2.3.1) or nonlinear (subsection 2.3.2) differential equations.

2.3.1. Linear Differential Equations

Formula 2.3 contains a set of linear differential equations that describe a kinematic process where \vec{x} represents an object state vector and F represents a differential operator.

$$\dot{\vec{x}}(t) = F\vec{x}(t) \quad (2.3)$$

Let's assume that at time point t_k there is an estimated image of an object state which is the expected object state vector, $\hat{\vec{x}}(t_k) = E(\vec{x}(t_k))$, derived from object state observations, $\{\vec{z}(t_k) : k = 1, 2, \dots\}$. The ECM of $\hat{\vec{x}}(t_k)$ is denoted as $P(t_k)$.

If an image of a future object state $\hat{\vec{x}}(t_{k+1})$ has to be predicted from this estimated image, the general solution using formula 2.3 is given by

$$\begin{aligned} \hat{\vec{x}}(t_{k+1}) &= E\left(e^{F\Delta t}\vec{x}(t_k)\right) \\ &= e^{F\Delta t}\hat{\vec{x}}(t_k) \end{aligned} \quad (2.4)$$

with

$$e^{F\Delta t} = I + F\Delta t + \frac{(F\Delta t)^2}{2!} + \dots \quad (2.5)$$

and I as identity matrix. $P(t_{k+1})$ representing the ECM of $\vec{x}(t_{k+1})$ is calculated by

$$P(t_{k+1}) = \left(e^{F\Delta t}\right) P(t_k) \left(e^{F\Delta t}\right)^T + Q(\Delta t) \quad (2.6)$$

where $Q(\Delta t)$ is referred to as integrated process noise. For the special case of object tracking and under the assumption that the object state quantities' derivatives with respect to time that are not contained in the object state vector can be described by a probability density function (PDF) and have zero mean², $Q(\Delta t)$ can be explicitly calculated as the integrated modeling error

$$\begin{aligned} Q(\Delta t) &= E \left(\left(\int_0^{\Delta t} e^{F\tau} d\tau \right) \vec{g}\vec{g}^T \left(\int_0^{\Delta t} e^{F\tau} d\tau \right)^T \right) \\ &= \left(\int_0^{\Delta t} e^{F\tau} d\tau \right) E(\vec{g}\vec{g}^T) \left(\int_0^{\Delta t} e^{F\tau} d\tau \right)^T \\ &= \left(\int_0^{\Delta t} e^{F\tau} d\tau \right) G \left(\int_0^{\Delta t} e^{F\tau} d\tau \right)^T \end{aligned} \quad (2.7)$$

with \vec{g} accounting for the derivatives with respect to time that are not contained in the object state vector but may be not equal to zero during the prediction interval.

2.3.2. Non-Linear Differential Equations

In fact, many kinematic processes can only be correctly described by a set of non-linear differential equations as shown in equation 2.8.

$$\dot{\vec{x}}(t) = \vec{f}(\vec{x}(t)) \quad (2.8)$$

In contrast to the solution of a set of linear differential equations, where the object state vector's PDF does not have to be explicitly known in order to predict a future image of the object state vector's mean and ECM, a prediction of the mean and ECM of a future image of the object state vector based on a solution of equation 2.8 requires full knowledge of the object state vector's PDF due to the solution's nonlinearity.

As the computation of a PDF would heavily increase the computational complexity and as a PDF cannot generally be described by a finite set of parameters [67, 66], practical estimators use approximations. A technique conventional in the state-of-the-art is to approximate the mean of the predicted image of the object state vector by the predicted mean of the image of the object state vector

²In fact, in a kinematic process all derivatives of object state quantities are deterministic, but since they are not known, it seems reasonable to treat them as statistical values with zero mean.

$$\begin{aligned} E(\dot{\hat{x}}(t)) &= E(\vec{f}(\vec{x}(t))) \\ &\approx \vec{f}(\hat{\vec{x}}(t)) \end{aligned} \quad (2.9)$$

The ECM of the predicted image of the object state vector can be predicted by

$$\dot{P}(t) = F^{lin}(\hat{\vec{x}}(t_k)) P(t) + P(t) F^{lin}(\hat{\vec{x}}(t_k))^T + Q(\Delta t) \quad (2.10)$$

using

$$F^{lin}(\hat{\vec{x}}(t_k)) := \left. \frac{\partial \vec{f}}{\partial \vec{x}} \right|_{\vec{x}=\hat{\vec{x}}(t_k)} \quad (2.11)$$

In general however, the result of the non-linear transformation using this approximation may be biased and the estimated ECM may be not consistent³.

2.4. Object State Association

As on the one hand, object state observations may be noisy, faulty, cluttered and these error sources may even be correlated [30, 28, 72] and on the other hand, the prediction of images of the future object states may only be an approximation of the real object state evolution, their association may not be a trivial exercise.

In the upper part of Figure 2.5, two stored object states and four object state observations are pictured. The values of their coordinates (x, y) follow a Gaussian probability distribution, indicated by 99% error ellipses.

To generate an image of the surrounding environment, which is consistent with reality, the following errors need to be avoided:

- The association of a predicted image of an object state with an object state observation although the association is wrong; and
- The non-association of a predicted image of an object state with an object state observation although the association would be correct.

The association process is often divided into two steps:

1. The “distance” between predicted images of the object states and object state observations (object pairs) is calculated using a metric based on the probability distributions of the objects state vectors [11]. Only object pairs whose distance is smaller than a defined value (see also subsection 2.4.1) are considered in step 2; and

³In the sense that the ECM is a statistical upper bound for the deviations between estimate and reality.

2. Object Tracking

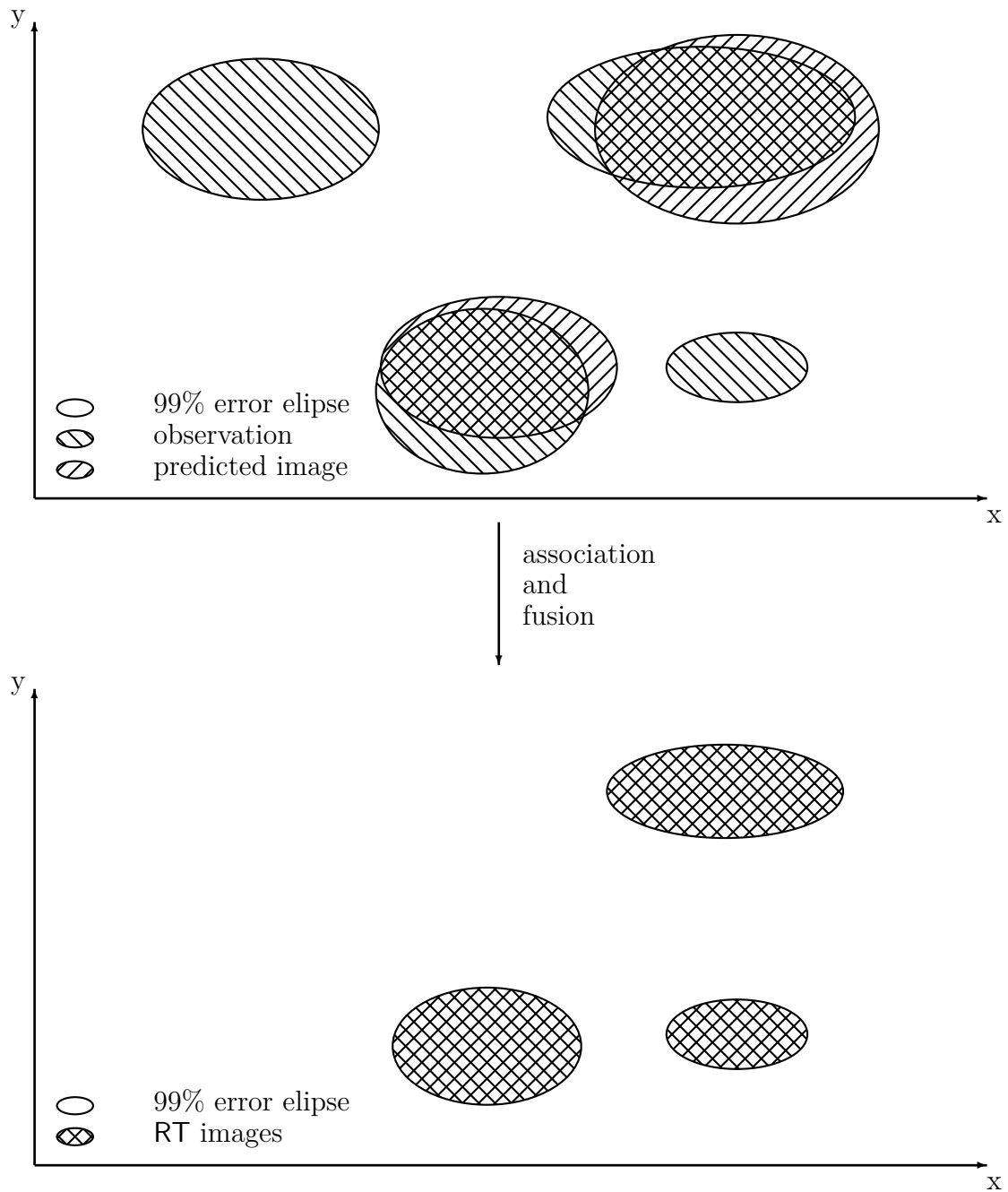


Figure 2.5.: Association and update of predicted images of the object states with object state observations

2. If, after step 1, every predicted image of an object state is associated to exactly one object state observation in an object pair, the association process is trivial. If not, the association can be carried on using a nearest neighbor algorithm (subsection 2.4.2), a probabilistic data association (subsection 2.4.3) or a multiple hypotheses tracking (subsection 2.4.4).

Once associated, predicted images of the object states are then updated with all associated object state observations (section 2.5). If an object state observation cannot be associated to any predicted image of the object states, but the object that is represented by said object state observation has been observed for some time period, a new RT image of this object state is generated. Otherwise, the information is dismissed.

2.4.1. Gating

The screening of improbable object state observations using a distance metric and an upper bound, where all object pairs whose distance is greater than the upper bound are neglected, is called “gating”.

There are two common gating methods. One method is based on checking whether the absolute numeric values of the differences between all equivalent variables of each object pair are smaller than the square-root of the diagonal element of the corresponding ECM, which results from addition of the associated covariance matrices, multiplied with a scaling factor k (e.g. $|a_1 - a_3| < k \cdot \sqrt{(P_1 + P_3)(1, 1)}$, $|b_1 - b_3| < k \cdot \sqrt{(P_1 + P_3)(2, 2)}, \dots$). All object pairs which have not passed this check are disregarded in the remaining association process. One disadvantage of this method is that the information which is contained in the non-diagonal elements of the object state ECMs is neglected.

The other gating method uses the whole information of the object state ECMs by checking whether the squared Mahalanobis distance

$$d^2 = (\vec{x}_1 - \vec{x}_2)^T (P_1 + P_3)^{-1} (\vec{x}_1 - \vec{x}_2) \quad (2.12)$$

is smaller than a value “ g ”. A formula for the calculation of a maximum likelihood gate g_{ML} can be found, e.g., in [18]. Although the method is computationally more demanding, it has the advantage of fully exploiting the available information contained in the object state ECMs.

2.4.2. Nearest Neighbor Algorithm

The nearest neighbor algorithm formulates the association process of the remaining possible object pairs as a linear optimization problem. This linear optimization

problem is based on a binary association matrix, X , which leads to the linear cost function

$$\sum_{i=1}^n \sum_{j=1}^m x_{i,j} d_{i,j} \quad (2.13)$$

with the linear restrictions $\sum_{j=1}^m x_{i,j} = 1$ for all i and $\sum_{i=1}^n x_{i,j} \leq 1$ for all j .⁴

The nearest neighbor algorithm achieves a unique association by assuming that every object is detected exactly once. However, this assumption disregards the possibility that an object may be observed several times and another object might not be detected at all.

2.4.3. Probabilistic Data Association

The probabilistic data association (PDA) method [14] assigns probabilities to all object pairs which are within one gate, i.e., only a gate specific distance apart. These probabilities are used to calculate a weighted average update of the predicted image of the object state and all object state observations that are within its gate according to the update algorithm (see also subsection 2.5). An extension to the PDA method is the joint probabilistic data association (JPDA) where the assigned probabilities are calculated using all object pairs [10].

2.4.4. Multiple Hypotheses Tracking

The discussed association methods have the disadvantage that an incorrect association at a time point, t_k , may lead to further incorrect associations in the future ($t_{k+n} \parallel n > 0$). This is a consequence of update algorithms, which compute a weighted sum of predicted images of object states and associated object state observation(s), where each wrong association increases the inconsistency between the RT image of the object states and reality.

The multiple hypotheses tracking (MHT) method [115] pursues multiple association alternatives over a sequence of updates as long as the alternatives are still probable. The probability of an alternative increases whenever an object state observation can be associated with said alternative and decreases if no object state observation can be associated. As the number of alternatives grows exponentially, an alternative is abandoned if its assigned probability drops under a lower bound. If the distance between two alternatives is smaller than a certain reference value, the alternatives are merged.

In contrast to the PDA, where the probability of associations leads to the update of a predicted image of the object state with a probabilistically weighted composite

⁴Linear optimization problems can be solved, e.g., by the Simplex algorithm [31].

of all object state observations, the MHT defers the decision to the future, where prospective object state observations can be used to resolve the uncertainty.

2.5. Object State Update

If a kinematic process can be modeled by a set of linear differential equations, a common method to update a predicted object state with object state observations is by employing a Kalman filter [12]. The Kalman filter calculates the RT image of an object state as a weighted average of the predicted object state and associated object state observations (see subsection 2.5.1). The tracking of kinematic processes which are governed by a set of non-linear differential equations has motivated multiple approximative update algorithms which are related to the Kalman filter. The most commonly used are the extended Kalman filter (EKF) (subsection 2.5.2), the unscented Kalman filter (UKF) (subsection 2.5.3) and the particle filter (PF) (subsection 2.5.4).

As the underlying constraints of a kinematic process may change over time, the interacting multiple model approach has been developed to account for these changes (subsection 2.5.5).

If heterogeneous sensors with different observation preprocessing times are used, object state observations may not arrive in chronological order at the object tracking subsystem leading to the so called “out-of-sequence measurement problem” (subsection 2.5.6).

For notational convenience, an object state vector reflecting time point t whose latest update has been by object state observations reflecting time point t_k is denoted $\hat{\vec{x}}(t | \vec{z}_{t_k} : t_k \leq t)$ or short $\hat{\vec{x}}(t | t_k)$.

2.5.1. Kalman Filter

The Kalman filter (KF) is designed for processes which can be modeled by linear differential equations according to equation 2.3.

Let's assume, that at time point t_k , an object state observation

$$\vec{z}_{t_k} = H_{t_k} \cdot \vec{x}(t_k) + \vec{v}_{t_k} \quad (2.14)$$

with ECM, $R_{t_k} = E[\vec{v}_{t_k} \cdot (\vec{v}_{t_k})^T]$, is available.

An algorithm is required which updates a predicted image of an estimated object state, $\hat{\vec{x}}(t_k | t_{k-1})$, and the corresponding ECM, $P(t_k | t_{k-1})$, with the (associated) object state observation, \vec{z}_{t_k} .

The difference between the object state observation \vec{z}_{t_k} and the predicted image of the object state in observation space, $\hat{\vec{z}}_{t_k} = H_{t_k} \cdot \hat{\vec{x}}(t_k | t_{k-1})$, is called innovation, $\vec{v}_{t_k} = \vec{z}_{t_k} - \hat{\vec{z}}_{t_k}$.

In the KF, a linear approach is chosen for the update of the predicted object state vector with the innovation

$$\hat{\vec{x}}(t_k|t_k) = \hat{\vec{x}}(t_k|t_{k-1}) + K_{t_k} \cdot \vec{v}_{t_k} \quad (2.15)$$

where K_{t_k} is determined as

$$K_{t_k} = P(t_k|t_{k-1}) \cdot H_{t_k}^T \cdot (H_{t_k} P(t_k|t_{k-1}) H_{t_k}^T + R_{t_k})^{-1}. \quad (2.16)$$

The corresponding object state ECM is updated by

$$P(t_k|t_k) = (I - K_{t_k} \cdot H_{t_k}) \cdot P(t_k|t_{k-1}) \cdot (I - K_{t_k} \cdot H_{t_k})^T + K_{t_k} \cdot R_{t_k} \cdot K_{t_k}^T. \quad (2.17)$$

2.5.2. Extended Kalman Filter

The EKF is designed for processes which can be modeled by non-linear differential equations according to equation 2.8.

Let's assume, that at time point t_k , an object state observation

$$\vec{z}_{t_k} = h(\vec{x}(t_k|t_{k-1})) + \vec{v}_{t_k} \quad (2.18)$$

with ECM, $R_{t_k} = E[\vec{v}_{t_k} \cdot (\vec{v}_{t_k})^T]$, is available.

In order to apply an algorithm which is similar to the KF, the EKF is based on approximating the object state ECM evolution according to subsection 2.3.2 and on approximating the state-space to measurement-space transformation by

$$H^{lin}(\hat{\vec{x}}(t_k|t_{k-1})) := \left. \frac{\partial \vec{h}}{\partial \vec{x}} \right|_{\vec{x}=\hat{\vec{x}}(t_k|t_{k-1})} \quad (2.19)$$

which is employed in the calculation of K_{t_k} analog to equation 2.16. The update of the object state ECM, $P(t_k|t_{k-1})$, is then analog to equation 2.17.

However, the made approximations imply that:

- The results may be biased; and
- The estimated ECM $P(t_k|t_k)$ may be inconsistent,

which may lead in turn to a diverging estimate.

2.5.3. Unscented Kalman Filter

The UKF is based on the premise that it is easier to approximate the PDF of an object state vector or object observation vector than to approximate the prediction or transformation described by non-linear differential equations or non-linear functions as in the EKF [67, 66]. The approximation of a PDF is achieved by deterministically choosing a set of so called “sigma points” whose weighted average corresponds to the mean of the object state vector and their ECM corresponds to the ECM of the object state vector. Future sigma points are then predicted according to the non-linear differential equation 2.8. The mean and ECM of the prediction are then approximated by the mean and ECM of the predicted sigma points (see Figure2.6).

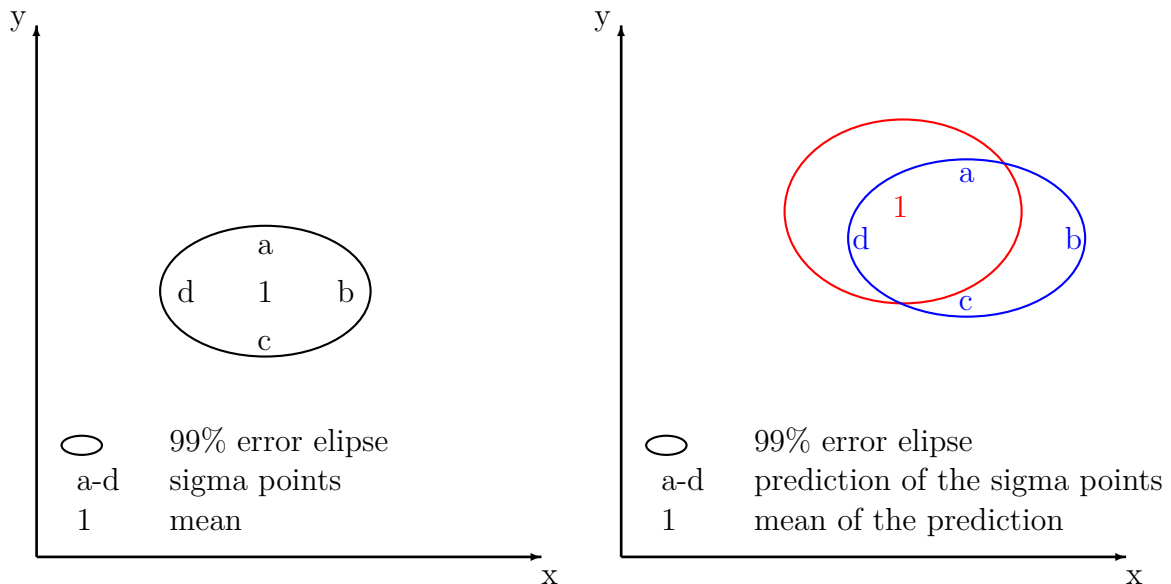


Figure 2.6.: Prediction of sigma points versus mean of the prediction

Figure 2.6 consists of two parts. The left part shows the 99% error elips of an object position where the mean is indicated by the figure 1. Furthermore, the position can be described by a set of sigma points indicated by figures a to d which can be used to calculate the position mean.⁵ In the right part there are two 99% error elipses, one which origins from predicting the mean indicated again by the figure 1, and a second one which origins from predicting the sigma points indicated again by the figures a to d, where the difference of the 99% error elipses is a direct result of the nonlinear prediction process.

⁵In practice the number of sigma points should be sufficiently large which is avoided here for sake of simplicity.

2.5.4. Particle Filter

The PF [4] resembles the UKF by approximating the PDF of an object state vector by a set of weighted “particles” whose mean and ECM converge to the mean and ECM of the object state vector. In contrast to the UKF, the particles of the PF are chosen randomly and the weighting of the particles must be consistent with the probabilistic interpretation of the PDF.

2.5.5. Interacting Multiple Model Approach

To reach high filter accuracy it is important to model the object state evolution with sufficient precision. Usually a more precise model requires additional object quantities in the object state vector.

As the computational complexity grows dramatically with the dimension of the object state vector, the deployment of a more realistic object state evolution model may lead to a higher demand for computational resources.

The interacting multiple model (IMM) approach tries to choose at runtime the smallest object state vector and thereby the least complex object state evolution model which still describes the object behavior with sufficient accuracy. This is performed by switching between multiple object state vectors and object state evolution models.

As an alternative, a weighted average of the estimated state vectors and ECMs from multiple simultaneously running filters is computed (which, of course, does not imply a decrease of computational complexity).

2.5.6. Out-of-Sequence Measurement Problem

In multi-sensor ADASOTs, object state observations of the same object can arrive out-of-sequence [129, 6], i.e., not in chronological order. Often, an out-of-sequence measurement (OOSM) (as opposed to an in-sequence measurement (ISM)) is caused by an indeterministic transmission system, where the transmission time of a measurement may vary so much that an older measurement may overtake a newer measurement. Such behavior is caused by transmission protocols with many retries such as many Internet protocols or in networks with dynamic routing (Internet, wireless sensor networks).

However, even if communication protocols with deterministic behavior, such as time-triggered approaches like FlexRay [45], TTCAN [57], TTP [131], or TTP/A [80] are used, the OOSM problem may arise.

Figure 2.7 depicts a situation with an OOSM problem that is independent from communication system issues, i.e., the transmission times of object state observations from both sensors to an object tracking subsystem, $\Delta t_{\text{PT}}^{\text{etb}_1/\text{ttb}_1}$ and $\Delta t_{\text{PT}}^{\text{etb}_2/\text{ttb}_2}$, are approximately equal. Due to different observation preprocessing times, $\Delta t_{\text{PT}}^{\text{sens}_1} >$

$\Delta t_{PT}^{sens_2}$, the measurement stemming from sensor 2 is received earlier at the object tracking subsystem than the measurement stemming from sensor 1, although the measurement from sensor 2 represents a more recent snap-shot of the surrounding environment.

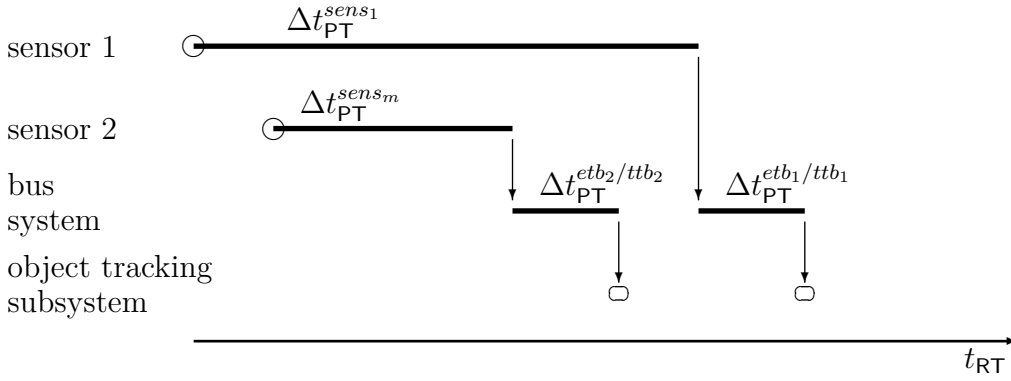


Figure 2.7.: Out-of-sequence measurement problem

As the KF cannot handle OOSMs, architectural and algorithmic solutions have been developed. Two solutions to the OOSM problem are the use of a measurement buffer in the object tracking subsystem that sorts the measurements chronologically (buffering approach (BUFF)) or algorithms based on the KF that allow the incorporation of OOSM (advanced algorithm approach (ADVA)), e.g., [9] (see also 3.3.2).

2.6. Performance Evaluation

According to [137], the tracking of an object state comprises four areas:

- Object state space;
- Object state observation space;
- Probabilistic relationship between object state space and object state observation space; and
- Estimation criteria.

In fact, the estimation criteria are usually interpreted as optimization criteria which leads to an optimization problem, e.g., how to minimize a statistical measure (subsection 2.6.1) for the accuracy of a tracked object state. Such a problem can be (optimally) solved by the object state tracking algorithms, e.g., a KF running on an object tracking subsystem.

In fact, most research papers base the performance evaluation of proposed or discussed algorithms on a statistical measure which is optimized in the optimization problem. For a given scenario, this statistical measure may then be determined using a set of (Monte Carlo) simulations (subsection 2.6.2).

According to Zhao and Li [149], it is important to differentiate between statistical optimality of a tracking algorithm and its performance. This is due to the fact that an estimation criterion must be simple enough to be mathematically manipulable, which usually prevents it from being a good measure of the performance.

As a result, in almost all cases there is a significant gap between estimation criteria and feature services subsystem requirements.

This leads to the necessity to verify the performance of an object tracking algorithm with real data (subsection 2.6.3), taking into account a set of performance measures that can hopefully be aggregated in order to establish dominance relations between available alternatives.

2.6.1. Statistical Measures

Two statistical measures for evaluating the performance of an object state estimation algorithm are the trace and the determinant of the estimated object state ECM [18]. Both are principal-axis-invariant statistical measures for the deviation between the estimated mean of an object state vector and a real object state. For a two-dimensional object state vector, the summands of the trace or the determinant of an object state ECM can be illustrated by the semi-axes or the area of an error ellipse [27].⁶ As the KF is optimal in that it minimizes the trace of the object state ECM [51], the trace of the object state ECM is the natural intrinsic performance measure when comparing simulations on KF based algorithms.

If a reference for the trajectory of an object state is available, a common measure is the mean-square or root-mean-square deviation between trajectory and reference (mean-square error (MSE) or root-mean-square error (RMSE)).

If the task of the object tracking subsystem is to maintain multiple tracks at the same time, a further performance measure of the object tracking subsystem, namely that of consistency can be expressed by the ratios of the number of not-tracked but existing objects to the number of tracked objects and the number of not-existing objects that are tracked (clutter) and the number of tracked objects at time t_k [93].

One reason for not tracking an existing object may be that the object just entered the perception range or that the object state observation could not be associated to the tracked object state for some period of time or that no corresponding object state observations were provided by the sensors. The reason for tracking non existing

⁶If a position error with a probability content of 95% is to be displayed, the standard deviations contained in the ECM are to be multiplied by a factor of 2.449 [56] before using them as values for the semi axes of the error ellipse.

targets may be that the sensor delivers object state observations from clutter (e.g., trees, houses, etc.).

2.6.2. (Monte Carlo) Simulations

Simulations for providing statistical performance measures can be classified into two categories [17].

The first category uses intrinsic statistical performance measures maintained by the applied fusion algorithm e.g. the ECM of a tracked object state vector [141].

The second category feeds the object tracking system with a sequence of pseudo measurements, e.g., each measurement being a compound of a predefined object state observation and a randomly generated error with known statistical characteristics. The output of a multitude of such Monte Carlo runs can then be compared to the predefined object state and the deviation thereof can be used for calculating extrinsic statistical measures [33, 83, 32].

2.6.3. Experimental Data

Experimental data gained from field test bears the advantage that it is generally closer to reality than any simulation and will therefore lead to more meaningful results.

However, according to Goodman et al. [53] “The comparison of the performance of two data fusion systems, or determining the performance of any individual system relative to some predefined standard, is far more daunting a prospect than at first might seem to be the case”. This is due to several facts:

- It is very hard to accomplish reproducible conditions for field tests;
- Even if reproducible conditions are accomplished, the statistical nature of the problem longs for a large number of test drives to produce meaningful results; and
- In most cases, the optimization with respect to one performance criterion will degrade another performance criterion.

2.7. Limitations of Object Tracking Algorithms

Besides the very simple forms of object tracking where object state observation quantities can be directly measured and fused, most object tracking algorithms are based on transformation models and prediction models that enable the synergistic use of object state observations from heterogeneous sensors.

Despite the statistically proved optimality of a fusion algorithm, the optimality of a system service is based on the assumption that the used models reflect all relevant facts of reality. As the capability of the human mind to deal with systems of high complexity is restricted, most models are only vague approximations of the real world [127] (for example a linear model as approximation of a non-linear real-world system). Only if a system accounts for these approximations, for example by increasing the process noise, meaningful results can be produced.

However, it is not only the reduction of complexity which may lead to results of low quality but also modeling errors in the sense of badly calibrated sensors.

Therefore, it must be kept in mind that object tracking subsystems have to be handled carefully in order to avoid overconfidence in the system's robustness and quality.

2.8. Chapter Summary

Object tracking is the continuous estimation of RT images of presumably evolving object states by fusing predicted images of object states with associated object state observations. The evolution of object states is often modeled by linear or non-linear differential equations as an approximation of reality. The association of predicted images of object states and object state observations is accomplished by calculating their distance and finding a globally optimal combination between predicted images of object states and object state observations. The update of associated predicted images of object states and object state observations leads to an optimization problem which is solved by algorithms that estimate an object state vector with minimal errors.

In order to compare these algorithms, one has either the possibility to run exhaustive simulations with artificial object state observations that are corrupted by zero-mean, statistically known errors and comparing the estimated object state to the artificial object state observations or to run exhaustive field tests with real data and comparing the estimated results with validation data representing the true object state trajectory.

Although object tracking employs algorithms that are directly derived from solutions of optimization problems and are thus statistically optimal, the fact that these optimization problems are usually based on models that only approximate reality may lead to suboptimal results.

3. Related Work

This chapter provides an overview of the relevant research in the fields of sensors (section 3.1), bus systems (section 3.2), object tracking (section 3.3), and feature services (section 3.4).

3.1. Sensors

The scheduling of sensors has received considerable attention over the last years, especially in the fields of military [124] and robotics [47]. This is due to the fact that in both fields multiple sensors provide object state observations for one or multiple feature services under a dynamically changing environment.

In such an environment, it seems feasible to activate those sensors which are most appropriate for delivering the desired information under the actual environmental conditions (subsection 3.1.1). Radar and laser sensors, e.g., are mostly unaffected by weather conditions like fog and rain and can deliver high quality measurements on longitudinal motion within a range of more than 100 meters, while vision sensors can provide high quality measurements on lateral motion as well as additional information like object dimensions and shape. If object state observations from an activated sensors cannot be used simultaneously by all services, sensor resources have to be allocated accordingly (subsection 3.1.2). If parameters of activated sensors can be adjusted in accordance with service needs, they can be optimized (subsection 3.1.3), e.g., for a service with highest priority.

3.1.1. Sensor Activation

If object state observations from heterogeneous sensors are available and the environmental conditions or the demand for object state observations changes drastically over time, the activation of the most appropriate sensor set can lead to improved results [127, 136] or the reduction of sensor usage costs [92].

3.1.2. Sensor Allocation

If more than one feature service requests object state observations from multiple sensors but the object state observations can either be used exclusively for one specific feature service or the resources are limited in such a way that not all requests

for object state observations can be handled simultaneously, a sensor allocation has to be performed. According to Schrage et al. [120], the goal of sensor allocation is to minimize the resource usage costs and to maximize the likelihood that all mission objectives will be completed.

In [135] Van Keuk et al. studied the efficient allocation of radar resources such as beam scheduling.

In [50], Gage et al. describe a sensor allocation in a mobile robot, where the sensor allocation is complicated by the non-existence of information about future assignments.

3.1.3. Sensor Parametrization

If sensors can be parameterized and thereby adapted to the needs of one specific feature service, a sensor parametrization has to be performed. According to sensor capabilities, one can differentiate between off line parametrization and on line parametrization.

In [105], Mehra uses different norms of the observability and the Fisher information matrix [121] as criteria for the optimization of measurement scheduling and shows that it is preferable to cluster measurements around specific design points t_k .

Avitzour and Rogers [5] present a theory of optimal measurement scheduling for least squares estimation which is based on the assumption that the cost of a measurement is inversely proportional to the variance of measurement noise and that it is possible to distribute the total measurement cost arbitrarily among a set of measurements.

Le Cadre differentiates between active measurements, i.e., with usage costs, and passive measurements, i.e., without usage costs, and shows an information gain which results from the usage of the active measurements in [89].

In [107], Mourikis et al. compute the localization uncertainty of a group of mobile robots wherein the localization uncertainty is determined by the covariance matrix of the equivalent continuous-time system at a steady state and is expressed as a function of the sensor measurement sampling frequencies. Based on these results, the optimal sensor sensing frequencies for each sensor on every robot can be determined and used for sensor parametrization.

3.2. Bus Systems

The specific requirements of different communication domains in vehicles have led to the use of multiple bus systems in automotive networks such as local interconnect network (LIN), J1850, CAN, Time Triggered Protocol (TTP), FlexRay, media-oriented system transport (MOST), IDB1394 and others [108]. Although the most widely used bus system for connecting sensors to the object tracking subsystem is

the CAN bus system [108] (subsection 3.2.1), the recent need for greater bandwidth and higher transmission reliability has led to the (additional) use of time-triggered bus systems such as TTP/C (subsection 3.2.2) or FlexRay [42] (subsection 3.2.3).

3.2.1. Controller Area Network

CAN transmits data serially and employs a carrier sense multiple access/collision resolution where bit arbitration is used in order to avoid collisions [111].

If the bus is free, any node may begin to transmit messages, each message starting with a message identifier. If two or more nodes begin sending messages at the same time, the message with the dominant message identifier will overwrite other nodes' message identifiers, so that after the arbitration on the message identifier only the dominant message remains and is received by all nodes. The message identifier of the dominant message is then checked by the receiving nodes in order to determine whether the message is relevant.

After the dominant has been received the messages whose message identifier has been overwritten by the message identifier of the dominant message are again tried to be sent by the respective node by starting to put their message identifiers on the bus. Due to the fact that a message can only be sent if the bus is free and no message with a dominant message identifier has to be sent, the transmission of a message can be arbitrarily delayed and is thus nondeterministic.

3.2.2. TTP/C

The TTP/C protocol uses a time division multiple access (TDMA) scheme. Time is partitioned into a sequence of slots of varying length. Each slot is statically assigned to a node of a cluster, during which the node is allowed to send its message. A TDMA round is formed by the sequence of slots and a sequence of TDMA rounds forms a cluster cycle [78]. The schedule defining the slot assignment is stored in the Message Descriptor List within the communication controller. As the schedule is defined and known to all nodes *a priori* the timely transmission of all messages can be guaranteed and is thus deterministic.

In order to provide distributed-clock synchronization, each node measures the difference between the *a priori* known and the observed arrival time of a correct message to learn about the difference between the sender's clock and the receiver's clock. A fault-tolerant average algorithm uses this information to periodically calculate a correction term for the local clock so that the clock is kept in synchrony with all other clocks of the cluster.

Furthermore TTP/C uses a membership service in order to arrive at an agreed view which nodes of the cluster have failed.

3.2.3. FlexRay

Every FlexRay round has a static and optionally a dynamic transmission segment. The static segment uses a TDMA scheme where every node has statically assigned transmission slots of fix length. In the dynamic segment, every message is assigned a unique identifier (similar to CAN) and messages are sent in increasing identifier order according to a flexible time division multiple access (FTDMA) scheme [22].

The clock-synchronization for nodes in a FlexRay cluster is based on macroticks and microticks. At a synchronization point every node calculates the difference between a global time base and its local clock in microticks. Having calculated the difference the node corrects its clock offset and corrects the number of microticks that form a macrotick in order to diminish the clock drift.

3.3. Object Tracking

As an object tracking subsystem processes object state observations provided by sensors and provides RT images of the object states to the feature service subsystem, the fusion of object state observations and related processes are usually triggered by incoming measurements and the demand for outgoing RT images of the object states.

Often, the provision of RT images to the feature service subsystem is fixed, because the employed control loop demands cyclic updates. Therefore, special focus is placed on the processing of incoming object state observations. In this regard, the multitude of proposed approaches for processing ISMs and OOSMs are discussed in subsection 3.3.1 and subsection 3.3.2.

In subsection 3.3.3 the research regarding the provision of the feature service subsystems with RT images is discussed.

Subsection 3.3.4 presents the research in the field of performance evaluation.

3.3.1. Processing of In-Sequence Measurements

If the time-stamp of an object state observation is more recent than the time instant which the associated object state represented before the prediction, the corresponding measurement is classified to be in-sequence. If the object state observation is not more recent than the instant which the associated object state represented before a retrodiction, the corresponding measurement is classified as out-of-sequence measurement.

3.3.2. Processing of Out-of-Sequence Measurements

The following techniques developed to deal with OOSM have been extensively explored in research throughout the fusion community:

1. The BUFF approach, which is based on storing measurements in a measurement buffer. In the buffer, the measurements are sorted chronologically and the oldest information is provided for fusion; and
2. The ADVA approach, where received OOSMs are directly fused using advanced algorithms, which exploit the correlation between the actual KF state and the object state observations which arrive too late.

Measurement Buffering

Kaempchen et al. [69] discuss the maximum latency (here defined as the time difference between the instant of measurement fusion and the measurement time-stamp) that arises when the BUFF approach is used to guarantee the fusion of chronologically ordered measurements.

It is differentiated between situations where only the maximum observation pre-processing time, $\max(\Delta t_{\text{PT}}^{\text{sens}_m})$, is known and situations where the observation pre-processing time, $\Delta t_{\text{PT}}^{\text{sens}_m}$, as such is known.

The time needed to process these object state observations will usually depend on the complexity of the surrounding environment, i.e., the number of object state observations and the number of possible associations. In peak load scenarios, the increasing computational load which is due to the increasing number of tracked objects may reach a critical level. Thereupon, the time during which the incoming measurements have to be kept in a buffer before they can be processed constantly increases.

A very simple solution to avoid this increase in computational load is to single out targets in regard to their relevance for the feature service subsystem. Thus, in very complex scenarios, a maximum number of objects would be tracked in the object tracking subsystem and the remaining object states that cannot be associated thereto would be discarded.

Advanced Algorithms

There are several ADVA approaches that deal with one-lag and multi-lag delays, filtering and tracking, linear and non-linear systems as well as single-model and multi-model systems.

In the following discussion, t_κ refers to the OOSM time stamp and t_k refers to the time stamp of the measurement which updated the fusion before the OOSM was received.

Larsen et al. present a suboptimal multi-lag filtering algorithm for linear systems [87]. If a measurement is expected to arrive out-of-sequence (OOS), a correction term derived from object state observations ECMs and object state vector ECM is set up after the last measurement representing the surrounding environment at a time point before t_κ is fused. Said correction term is then updated whenever

measurements are fused until the OOSM is available. As soon as the delayed measurement is available, the correction term is used to update the current object state estimate with the delayed measurement.

Bar-Shalom presents an optimal one-lag tracking algorithm for linear systems [7]. The delayed measurement is incorporated by computing the update of an object state at time point t_k with the residual of the OOSM and the retrodicted state to the time point t_κ as well as the covariance matrices between the object states at t_k and t_κ . In [13, 9], Bar-Shalom et al. extend the presented one-lag algorithm to deal with multi-lag OOSMs by virtually compressing the information of the measurements between t_κ and t_k into one update. This approach is further extended to a multi-model approach in [8].

Mallick et al. describe an extension to the algorithm presented in [7] toward a multi-lag, single-model and a one-lag, multi-model approach [95]. In [97], Mallick et al. present a multi-lag, single-model algorithm that includes data association, likelihood computation and hypothesis management and a particle filter for OOSM treatment in [96].

Orton and Marrs present the incorporation of OOSMs with particle filters [112, 113, 114].

Zhang describes an algorithm in [145] that is stated to be the general case of [7] and [95]. She further differentiates between globally optimal solutions and solutions optimal for the information given. In [147], she extends the previously discussed algorithms and establishes a connection to the work of Challa and Wang which will be discussed herein below.

Challa and Wang present an augmented state vector, which is a temporally staggered vector consisting of the present state and past states. This enables the fusion to incorporate measurements corresponding to past states in an optimal, elegant way but is computationally enormously expensive [26] (optimal multi-lag filtering algorithm for linear systems). To overcome the computationally expensive augmented state algorithm, Challa and Wang introduce the iterated augmented state algorithm [25]. In [24], they additionally describe the use of these algorithms in scenarios with clutter. Furthermore, Wang and Challa extend their algorithm towards an interacting multiple-model approach in [140].

In [3], Anxi et al. present a unified suboptimal one-lag, multi-lag and mixed-lag ADVA (i.e. a sequence with two OOSMs 1 and 2 with $t_{\kappa_1} < t_{\kappa_2}$ and $t_{k_1} > t_{k_2}$).

Julier and Uhlmann present an algorithm for the incorporation of OOSMs in situations where t_κ is not known exactly but can be statistically characterized [68]. The time-stamp uncertainty is directly accommodated into the object state observation ECM so that consistency is always maintained using the covariance union (CU) algorithm [133].

In [143], Wen et al. propose a recursive algorithm based on [7] by using a sequential fusion technique for multiple OOSMs.

Besada-Portas et al. [16] propose two algorithms for processing OOSMs, the first

for linear systems being based on a junction tree (JT) algorithm [88] and a second algorithm for non-linear systems.

Another approach is the distributed multi-rate multiple-model fusion algorithm by Hong et al. This algorithm computes a proper target update-rate for every object, depending on the dynamic model that currently describes the target best. Targets with no or only marginal dynamics will be updated less frequently in contrast to highly dynamical targets being updated at the highest possible rate. In [60, 61, 62] Hong et al. extend their algorithm to out-of-sequence scenarios. Using the object state and object state covariance matrix corresponding to dynamical models with low update rate, the incorporation of OOSM is simplified, because of the smaller number of updates that have been computed between κ and k (see subsection 3.3.2).

In [81], Koplín and Elmenreich analyze the effect of systematic errors such as offsets in regard to the treatment of OOSMs. The results show that measurement buffering is advantageous over ADVA algorithms, if sensors have offsets in the value domain that have different algebraic signs.

3.3.3. Provision of the Feature Service Subsystems with Real-Time Images

In [154, 153, 152, 151] Mauthner et al. optimize a time-triggered, KF based, multi-sensor fusion system, used as an environmental perception platform for advanced driver assistance systems while satisfying constraints that are typical of a safety related application. The papers evaluate how the sensor, bus and fusion schedules influence the accuracy of the RT image of the surrounding environment, provided to a safety related feature service subsystem. The papers show that a system schedule with minimal overall system execution time (OSET) achieves the most accurate RT images of the object states.

3.3.4. Performance Evaluation

Mallick and Marrs [98] compare multi-lag ADVA approaches for linearized systems based on a PF algorithm [112] and a KF algorithm [145] and show that although the PF based algorithm is suboptimal, its results are comparable to the KF based algorithm.

In [128], Tasoulis et al. compare the ADVA methods presented in [87, 94, 147] on the basis of the mean absolute difference between state estimates of an ADVA approach and state estimates of a baseline Kalman filter estimated where a BUFF approach is used to treat OOSMs. It is argued that not all OOSMs should be incorporated, as the performance of the system degrades gracefully and the effort for OOSM incorporation can thus be saved.

In [143], Wen et al. compare a newly proposed ADVA approach with the ADVA approach presented in [148] on performance indexes such as RT processing capabil-

ity, storage, applicability, fault tolerance and fusion accuracy. Although the fusion accuracy of the newly proposed algorithm is smaller, it is argued that the newly proposed algorithm is preferable due to its superiority in regard to the other performance indexes.

In [32], Danu et al. compare track-to-track fusion, tracklet fusion and associated measurement fusion on performance metrics such as track completeness, track accuracy and track consistency.

Besadas-Portas et al. compare the algorithms of [7, 95, 110, 13, 24, 145, 146] in [16].

3.4. Feature Services

There are various classifications of feature services provided in ADASOTs [134]. The most obvious seems to be the classification into services that assist by controlling the longitudinal motion (subsection 3.4.1) and services that assist by controlling the lateral motion [63] (subsection 3.4.2). This is due to the fact that services that assist with the longitudinal and lateral control usually employ different types of sensors and actuators (subsection 3.4.3).

3.4.1. Longitudinal Control

Longitudinal Control is provided in services like adaptive cruise control (ACC) and AEB.

ACC automatically adjusts the vehicle speed in order to maintain a safe headway distance to vehicles in the same lane. A typical sensor for ACC is a millimeter-wave radar system operating in the 77 GHz frequency range that measures within a longitudinal range of 1 m to 150 m and within an azimuthal range of $\pm 8^\circ$ [1].

As the demand for dependability of an AEB system is much greater than the demand imposed on an ACC system, the AEB system usually employs multiple heterogeneous sensors, e.g., a laser scanner and a stereo vision system [84]. The combination of these sensors becomes necessary as a laser scanner may produce false alarms when the laser beams collide with the road surface because of road geometry and vehicle pitching. The stereo vision system however, does not provide exact velocities and distances because of the size of the back-projected area in the road scene corresponding to a pixel in the image [85].

3.4.2. Lateral Control

Lateral control is provided in services like ALK and automated lane change maneuver [23].

An ALK system is based on lane information which is usually provided by a vision system [48] or a laser scanner [74] and controls the steering wheel such that the vehicle remains within the detected lane markings as long as the driver does not overrule the system.

A system for automated lane change maneuver uses information about lanes and obstacles around the vehicle in order to navigate the vehicle along a desired path that includes different possible lanes without risking a collision with obstacles. Such a system usually employs heterogeneous sensors in order to cover the high azimuthal detection range necessary for obstacle detection and to increase system reliability [82].

3.4.3. Integrated Longitudinal and Lateral Control

Integrated approaches are currently being developed for ADASOT operating at low speed, such as parking assistants (PA) where the free parking space is detected by vision, radar and laser scanners and the vehicle is autonomously parked [70].

Besides the field of ADASOTs the development of integrated approaches has received great attention in the military field, e.g., in the DARPA Grand Challenge [130]. However these approaches are quite different from the development of current ADASOT due to the tremendous amount of used sensor resources and the environment wherein the vehicles operate.

3.5. Chapter Summary

In this chapter, the related work regarding sensors, bus systems, object tracking and feature services employed in state-of-the-art ADASOTs has been discussed.

Although many state-of-the-art do not provide the possibility to be controlled but merely operate independently from other subsystems, recent development points in the direction of intelligent sensor networks in which sensors can be triggered, parameterized and allocated in accordance with the changing demands of a feature service subsystem. The development in the domain of bus systems points in a similar direction, where event-triggered bus systems are replaced by time-triggered bus systems, whose deterministic nature eases control of the connected sensors and provides a higher level of dependability.

Within an object tracking subsystem, the fusion of incoming object state observations using algorithms such as the Kalman filter is straightforward, as long as the object state observations arrive in-sequence. Whenever object state observations arrive out-of-sequence, the object state observations can either be discarded, chronologically reordered by buffering, or fused using advanced algorithms. As all object state observations from a particular sensor may arrive out-of-sequence, discarding this information is not feasible and therefore recent research has focused on

3. Related Work

the development of advanced algorithms as an elegant and efficient way to deal with them.

A snapshot of the environment taken by the sensors is preprocessed in the sensor, transmitted from the sensor to the object tracking subsystem, and fused with predicted object states. Due to delays caused by the time which is required for object state observation preprocessing, transmission, and fusion, the fused object states do not represent the present state of the environment, as the environment is constantly evolving. This discrepancy can be diminished by predicting RT images of the object states on the basis of the fused object states. Therefore minimizing the system OSET ameliorates the fusion results.

The feature services of ADASOT can be classified into systems assisting in the longitudinal or lateral control of a vehicle and systems where lateral and longitudinal control are integrated. Whereas there are various approaches for services like ACC, AEB, ALK, autonomous lane change, and PA, the development tends toward multi-sensor approaches due to perception range coverage and dependability requirements.

4. Numerical Simulation

This chapter's intent is to provide an answer to the research question raised in section 1.2 through a numerical simulation of a model of state-of-the-art multi-sensor ADASOTs (section 4.1) and a model of time-triggered multi-sensor ADASOTs (section 4.2) in a generalized environment (section 4.3) in regard to a well-defined performance measure (section 4.4).

The results of the numerical simulation are presented in section 4.5 and analyzed in section 4.6. Based thereon, a conclusion is drawn in section 4.7.

The chapter is concluded by a summary (section 4.8).

4.1. Model of a State-of-the-Art Multi-Sensor Advanced Driver Assistance System

In the further, it is assumed that the here treated multi-sensor ADASOT consists of two sensors (subsection 4.1.1), an object tracking subsystem (subsection 4.1.3), and a feature service subsystem, interconnected via a bus system (subsection 4.1.2), as schematically depicted in Figure 4.1.

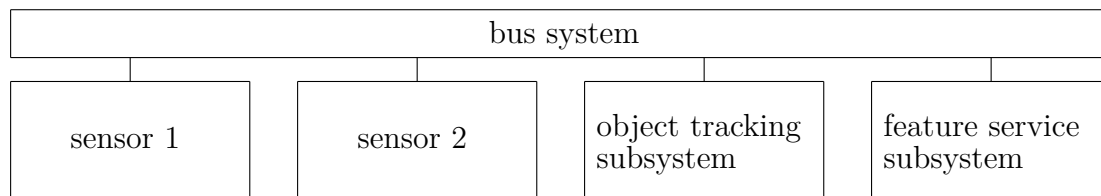


Figure 4.1.: Model of a multi-sensor ADASOT

The state-of-the-art model schedule is discussed in subsection 4.1.4.

4.1.1. Sensors

In an automotive environment, many obstacle detection systems achieve good results with a combination of active sensors such as radars and lasers and passive sensors such as cameras [35]. Thus, sensor 1 is an abstraction of an automotive vision sensor providing position observations, \bar{z}_1 , and sensor 2 is an abstraction of an automotive

radar or laser sensor providing position and velocity (Doppler) observations, \vec{z}_2 , which are calculated with reference to a Cartesian coordinate frame.

Every object state observation vector is assumed to be decomposable into quantities of the true object state vector \vec{x} and a Gaussian distributed error vector \vec{r} with zero mean [117] as shown in equation 4.1 and equation 4.2.

$$\vec{z}_1 = H_1 \cdot \vec{x} + \vec{r}_1 = \begin{pmatrix} x_1 + r_{x_1} \\ y_1 + r_{y_1} \end{pmatrix} \quad (4.1)$$

$$\vec{z}_2 = H_2 \cdot \vec{x} + \vec{r}_2 = \begin{pmatrix} x_2 + r_{x_2} \\ y_2 + r_{y_2} \\ v_{x_2} + r_{v_{x_2}} \\ v_{y_2} + r_{v_{y_2}} \end{pmatrix} \quad (4.2)$$

The object state observation ECMs, $R_1 = E(\vec{r}_1 \vec{r}_1')$ and $R_2 = E(\vec{r}_2 \vec{r}_2')$, are assumed to consist of position independent variance values (for accuracy of vision sensors see [109, 100, 99], for accuracy of radar or laser sensors see [43, 46, 54], for conversion of range and bearing measurements to Cartesian coordinate measurements see [36]).¹

$$R_1 = \begin{pmatrix} 1 \text{ m}^2 & \pm 0.001 \text{ m}^2 \\ \pm 0.001 \text{ m}^2 & 0.01 \text{ m}^2 \end{pmatrix} \quad (4.3)$$

$$R_2 = \begin{pmatrix} 0.01 \text{ m}^2 & \pm 0.001 \text{ m}^2 & 0 & 0 \\ \pm 0.001 \text{ m}^2 & 1 \text{ m}^2 & 0 & 0 \\ 0 & 0 & 0.01 \frac{\text{m}^2}{\text{s}^2} & \pm 0.001 \frac{\text{m}^2}{\text{s}^2} \\ 0 & 0 & \pm 0.001 \frac{\text{m}^2}{\text{s}^2} & 1 \frac{\text{m}^2}{\text{s}^2} \end{pmatrix} \quad (4.4)$$

The object state observation ECMs are assumed to be slightly higher than specified in the cited papers. This is due to the fact that the specified precision of both sensors refers to measuring coordinates of points or edges of a non-planar contour of a vehicle.

However, in scenarios where the measured coordinates of points or edges are used for estimating a vehicle's geometrical center, observations of the vehicle's dimensions such as width and length are additionally required [125]. When estimating the vehicle's geometrical center using width and length observations, the potential inaccuracy of the width and length observations has to be taken into account.

Furthermore, the reflection of a laser scanner or radar beam on a vehicle contour or the edges that a vision sensor detects when analyzing a vehicle contour may shift during a maneuver due to changing aspect angles. Said shifting adds further

¹Although it is clear that real sensors will provide position dependent variance values, the assumption on object position independent variance values seems feasible as the comparison of both models has to be independent of random object position trajectories.

uncertainty to the estimation of the vehicle's geometrical center and has to be taken into account in the tracking process, for example, by increasing the object state observation ECMs.²

The PTs of the sensors are assumed to be dependent on the complexity of the surrounding environment as discussed in subsection 2.2.3. It is assumed, however, that there are upper bounds for the sensor PTs as each sensor does not detect more than a maximum number of objects (see also subsection 2.2.4). Accordingly, the PT of sensor 1 is assumed to vary within a range of $c \cdot 160 \text{ ms}$ to 160 ms and the PT of sensor 2 is assumed to vary within a range of $c \cdot 80 \text{ ms}$ to 80 ms due to changes in the complexity of the environment [126], where c accounts for different complexity variances as modeled in subsection 4.3.1.³

Furthermore, it is assumed that the sensors do not continuously provide object state observations, but tend to lose an object from time to time, which can result, for example, from object occlusions, difficulties in the observation preprocessing or a badly working association process. The recognition ability is modeled for both sensors independently by a Markov process with binary states $j = 0$ and $j = 1$

$$\vec{j} = \begin{pmatrix} 0 \\ 1 \end{pmatrix} \quad (4.5)$$

where 0 indicates that a sensor has not observed an object and 1 indicates that a sensor has observed an object, the Markov process being governed by the following transition probability matrix (see also [29])

$$J = \begin{pmatrix} 0.975 & 0.025 \\ 0.01 & 0.99 \end{pmatrix} \quad (4.6)$$

4.1.2. Bus System

The bus system within the state-of-the-art model is assumed to be a CAN which operates event-triggered using a carrier sense multiple access/collision resolution

²In fact, the direct increase of the object state observation ECMs is selected here due to the association process not being explicitly modeled but only taken implicitly into account. However, when explicitly modeling the association process, the additional uncertainty would be part of the innovation ECMs, when adding the object contour parameters to the object state and estimating an object state observation using a state space to measurement space transition matrix, H , which takes into account the added object contour parameters.

³In contrast to other sensor parameters, the here specified variance of PT could not be supported by references to recent research publications or sensor manuals due to the fact that in many research publications the PT is either assumed to be constant and equal to the sensor CT [49] or not modeled at all. Furthermore, in many sensor manuals only the sensor CT is specified without an exact or convincing explanation of the sensor PT. Therefore, the specified variation of PT described herein is only supported by the author's experience with real sensors and should be critically looked upon when using the author's results.

scheme. Furthermore, it is assumed that the CAN is exclusively used for transmitting object state observations from the sensors to the object tracking subsystem.⁴

The time for transmitting the object state observation vectors from a sensor to the object tracking subsystem is typically smaller than 2 ms. For sake of simplicity it is assumed in the further that the transmission time is $\Delta t_{\text{PT}}^{\text{tb}} = 2 \text{ ms}$.⁵

4.1.3. Object Tracking Subsystem

It is further assumed that associated in-sequence object state observations and predicted images of the object states are fused by a KF algorithm (see section 2.5) using a white-noise jerk model [118] with

$$\vec{x} = \begin{pmatrix} x \\ y \\ v_x \\ v_y \\ a_x \\ a_y \end{pmatrix}, \quad (4.7)$$

$$e^{F\Delta t} = \begin{pmatrix} 1 & 0 & \Delta t & 0 & \frac{\Delta t^2}{2} & 0 \\ 0 & 1 & 0 & \Delta t & 0 & \frac{\Delta t^2}{2} \\ 0 & 0 & 1 & 0 & \Delta t & 0 \\ 0 & 0 & 0 & 1 & 0 & \Delta t \\ 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 \end{pmatrix} \quad (4.8)$$

and

$$Q = \begin{pmatrix} \frac{\Delta t^5}{20} & 0 & \frac{\Delta t^4}{8} & 0 & \frac{\Delta t^3}{6} & 0 \\ 0 & \frac{\Delta t^5}{20} & 0 & \frac{\Delta t^4}{8} & 0 & \frac{\Delta t^3}{6} \\ \frac{\Delta t^4}{8} & 0 & \frac{\Delta t^3}{3} & 0 & \frac{\Delta t^2}{2} & 0 \\ 0 & \frac{\Delta t^4}{8} & 0 & \frac{\Delta t^3}{3} & 0 & \frac{\Delta t^2}{2} \\ \frac{\Delta t^3}{6} & 0 & \frac{\Delta t^2}{2} & 0 & \Delta t & 0 \\ 0 & \frac{\Delta t^3}{6} & 0 & \frac{\Delta t^2}{2} & 0 & \Delta t \end{pmatrix} \cdot q. \quad (4.9)$$

The time required for fusing all object state observations from one sensor is assumed to be dependent on the complexity of the environment as every additional object increases the required fusion time.

⁴The assumption of exclusive use is idealistic but seems to be reasonable in regard to this thesis focusing on a mean performance comparison, as the problem of unpredictable transmission delays that may occur when using an event-triggered bus system is more relevant for a worst case performance comparison.

⁵This assumption seems feasible since the sensor PTs are much longer than the transmission time.

As the maximum number of object state observations is assumed to be restricted, there exists an upper bound (UB) for the time required to fuse ISM, $\Delta t_{\text{PT}}^{\text{fusISM}} \leq UB_{\text{fus}}$. In order to account for different hardware resources available, the upper bound is assumed to range between 2 ms and 25 ms, $UB_{\text{fus}} = \{2, 5, 10, 15, 20, 25\}$ ms. Furthermore, it is assumed that $\Delta t_{\text{PT}}^{\text{fusISM}}$ varies within a range of $\text{ceil}(c \cdot UB_{\text{fus}})$ to UB_{fus} depending on the complexity of the surrounding environment where c accounts for different complexity variances as modeled in subsection 4.3.1.

The occurrence of OOSMs is either dealt with by a BUFF approach as described in subsection 2.5.6 or an ADVA approach as presented by Bar-Shalom in [9]. The ADVA approach is assumed to demand additional processing time following $\Delta t_{\text{PT}}^{\text{fusOOSM}} = \text{ceil}\left(\frac{3}{2}\Delta t_{\text{PT}}^{\text{fusISM}}\right)$.

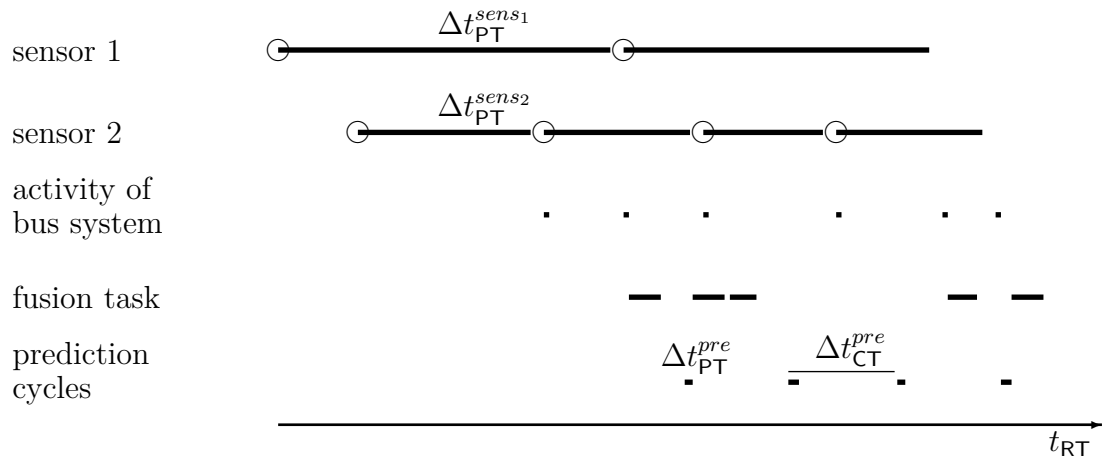
Furthermore, the object tracking subsystem does not buffer more than two object state observations in order to avoid an object state observation jam. If an object state observation from sensor 1 is received at the object tracking subsystem and one other object state observation from the same sensor awaits its fusion, the object tracking subsystem discards the older object state observation from sensor 1. If an object state observation from sensor 2 is received at the object tracking subsystem and two other object state observation from the same sensor await their fusion, the object tracking subsystem discards the older object state observations from sensor 2.

At predefined points in time, the object tracking subsystem starts to predict images of the object states in order to generate RT images of the object states which are provided to the feature service subsystem (see subsection 3.3.3). The time required for predicting RT images of the object states is assumed to be $\Delta t_{\text{PT}}^{\text{pre}} = \text{ceil}\left(\frac{1}{3}\Delta t_{\text{PT}}^{\text{fusISM}}\right)$. The time difference between the instant where the most recent scan or frame which has been fused to the image of the object states has been taken and the instant where RT images of the object states are delivered to the feature service subsystem is the interval over which the images of the object states are being predicted in order to produce RT images of the object states, said interval being referred to as OSET throughout this thesis.

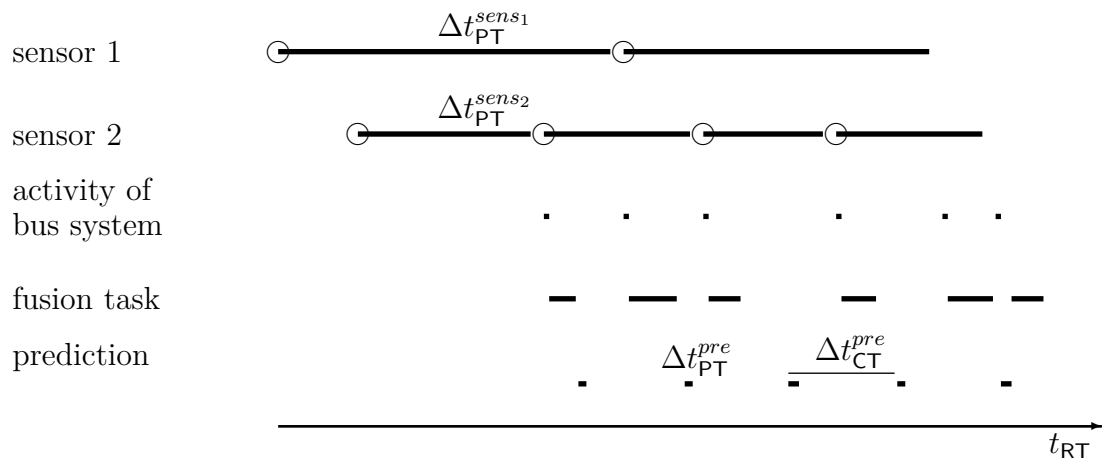
The RT images of the object states are then transmitted to the feature service subsystem. It is assumed that the control loop performed within the feature service subsystem has a frequency of 25 Hz which is a typical value for vehicle control [86, 58, 59]. It is further assumed that at the beginning of each control loop, the feature service subsystem has to be provided with RT images of the the object states.

4.1.4. State-of-the-Art Model Schedule

Figures 4.2(a) and 4.2(b) visualize the schedule of the state-of-the-art model for the BUFF or ADVA approach, each process being visualized by a horizontal bar.



(a) BUFF approach



(b) ADVA approach

Figure 4.2.: State-of-the-art model schedule

Within the state-of-the-art model as depicted in Figures 4.2(a) and 4.2(b), the two sensors (“sensor 1” and “sensor 2”) measure with CTs, Δt_{CT}^{sens1} and Δt_{CT}^{sens2} , that vary over RT and are equal to the corresponding sensor PTs, Δt_{PT}^{sens1} and Δt_{PT}^{sens2} . The sensor PTs are not constant due to the complexity variance as described in subsection 4.3.1. The PHs of the sensors are uncontrolled as the internal sensor clocks run free without being synchronized.

The transmission of an object state observation is indicated in Figure 4.2(a) and Figure 4.2(b) by bars labeled “activity of bus system”.

As soon as object state observations are received by the object tracking subsystem and no task is currently processed, the object position observations can be fused with associated images of the object states (“fusion task”) hereby taking into account the particulars of OOSM as described in subsection 2.5.6.

In Figure 4.2(a), the received object state observations are sorted chronologically within an object state observation buffer which allows the fusion of all object state observations without the use of advanced algorithms. However, as can be seen from Figure 4.2(a), the buffering of object state observations adds additional delays to the system.

In Figure 4.2(b) the received object state observations are fused as soon as sufficient processing resources are available. The fusion process task interval, Δt_{PT}^{fusISM} , varies as described above.

Every Δt_{CT}^{pre} , RT images of the object states are generated (“prediction cycles”) and transmitted over the bus system to the feature service subsystem.

The parameters that specify the state-of-the-art configurations are

$$\mathcal{S} = \{\text{BUFF/ADVA}\}. \quad (4.10)$$

4.2. Proposed Paradigm Shift

In order to avoid tedious repetitions, the following subsections focus on the differences between the state-of-the-art and the time-triggered model regarding sensors (subsection 4.2.1), bus system (subsection 4.2.2), object tracking subsystem (subsection 4.2.3), and schedule (subsection 4.2.4).

4.2.1. Sensors

The sensors in the proposed time-triggered multi-sensor ADASOT have fixed sensor CTs that are equal to the maximum sensor PTs, $\Delta t_{PT}^{sens1} = 160 \text{ ms}$ and $\Delta t_{PT}^{sens2} = 80 \text{ ms}$, i.e., the sensors are scheduled to account for the WCET of observation preprocessing [73]. The sensor PH, $\Delta t_{PH,1,1}^{sens2}$, can be controlled and chosen by a system designer in order to arrive at an optimal schedule.

4.2.2. Bus System

The bus system within the time-triggered model is assumed to be time-triggered using a TDMA scheme, which results in well defined transmission slots and bounded transmission jitter. Δt_{CT}^{ttb} is chosen to be a factor of Δt_{CT}^{sens1} and Δt_{CT}^{sens2} and has the typical value of 10 *ms* [42, 55].

The time for transmitting object state observation vectors from a sensor to the object tracking subsystem is assumed to be $\Delta t_{PT}^{ttb} = 2$ *ms* [42].

Please note that the transmission delays introduced by the event-triggered bus system as described in subsection 4.1.2 and the time-triggered bus system are assumed to be equal. This assumption seems feasible as the focus of this thesis is not on any particular event-triggered or time-triggered bus system but on the paradigm shift toward time-triggered ADASOTs.

4.2.3. Object Tracking Subsystem

The object tracking subsystem fuses the incoming object state observations with associated images of the object states, taking into account the particulars of OOSM processing.

The time-triggered model schedule is set up taking into account the upper bound for the fusion process task interval UB_{fus} which is assumed to vary between 2 *ms* and 25 *ms*, depending on the hardware resources provided for the object tracking subsystem.

The occurrence of OOSM is either dealt with by a BUFF approach as described in subsection 2.5.6 or an ADVA approach as presented by Bar-Shalom in [9].

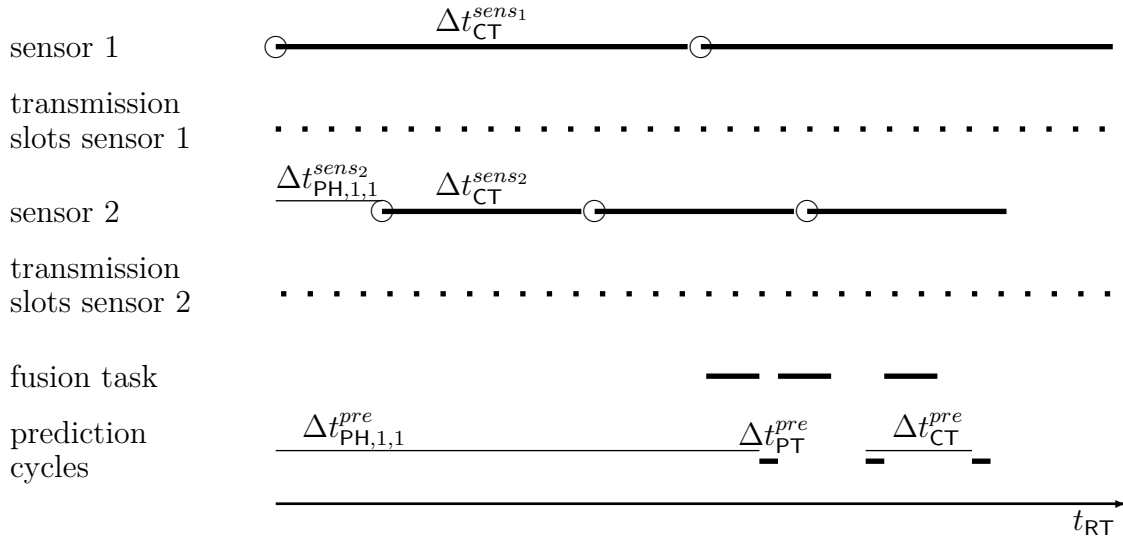
At predefined points in time, the object tracking subsystem starts to predict images of the object states in order to generate RT images of the object states (see subsection 3.3.3). The scheduling of the prediction can be chosen by a system designer in order to arrive at an optimal schedule.

The RT images of the object states are then transmitted to the feature service subsystem.

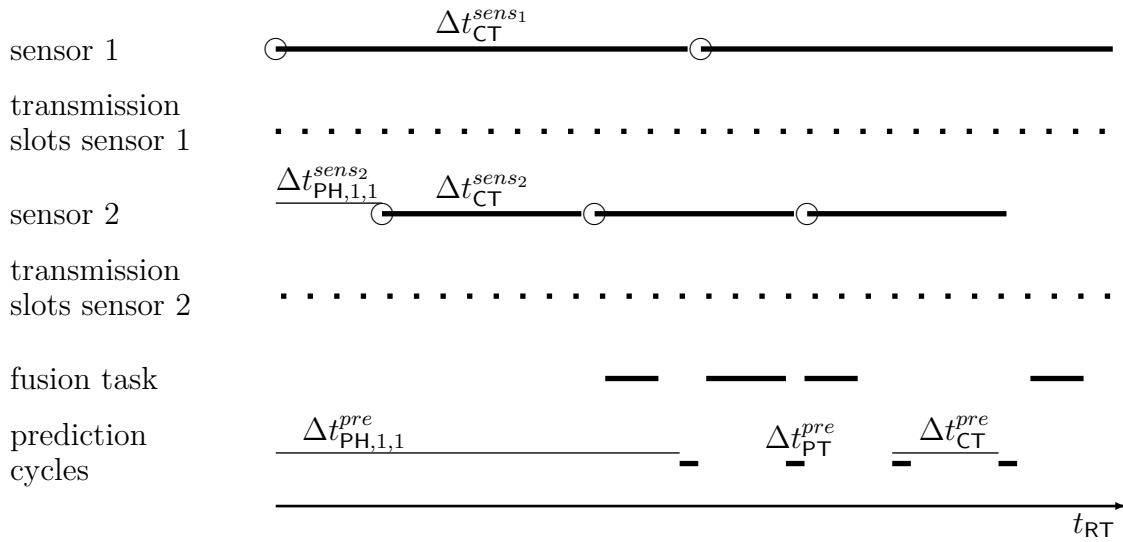
4.2.4. Time-Triggered Model Schedule

Figures 4.3(a) and 4.3(b) depict an unsynchronized time-triggered model schedule for the BUFF approach or the ADVA approach and Figure 4.4 depicts a synchronized time-triggered model schedule.

Within the time-triggered model schedules as depicted in Figures 4.3(a), 4.3(b) and 4.4, the two sensors have constant CTs, $\Delta t_{CT}^{sens1} = 160$ *ms* and $\Delta t_{CT}^{sens2} = 80$ *ms* (observation preprocessing being referred to by the label “sensor 1” respectively “sensor 2”). The PH between sensor 1 and sensor 2 is chosen to be $\Delta t_{PH,1,1}^{sens2} = \Delta t_{CT}^{pre} = 40$ *ms* in Figure 4.3 and $\Delta t_{PH,1,1}^{sens2} = 0$ *ms* in Figure 4.4.



(a) BUFF approach



(b) ADVA approach

Figure 4.3.: Unsynchronized time-triggered model schedule

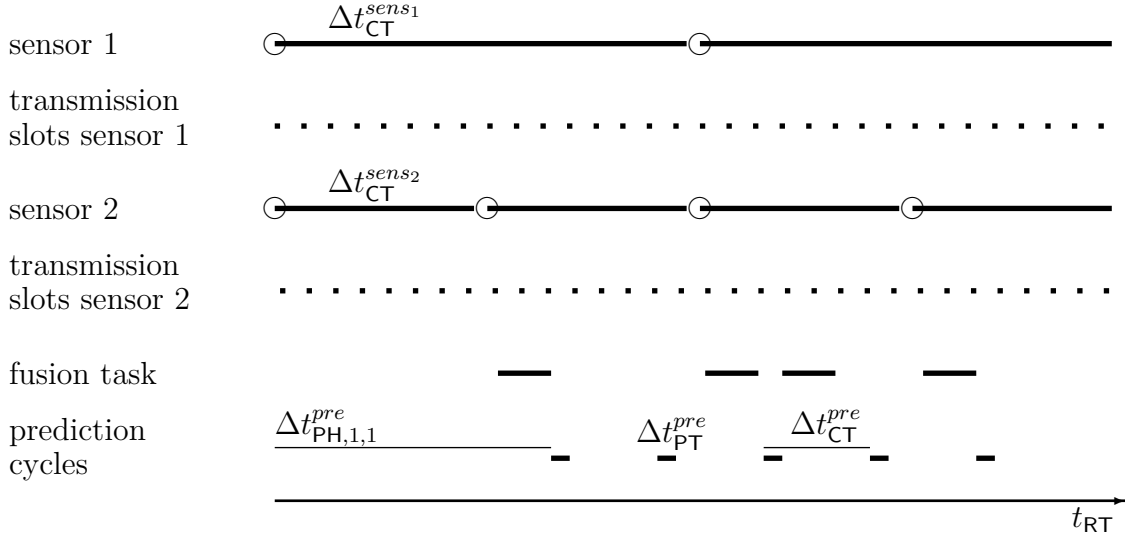


Figure 4.4.: Synchronized time-triggered model schedule

The transmission slots of sensor 1 and sensor 2 in Figures 4.3(a), 4.3(b) and 4.4 are so scheduled that the object state observations of sensor 1 are transmitted without any further delay, $\Delta t_{PH,1,1}^{ttb1} = 0$, and that the object state observations of sensor 2 are transmitted after a delay of 2 ms, $\Delta t_{PH,1,1}^{ttb2} = 2 \text{ ms}$.

In Figure 4.3(a), the received object state observations are sorted chronologically within an object state observation buffer, which allows the fusion of all object state observations without the use of advanced algorithms. However, as can be seen from Figure 4.3(a), the buffering of object state observations adds additional delays to the system.

In Figure 4.3(b), the received object state observations are fused as soon as sufficient processing resources are available, using an ADVA approach for OOSMs.

In Figure 4.4, the received object state observations are fused as soon as sufficient processing resources are available, as no OOSM problem can occur.

Every Δt_{CT}^{pre} , RT images of the object states are predicted from the fused images of the object states and then transmitted over the bus system to the feature service subsystem (“prediction cycles”). For the time-triggered unsynchronized ADVA configuration, the prediction cycle PH is chosen to be

$$\Delta t_{PH,1,1}^{pre} = \Delta t_{PT}^{fusOOSM} + 2 \cdot \Delta t_{PT}^{ttb}. \quad (4.11)$$

For the time-triggered unsynchronized BUFF configuration the prediction cycle PH is chosen to be

$$\Delta t_{PH,1,1}^{pre} = 2 \cdot \Delta t_{PT}^{fusISM} + \Delta t_{PT}^{ttb} \quad (4.12)$$

for $2 \cdot \Delta t_{PT}^{fusISM} + \Delta t_{PT}^{pre} < \Delta t_{CT}^{pre}$ and

$$\Delta t_{PH,1,1}^{pre} = \Delta t_{PT}^{fusISM} + \Delta t_{PT}^{ttb} \quad (4.13)$$

for $2 \cdot \Delta t_{PT}^{fusISM} + \Delta t_{PT}^{pre} > \Delta t_{CT}^{pre}$.

For the time-triggered synchronized configuration the prediction cycle PH is chosen to be

$$\Delta t_{PH,1,1}^{pre} = 2 \cdot \Delta t_{PT}^{fusISM} + \Delta t_{PT}^{ttb} \quad (4.14)$$

for $2 \cdot \Delta t_{PT}^{fusISM} + \Delta t_{PT}^{pre} < \Delta t_{CT}^{pre}$ and

$$\Delta t_{PH,1,1}^{pre} = \Delta t_{PT}^{fusISM} + 2 \cdot \Delta t_{PT}^{ttb} \quad (4.15)$$

for $2 \cdot \Delta t_{PT}^{fusISM} + \Delta t_{PT}^{pre} > \Delta t_{CT}^{pre}$.

Due to the deterministic nature of the time-triggered approach and the fact that the jitter of all processes is assumed to be sufficiently small compared to the CTs and can therefore be neglected, the whole system schedule is defined by the constant CTs and the PHs of all processes.

The parameters that specify the time-triggered configurations are

$$\mathcal{T} = \{ \Delta t_{PH,1,1}^{sens2}, \Delta t_{PH,1,1}^{ttb1}, \Delta t_{PH,1,1}^{ttb2}, \Delta t_{PH,1,1}^{pre}, \text{BUFF/ADVA} \}. \quad (4.16)$$

4.3. Model of the Environment

The environment is modeled in regard to two aspects, the variance of its complexity (subsection 4.3.1), i.e., how the PTs of the sensors and the object tracking subsystem depend on the environment, and the process noise (subsection 4.3.2) which is a measure of how good the employed KF prediction model describes reality.

4.3.1. Variance of the Complexity of the Environment

The changes in the complexity of the environment are modeled by a random walk with step size 1 *ms*. The Markov processes regarding the varying object observation preprocessing times and the varying object observation fusion time are modeled by Markov chains comprising states from $c \cdot 160$ *ms* to 160 *ms*, $c \cdot 80$ *ms* to 80 *ms*, and $c \cdot \{2, 5, 10, 15, 20, 25\}$ *ms* to $\{2, 5, 10, 15, 20, 25\}$ *ms* here exemplary shown for $c = 0.5$.⁶

⁶States and transition probability matrices for $c = \{0.6, 0.7, 0.8, 0.9\}$ can be found in appendix A

$$\vec{m}_{c=0.5} = \begin{pmatrix} \frac{\Delta t_{PT}^{sens1}}{ms} & \frac{\Delta t_{PT}^{sens2}}{ms} & \frac{\Delta t_{PT}^{fus}}{ms} \\ 80 & 40 & \{1, 3, 5, 8, 10, 13\} \\ 88 & 44 & \{1, 3, 5, 9, 11, 14\} \\ 96 & 48 & \{1, 3, 6, 9, 12, 15\} \\ 104 & 52 & \{1, 4, 6, 10, 13, 17\} \\ 112 & 56 & \{1, 4, 7, 11, 14, 18\} \\ 120 & 60 & \{1, 4, 7, 11, 15, 19\} \\ 128 & 64 & \{2, 4, 8, 12, 16, 20\} \\ 136 & 68 & \{2, 5, 8, 13, 17, 22\} \\ 144 & 72 & \{2, 5, 9, 14, 18, 23\} \\ 152 & 76 & \{2, 5, 9, 14, 19, 24\} \\ 160 & 80 & \{2, 5, 10, 15, 20, 25\} \end{pmatrix} \quad (4.17)$$

$$M_{c=0.5} = \begin{pmatrix} 0.5 & 0.5 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0.25 & 0.5 & 0.25 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0.25 & 0.5 & 0.25 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0.25 & 0.5 & 0.25 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0.25 & 0.5 & 0.25 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0.25 & 0.5 & 0.25 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0.25 & 0.5 & 0.25 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0.25 & 0.5 & 0.25 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0.25 & 0.5 & 0.25 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0.25 & 0.5 & 0.25 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0.5 & 0.5 \end{pmatrix} \quad (4.18)$$

4.3.2. Process Noise

The process noise of the object state evolution is assumed to be white with power spectral density, q , and to account for modeling errors such as derivatives with respect to time that are not contained in the object state vector but may be not equal zero during the prediction interval as described in subsection 2.3.1. In an automotive environment, the choice of one specific q seems to be unfeasible as q in an traffic jam environment for example, will be much smaller than q in a freeway environment.

As a result, q is assumed to vary in the range of $q = [0.01, 100] \frac{m^2}{s^5}$ due to not modeling the derivative of acceleration in the white noise jerk model of subsection 4.1.3 (see also [2, 19, 71, 34]).⁷

⁷The wide range of q is further motivated by the applicability of the results to linearized state

4.4. Performance Measure

As mentioned in section 1.2, the basis for achieving a correct ADASOT feature service is a correct assessment of the surrounding environment. The correctness of this assessment depends on the deviations between the RT images of the object states and reality, which have to be smaller than a feature specific upper bound.

Assuming that all relevant objects are detected by the sensors and that the number of ghost-objects and non-detects is negligible (otherwise the sensors would not be suited for use in ADASOTs), the mean performance (MP) of both models can be expressed by the mean ECM trace of the RT images of the object states (for ECM trace see also [18] and subsection 2.6.1). As the state-time (ST) of the images of the object states is delayed to RT due to object state observation preprocessing, transmission and fusion (see also subsection 2.2.3), it is assumed that the RT images of the object states are predicted from the ST images of the object states using the object state evolution model of the KF, which leads to

$$\text{MP} := \text{mean}_{n \in \mathbb{N}} \left(\text{trace} \left(P \left(t_{\text{RT}} | t_{\text{ST}} \right) \right) \right) \quad (4.19)$$

with $t_{\text{RT}} = \Delta t_{\text{PH},1,1}^{\text{pre}} + n \cdot \Delta t_{\text{CT}}^{\text{pre}}$ and $t_{\text{ST}} = t_{\text{ST}}(t_{\text{RT}})$.⁸

4.5. Simulation Results

Subsection 4.5.1 depicts the optimal configurations and subsection 4.5.2 depicts the best state-of-the-art configurations and the best time-triggered configurations as well as the corresponding best state-of-the-art configuration MP/ time-triggered configuration MP ratios for different regions of a parameter space spanned by the environment parameters and the upper bound for fusion processing time.

4.5.1. Optimal Configurations

Figure 4.5 depicts a three-dimensional parameter space grid spanned by the parameters for complexity variance, c , upper bound for the fusion PT, UB_{fus} , and process noise power spectral density, q . Therein, every grid point is identified by a symbol indicating the configuration which is optimal for the respective parameter set with regard to the simulated MP. The symbols are referenced by numbers 1 to 5 in the figure legend, the numbers referring to the model configurations:

space models, e.g. EKF, that have greater process noise power spectral density q as linear models due to the linearization error as shown in appendix B.

⁸Please note that the differentiation between RT and ST is crucial for this thesis and leads to a performance measure that does not only take into account the accuracy level of fused object state observations but also their accuracy interval.

4. Numerical Simulation

- State-of-the-art BUFF (1);
- State-of-the-art ADVA (2);
- Time-triggered unsynchronized BUFF (3);
- Time-triggered unsynchronized ADVA (4); and
- Time-triggered synchronized (5).

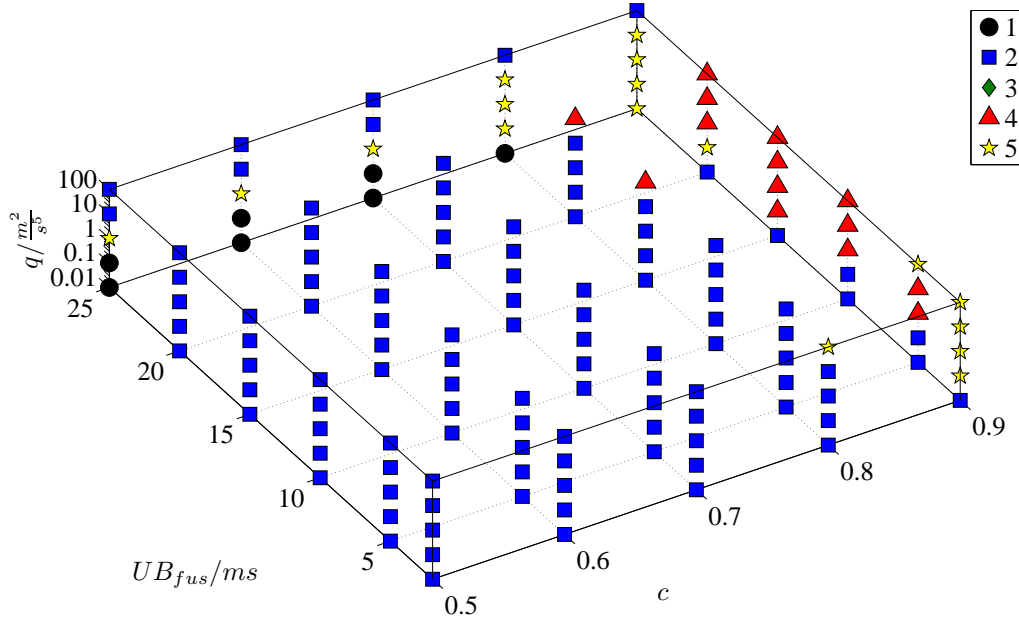


Figure 4.5.: Optimal state-of-the-art or time-triggered configurations for different points in parameter space

Figure 4.5 shows that the state-of-the-art ADVA configuration (indicated by blue squares) is optimal for most grid points in the three-dimensional parameter space grid spanned by complexity variance, upper bound for the fusion PT, and process noise power spectral density.

However, there are boundary grid points where the state-of-the-art ADVA configuration is outperformed by other configurations.

For big to medium complexity variance in combination with slow object tracking subsystem and small process noise power spectral density, the state-of-the-art BUFF configuration (indicated by black circles) yields the best results among all possible configurations.

For small complexity variance in combination with medium-slow to medium-fast object tracking subsystem and medium to high process noise, the time-triggered unsynchronized ADVA configuration (indicated by red triangles) is optimal.

The time-triggered synchronized configuration (indicated by yellow stars) is optimal for small complexity variance in combination with a fast or slow object tracking subsystem, and small to high process noise power spectral density.

It is also noteworthy that the time-triggered unsynchronized BUFF configuration is suboptimal over the whole parameter region.

4.5.2. Best State-of-the-Art and Time-Triggered Configurations

Figures 4.6, 4.7, 4.8, 4.9, and 4.10 each consist of three subfigures, two two-dimensional parameter grids and a three-dimensional figure.

In the top two-dimensional parameter grids, every grid point is identified by a symbol, the symbol indicating the respective state-of-the-art configuration that is best suited for the corresponding parameters with regard to the simulated configuration MP. The middle two-dimensional parameter grids are organized analogous to the top two-dimensional parameter grids but indicate the best suited time-triggered configurations. The two-dimensional parameter grids are spanned by the parameter for the complexity variance and the speed of the object tracking subsystem. The three-dimensional figures depict the ratio of the best state-of-the-art configurations MP to the best time-triggered configurations MP in the form of colored meshes over two-dimensional parameter grids.

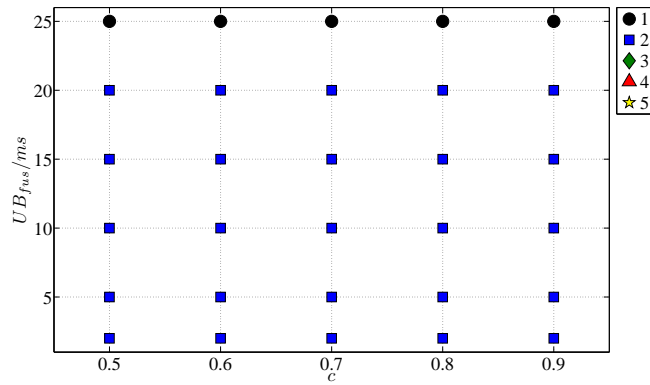
Every set of three subfigures represents one of $q = 0.01 \frac{m^2}{s^5}$, $q = 0.1 \frac{m^2}{s^5}$, $q = 1 \frac{m^2}{s^5}$, $q = 10 \frac{m^2}{s^5}$ and $q = 100 \frac{m^2}{s^5}$.

In regard to the best state-of-the-art configurations, Figures 4.6(a), 4.7(a), 4.8(a), 4.9(a), and 4.10(a) show that the state-of-the-art BUFF configuration outmatches the state-of-the-art ADVA configuration for slow object tracking subsystem, $UB_{fus} = 25 \text{ ms}$, in combination with small to medium process noise power spectral density, $q = \{0.01, 0.1, 1\} \frac{m^2}{s^5}$. However for the remaining parameter grid points, the state-of-the-art ADVA configuration yields better results than the state-of-the-art BUFF configuration.

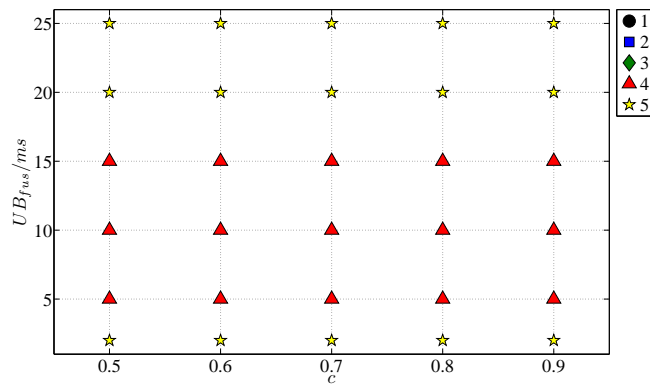
In regard to the best time-triggered configurations, Figures 4.6(b), 4.7(b), 4.8(b), 4.9(b), and 4.10(b) show that for medium upper bound for the fusion PT, $UB_{fus} = \{5, 10, 15\} \text{ ms}$ in combination with small process noise power spectral density, $q = \{0.01, 0.1\} \frac{m^2}{s^5}$, the time-triggered unsynchronized ADVA configuration outmatches the time-triggered synchronized configuration and the time-triggered unsynchronized BUFF configuration. For small and high upper bounds for the fusion PT, $UB_{fus} = \{2, 20, 25\} \text{ ms}$, however, the time-triggered synchronized configuration outmatches the time-triggered unsynchronized ADVA configuration and the time-triggered unsynchronized BUFF configuration.

With increasing process noise power spectral density, $q = \{1, 10\} \frac{m^2}{s^5}$, the time-triggered unsynchronized ADVA configuration outmatches the time-triggered synchronized configuration also for an upper bound for the fusion PT of $UB_{fus} =$

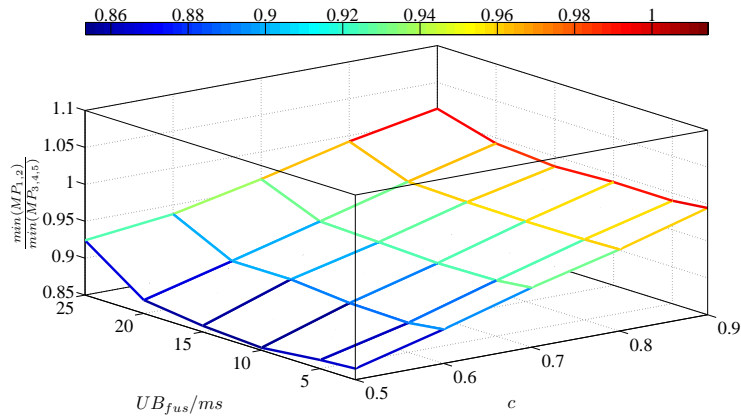
4. Numerical Simulation



(a) Best state-of-the-art configurations ($\text{argmin}(\text{MP}_{1,2})$)

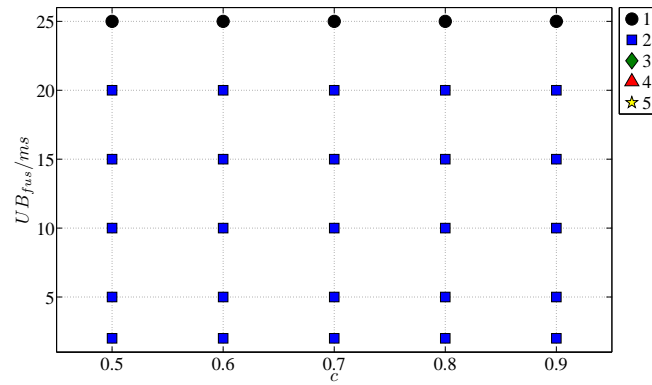


(b) Best time-triggered configurations ($\text{argmin}(\text{MP}_{3,4,5})$)

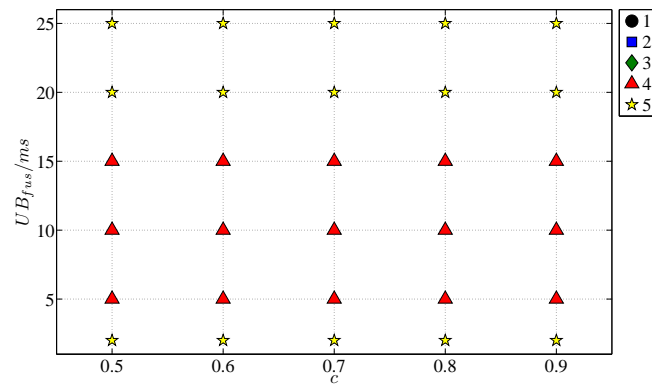


(c) Best state-of-the-art configurations MP/best time-triggered configurations MP ratios

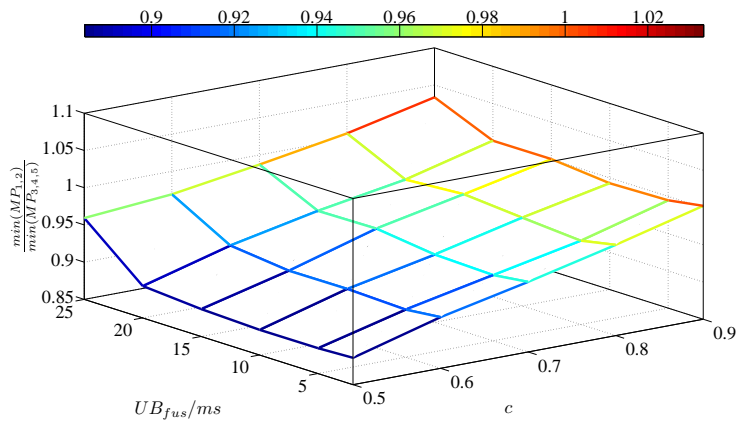
Figure 4.6.: Comparison of best state-of-the-art configurations and best time-triggered configurations for $q = 0.01 \frac{m^2}{s^5}$



(a) Best state-of-the-art configurations ($\text{argmin}(\text{MP}_{1,2})$)



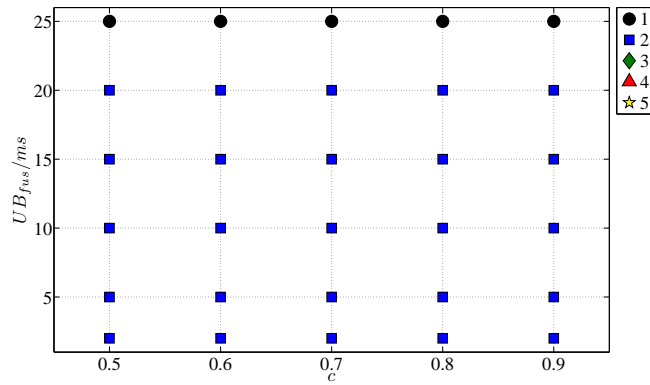
(b) Best time-triggered configurations ($\text{argmin}(\text{MP}_{3,4,5})$)



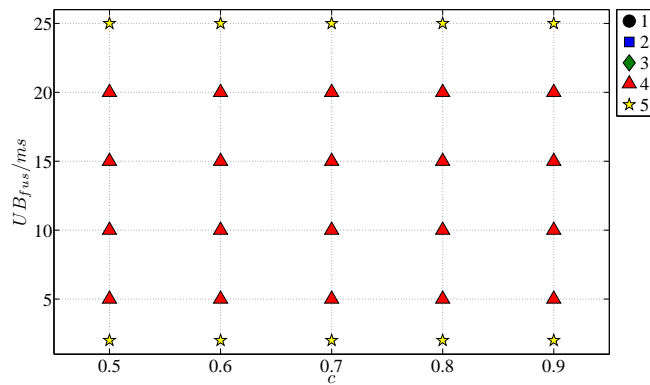
(c) Best state-of-the-art configurations MP/best time-triggered configurations MP ratios

Figure 4.7.: Comparison of best state-of-the-art configurations and best time-triggered configurations for $q = 0.1 \frac{m^2}{s^5}$

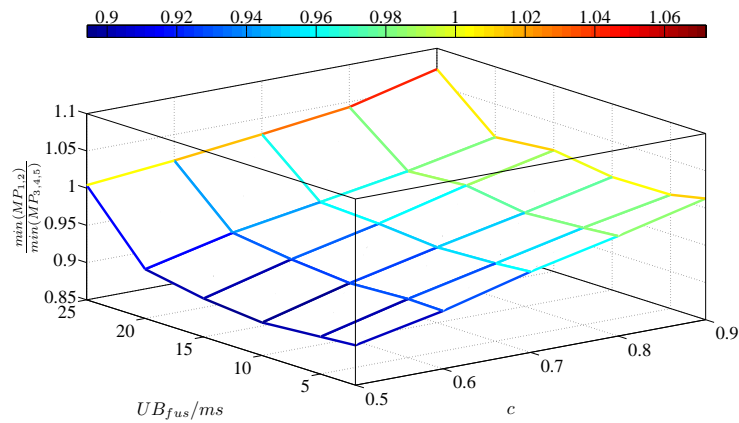
4. Numerical Simulation



(a) Best state-of-the-art configurations ($\text{argmin}(\text{MP}_{1,2})$)

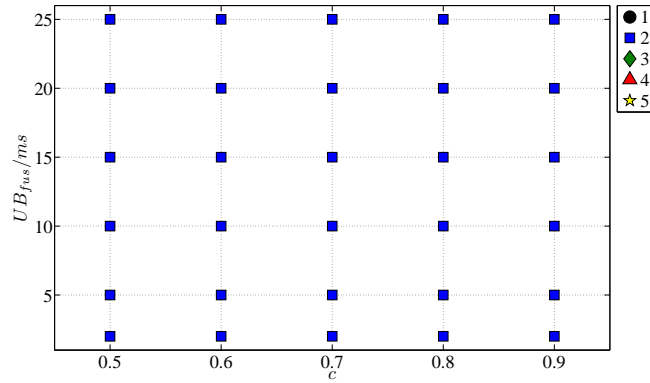


(b) Best time-triggered configurations ($\text{argmin}(\text{MP}_{3,4,5})$)

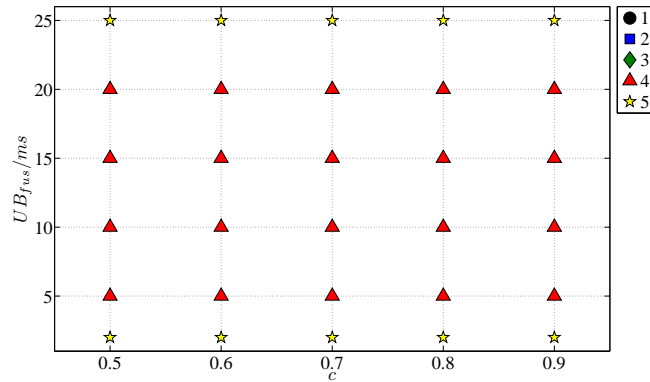


(c) Best state-of-the-art configurations MP/best time-triggered configurations MP ratios

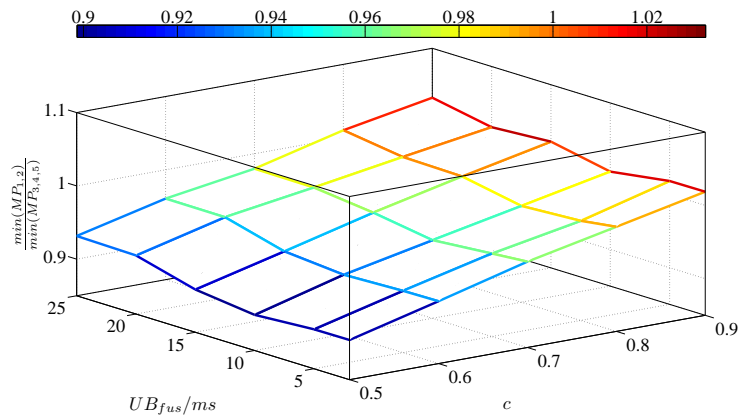
Figure 4.8.: Comparison of best state-of-the-art configurations and best time-triggered configurations for $q = 1 \frac{m^2}{s^5}$



(a) Best state-of-the-art configurations ($\text{argmin}(\text{MP}_{1,2})$)



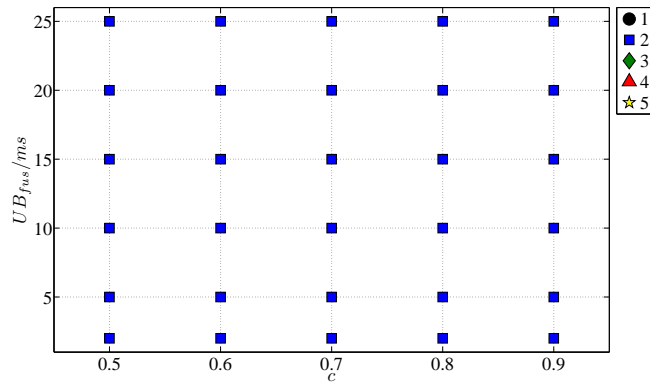
(b) Best time-triggered configurations ($\text{argmin}(\text{MP}_{3,4,5})$)



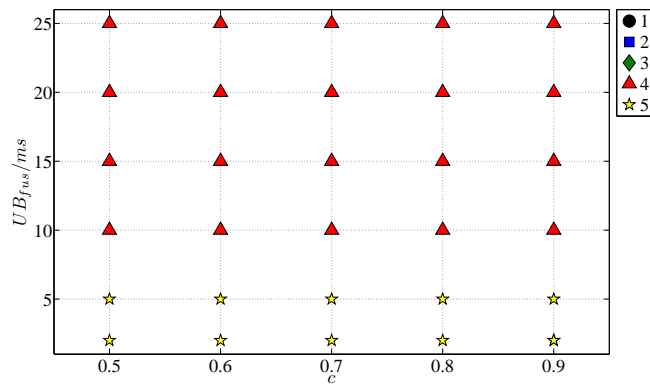
(c) Best state-of-the-art configurations MP/best time-triggered configurations MP ratios

Figure 4.9.: Comparison of best state-of-the-art configurations and best time-triggered configurations for $q = 10 \frac{m^2}{s^5}$

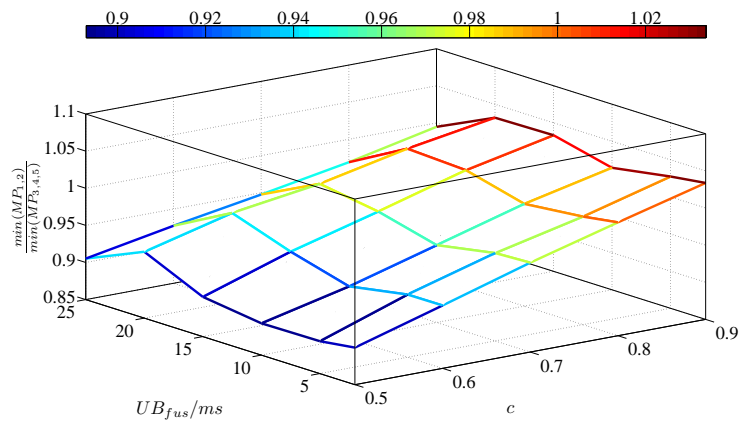
4. Numerical Simulation



(a) Best state-of-the-art configurations ($\text{argmin}(\text{MP}_{1,2})$)



(b) Best time-triggered configurations ($\text{argmin}(\text{MP}_{3,4,5})$)



(c) Best state-of-the-art configurations MP/best time-triggered configurations MP ratios

Figure 4.10.: Comparison of best state-of-the-art configurations and best time-triggered configurations for $q = 100 \frac{m^2}{s^5}$

$\{20\}$ ms , i.e., the best time-triggered configuration for $UB_{fus} = 20$ ms shifts from the time-triggered synchronized configuration to the time-triggered unsynchronized ADVA configuration.

For high process noise power spectral density, $q = 100 \frac{m^2}{s^5}$, the time-triggered unsynchronized ADVA configuration outmatches the time-triggered synchronized configuration for medium to high upper bounds for the fusion PT, $UB_{fus} = \{10, 15, 20, 25\}$ ms , and the time-triggered synchronized configuration outmatches the time-triggered unsynchronized ADVA configuration for a fast object tracking subsystem, $UB_{fus} = \{2, 5\}$ ms . This shows that if the process noise power spectral density increases from $q = 10 \frac{m^2}{s^5}$ to $q = 100 \frac{m^2}{s^5}$, the best time-triggered configuration shifts from the time-triggered unsynchronized ADVA configuration to the time-triggered synchronized configuration for an upper bound for the fusion PT of $UB_{fus} = 5$ ms . For an upper bound for the fusion PT of $UB_{fus} = 25$ ms however, the best time-triggered configuration shifts vice versa.

In regard to the MP of the best state-of-the-art configurations and the best time-triggered configurations, Figures 4.6(c), 4.7(c), 4.8(c), 4.9(c), and 4.10(c) show the MP ratios of the best state-of-the-art configurations to the best time-triggered configurations which have been identified in Figures 4.6(a), 4.7(a), 4.8(a), 4.9(a), 4.10(a), 4.6(b), 4.7(b), 4.8(b), 4.9(b), and 4.10(b). The figures show that the difference between the best state-of-the-art configurations and the best time-triggered configurations range from -15% to $+6\%$ of the mean RT ECM trace of the respective best time-triggered configuration.

For small process noise power spectral density, $q = 0.01 \frac{m^2}{s^5}$, the difference ranges from -15% to $+1\%$ and for high process noise power spectral density, $q = 100 \frac{m^2}{s^5}$, the difference ranges from -10% to $+2\%$. It should be noted however, that the biggest difference of $+6\%$ is found for medium process noise power spectral density, $q = 1 \frac{m^2}{s^5}$.

4.6. Analysis of Simulation Results

As the time-triggered configurations schedule all processes in accordance with their WCET, the MP measures of the time-triggered model configurations are unaffected by a decrease of the lower bounds for sensor and fusion PTs, indicated by a decrease of the complexity variance parameter. As a state-of-the-art configuration may start a new task as soon as the preceding task has been finished, the state-of-the-art configuration profit from shorter sensor and fusion PTs. This leads to the observed behavior where the state-of-the-art configurations outmatch the time-triggered configurations for a decreasing complexity variance parameter, as shown in Figures 4.6(c), 4.7(c), 4.8(c), 4.9(c), and 4.10(c).

However, there are also aspects for which there is no straightforward explanation:

1. For a medium to small complexity variance parameter, $c \leq 0.8$, a slow object tracking subsystem, $UB_{fus} = 25 \text{ ms}$, and a low process noise power spectral density, $q \leq 0.1 \frac{m^2}{s^5}$, the state-of-the-art **BUFF** configuration yields better results than all other configurations. The state-of-the-art **BUFF** configuration is outmatched by the time-triggered synchronized configuration for the process noise power spectral density increasing to a medium process noise power spectral density, $q = 1 \frac{m^2}{s^5}$. The time-triggered synchronized configuration however, is in turn outmatched by the state-of-the-art **ADVA** configuration if the process noise power spectral density increases to a high process noise power spectral density, $q \geq 10 \frac{m^2}{s^5}$ (see Figure 4.5). This behavior is unclear in that it is not obvious why the increasing process noise power spectral density promotes the time-triggered synchronized and the state-of-the-art **ADVA** configuration over the state-of-the-art **BUFF** configuration, which leads to the time-triggered synchronized configuration being optimal for medium process noise power spectral density and ends up by the state-of-the-art **ADVA** configuration outmatching the time-triggered synchronized configuration for high process noise power spectral density. The observed behavior is even more dubious when considering that the time-triggered synchronized configuration seems to be further “apart”, in a structural sense, from the state-of-the-art **BUFF** configuration than the state-of-the-art **ADVA** configuration;
2. For a fast or slow object tracking subsystem (small or high upper bound for the fusion **PT**, $UB_{fus} = \{2, 25\} \text{ ms}$) in combination with medium process noise power spectral density, $0.1 \leq q \leq 10 \frac{m^2}{s^5}$, the time-triggered synchronized configuration outmatches the time-triggered unsynchronized **ADVA** configuration. However, for medium upper bound for the fusion **PT**, $UB_{fus} = \{5, 10, 15, 20\} \text{ ms}$, the time-triggered configuration is outmatched by the time-triggered unsynchronized **ADVA** configuration (see Figures 4.6(b), 4.7(b), 4.8(b), and 4.9(b)). This behavior is unclear as if an increase of the upper bound for the fusion **PT** from a small upper bound for the fusion **PT** to a medium upper bound for the fusion **PT** promotes the time-triggered unsynchronized **ADVA** configuration, why would a further increase of of the upper bound for the fusion **PT** from a medium upper bound for the fusion **PT** to a high upper bound for the fusion **PT** then promote the time-triggered synchronized configuration; and
3. The speed of the object tracking subsystem⁹ and the process noise power spectral density seem to be interrelated with regard to their impact on the configuration **MP**. This can be recognized when comparing Figures 4.6(b), 4.7(b), 4.8(b), 4.9(b), and 4.10(b). Therein, the region where the time-triggered un-

⁹The speed of an object tracking subsystem is directly related to the upper bound for the fusion **PT**.

synchronized ADVA configuration yields better results than the time-triggered synchronized configuration shifts with increasing process noise power spectral density from $UB_{fus} = \{5, 10, 15\} ms$ to $UB_{fus} = \{10, 15, 20, 25\} ms$. Furthermore, an increasing process noise power spectral density seems to promote the time-triggered unsynchronized ADVA configuration and the time-triggered synchronized configuration as long as the upper bound for fusion PT is smaller than $UB_{fus} = 25 ms$. However, for a high upper bound for the fusion PT, $UB_{fus} = 25 ms$, the described behavior is inverted and an increase of the process noise power spectral density seems to promote the state-of-the-art ADVA configuration (see Figure 4.5).

These aspects motivate a further analysis of the impact of a variation of the upper bound for the fusion PT and the process noise power spectral density on the configuration MP measure.

4.6.1. Relationship Between Mean Performance and Parameters

According to section 4.4, the MP measure is the mean ECM trace of the RT image of an object state calculated from a sequence of RT images:

$$\text{MP} := \text{mean}_{n \in \mathbb{N}} \left(\text{trace} \left(P \left(t_{\text{RT}} | t_{\text{ST}} \right) \right) \right) \quad (4.20)$$

with $t_{\text{RT}} = \Delta t_{\text{PH},1,1}^{\text{pre}} + n \cdot \Delta t_{\text{CT}}^{\text{pre}}$ and $t_{\text{ST}} = t_{\text{ST}}(t_{\text{RT}})$.¹⁰

Inserting equation 2.6 into equation 4.20 leads to

$$\text{MP} = \text{mean}_{n \in \mathbb{N}} \left(\text{trace} \left(\left(e^{F \Delta t_{\text{RT-ST}}} P \left(t_{\text{ST}}(t_{\text{RT}}) \right) \left(e^{F \Delta t_{\text{RT-ST}}} \right)^T + Q \left(\Delta t_{\text{RT-ST}} \right) \right) \right) \right) \quad (4.21)$$

with $\Delta t_{\text{RT-ST}} = t_{\text{RT}} - t_{\text{ST}}$, which can be evaluated to¹¹

¹⁰Please note, that the sequence is determined by $t_{\text{RT}} = \Delta t_{\text{PH},1,1}^{\text{pre}} + n \cdot \Delta t_{\text{CT}}^{\text{pre}}$ and is therefore schedule dependent over $\Delta t_{\text{PH},1,1}^{\text{pre}}$.

¹¹Please note that F^3 is zero for the white noise jerk model.

$$\begin{aligned}
 \text{MP} = & \text{mean}_{n \in \mathbf{N}} \left(\text{trace} \left(P \left(t_{\text{ST}(\text{RT})} \right) \right) \right) + \\
 & + \text{mean}_{n \in \mathbf{N}} \left(\text{trace} \left((F \Delta t_{\text{RT}-\text{ST}}) P \left(t_{\text{ST}(\text{RT})} \right) \right) \right) + \\
 & + \text{mean}_{n \in \mathbf{N}} \left(\text{trace} \left(P \left(t_{\text{ST}(\text{RT})} \right) (F \Delta t_{\text{RT}-\text{ST}})^T \right) \right) + \\
 & + \text{mean}_{n \in \mathbf{N}} \left(\text{trace} \left((F \Delta t_{\text{RT}-\text{ST}}) P \left(t_{\text{ST}(\text{RT})} \right) (F \Delta t_{\text{RT}-\text{ST}})^T \right) \right) + \\
 & + \text{mean}_{n \in \mathbf{N}} \left(\text{trace} \left(\left(F^2 \frac{(\Delta t_{\text{RT}-\text{ST}})^2}{2} \right) P \left(t_{\text{ST}(\text{RT})} \right) \right) \right) + \\
 & + \text{mean}_{n \in \mathbf{N}} \left(\text{trace} \left(P \left(t_{\text{ST}(\text{RT})} \right) \left(F^2 \frac{(\Delta t_{\text{RT}-\text{ST}})^2}{2} \right)^T \right) \right) + \\
 & + \text{mean}_{n \in \mathbf{N}} \left(\text{trace} \left(\left(F^2 \frac{(\Delta t_{\text{RT}-\text{ST}})^2}{2} \right) P \left(t_{\text{ST}(\text{RT})} \right) (F \Delta t_{\text{RT}-\text{ST}})^T \right) \right) + \\
 & + \text{mean}_{n \in \mathbf{N}} \left(\text{trace} \left((F \Delta t_{\text{RT}-\text{ST}}) P \left(t_{\text{ST}(\text{RT})} \right) \left(F^2 \frac{(\Delta t_{\text{RT}-\text{ST}})^2}{2} \right)^T \right) \right) + \\
 & + \text{mean}_{n \in \mathbf{N}} \left(\text{trace} \left(\left(F^2 \frac{(\Delta t_{\text{RT}-\text{ST}})^2}{2} \right) P \left(t_{\text{ST}(\text{RT})} \right) \left(F^2 \frac{(\Delta t_{\text{RT}-\text{ST}})^2}{2} \right)^T \right) \right) + \\
 & + \text{mean}_{n \in \mathbf{N}} \left(\text{trace} \left(Q \left(\Delta t_{\text{RT}-\text{ST}} \right) \right) \right) \tag{4.22}
 \end{aligned}$$

Equation 4.22 shows that the configuration specific features that have an impact on the MP measure are the sequence of object state ST image ECM traces, $\text{trace} \left(P \left(t_{\text{ST}(\text{RT})} \right) \right)$, the sequence of object state ST image ECM traces in combination with the sequence of intervals between ST and RT, $\Delta t_{\text{RT}-\text{ST}}$, and the sequence of integrated process noise traces, $\text{trace} \left(Q \left(\Delta t_{\text{RT}-\text{ST}} \right) \right)$.

It is therefore necessary to analyze how a variation of the upper bound for the fusion PT and the process noise power spectral density influence the sequence of intervals between ST and RT, the sequence of object state ST image ECM traces, and the sequence of integrated process noise traces.

As the process noise power spectral density has no impact on the sequence of intervals between ST and RT, the following points have to be answered:

- How does a variation of the object tracking subsystem speed impact the sequence of intervals between ST and RT;
- How does a variation of the object tracking subsystem speed impact the sequence of object state ST image ECM traces;
- How does a variation of the process noise power spectral density impact the sequence of object state ST image ECM traces; and

- How does a variation of the object tracking subsystem speed and the process noise power spectral density impact the sequence of integrated process noise traces.

4.6.2. Upper Bound for Fusion Process Time

In order to analyze the impact of a variation of the upper bound for the fusion PT on the sequence of intervals between ST and RT and on the sequence of ST image object state ECM traces within the state-of-the-art configurations and the time-triggered configurations, the temporal evolution of the object state images has to be investigated.

Figures 4.11, 4.12, 4.13, 4.14, and 4.15 show the temporal evolution of the object state images for the state-of-the-art BUFF configuration, the state-of-the-art ADVA configuration, the time-triggered unsynchronized ADVA configuration, the time-triggered unsynchronized BUFF configuration, and the time-triggered synchronized configuration.

Each of the figures consists of a lower part, which depicts the processes on the sensors, bus system, and object tracking subsystem and an upper part, which depicts the ST of the object state images over RT. The lower parts are identical to Figures 4.2(a), 4.2(b), 4.3(a), 4.3(b), and 4.4 which allows to concentrate the discussion on the evolution of the object state ST images over RT.

In the upper parts of Figures 4.11, 4.12, 4.13, 4.14, and 4.15, snapshots taken by the sensors are indicated by circles. The circles are placed at time points where $t_{ST} = t_{RT}$, as the information which is contained in a snapshot refers to the time point where the snapshot is taken. After preprocessing the snapshot, the extracted object state observations are transmitted to the object tracking subsystem. The reception of object state observations is indicated by ovals that are marked with an “x” inside, if the respective object state observation set contains OOSMs. The ovals are horizontally aligned to the circles, as the information contained in the received object state observations refers to the time point at which the respective snapshot was taken. Furthermore, the ovals are displaced along the positive RT axis, the displacement indicating the time required for snapshot preprocessing and transmission.

Depending on whether it is a BUFF or an ADVA configuration, received object state observations are then either delayed until they can be fused in chronological order and processing resources are available (Figures 4.11 and 4.13), or fused as soon as processing resources are available (Figures 4.12, 4.14, and 4.15).

After an object state observation set has been fused, the fusion ST, indicated by horizontal bars, increases to the time point where the snapshot of the surrounding environment has been taken and remains constant until a more recent object state observation set is fused, resulting in a ST step function.

At predefined points in time, the object tracking subsystem starts to predict RT

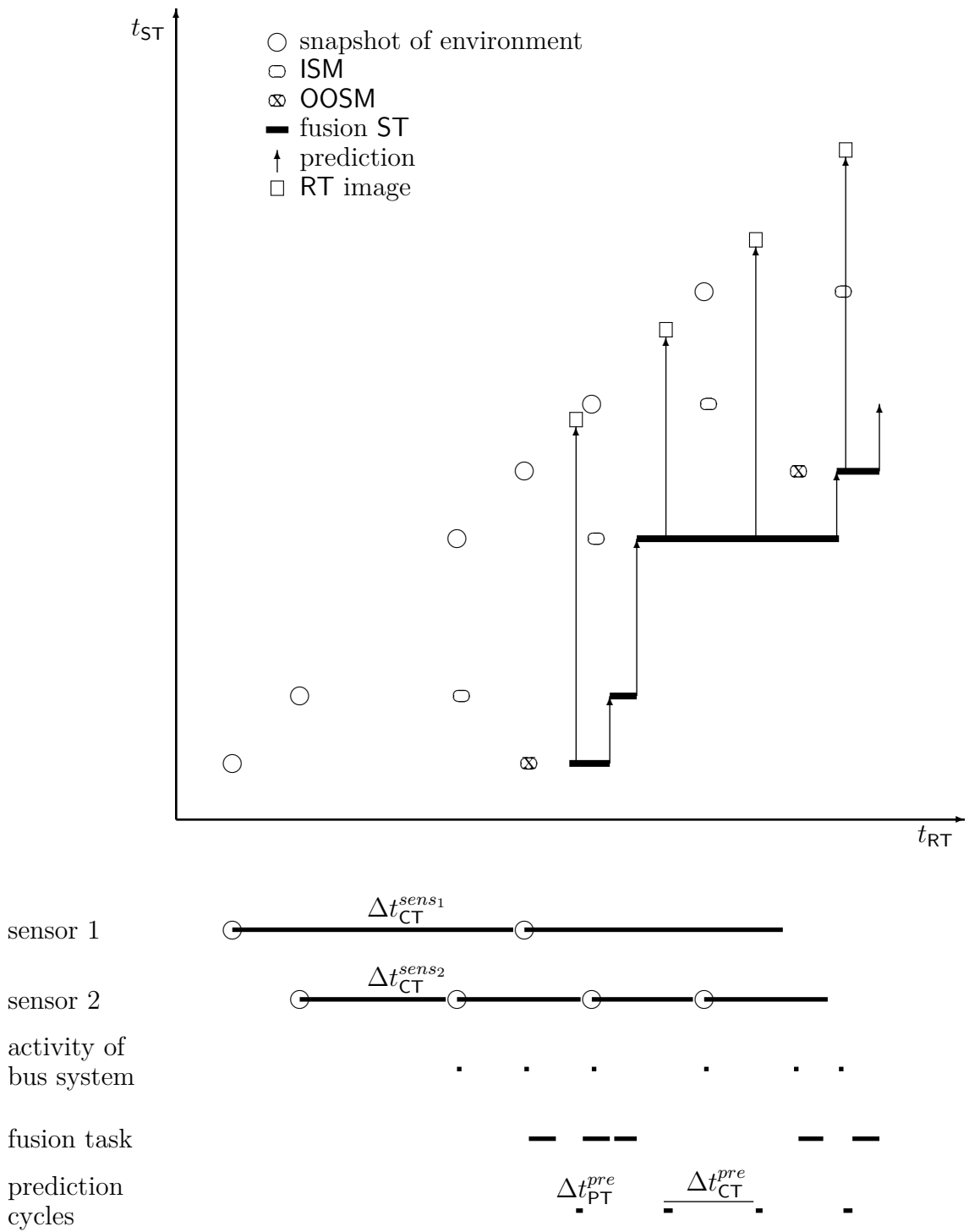


Figure 4.11.: Temporal evolution of an object state ST image in the state-of-the-art BUFF configuration

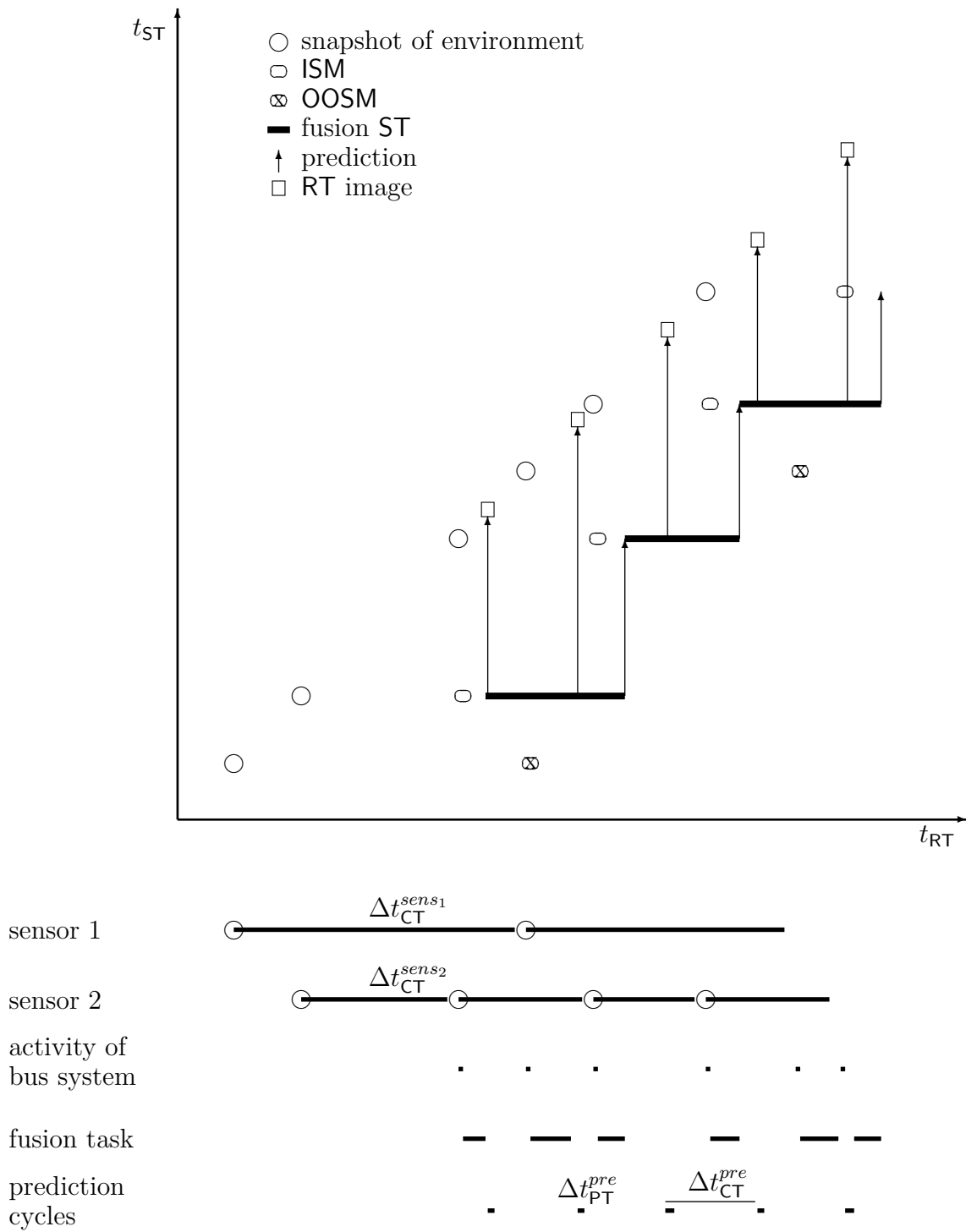


Figure 4.12.: Temporal evolution of an object state ST image in the state-of-the-art ADVA configuration

4. Numerical Simulation

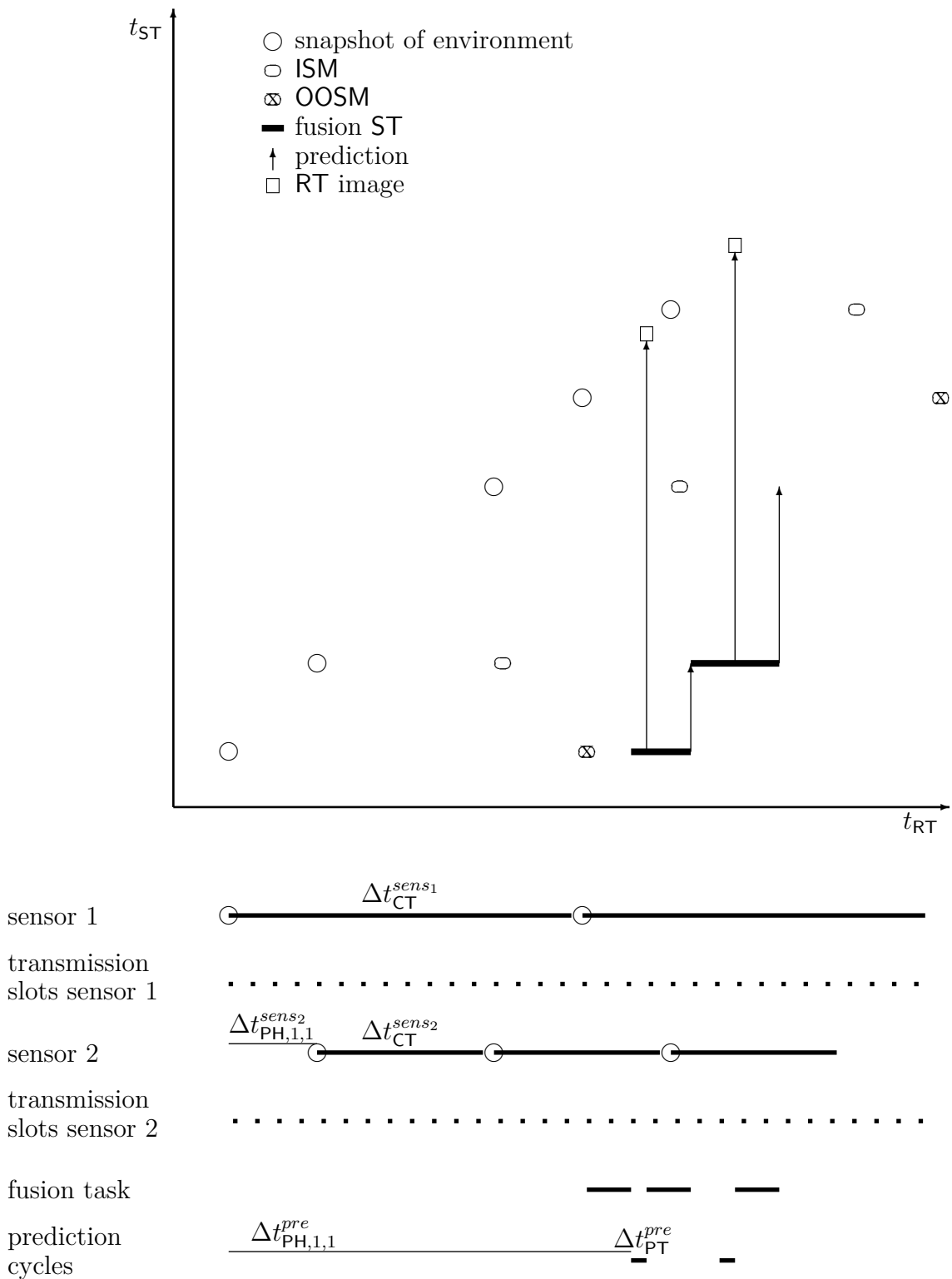


Figure 4.13.: Temporal evolution of an object state ST image in the time-triggered unsynchronized BUFF configuration

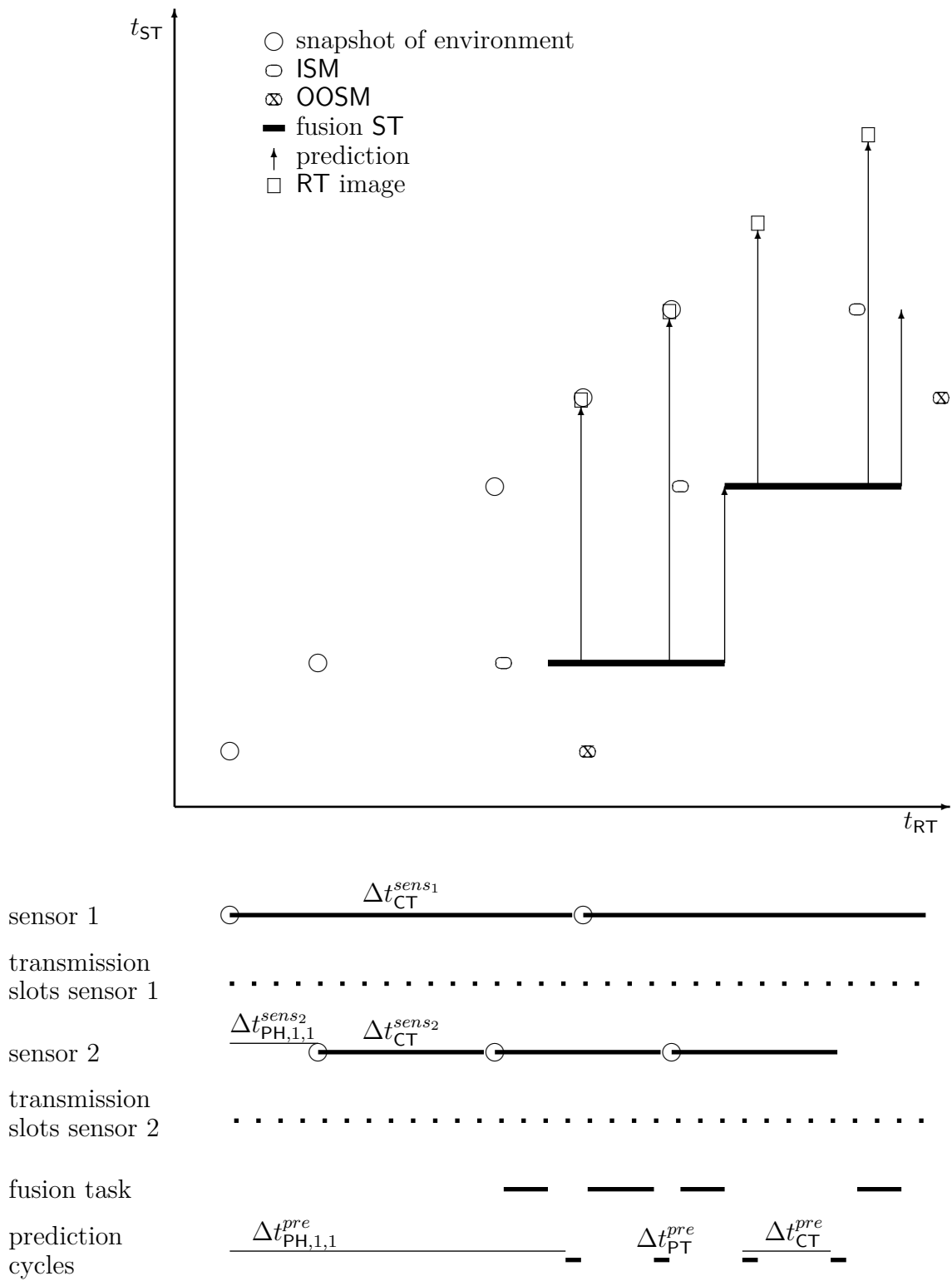


Figure 4.14.: Temporal evolution of an object state ST image in the time-triggered unsynchronized ADVA configuration

4. Numerical Simulation

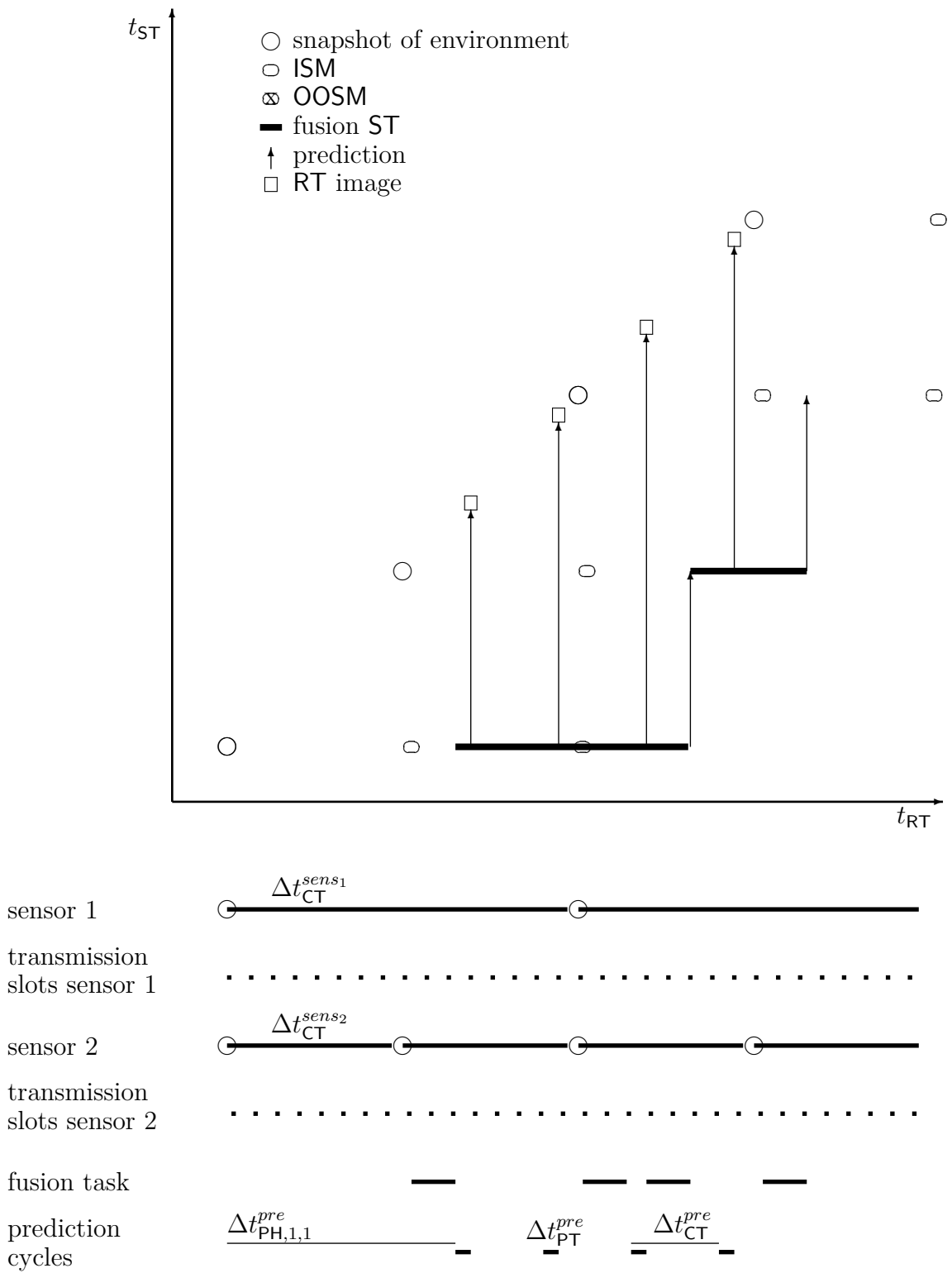


Figure 4.15.: Temporal evolution of an object state ST image in the time-triggered synchronized configuration

images of the object states which are indicated by squares. To predict an RT image of the surrounding environment for the feature service subsystem, the fused ST images of the object states must be predicted over the interval from ST to RT which is indicated by the vertical arrows ending at the squares.

In regard to the impact of a variation of the upper bound for fusion PT on the sequence of intervals between ST and RT and on the sequence of object state ST image ECM traces, it should be noted that the sequence of object state ST image ECM traces is affected by every fused object state observation, whereas the sequence of intervals between ST and RT is only affected by object state observations with a more recent time-stamp than the current ST.

This is apparent when examining the state-of-the-art ADVA configuration, the time-triggered unsynchronized ADVA configuration and the time-triggered synchronized configuration where the fusion of an object state observation set from sensor 1 does not affect the sequence of intervals between ST and RT.

Therefore, it seems reasonable to analyze the impact of the object tracking subsystem speed on the sequence of intervals between ST and RT and on the sequence of object state ST image ECM traces separately.

Impact of the Upper Bound for Fusion Processing Time on the Sequence of Intervals Between State-Time and Real-Time

The influence of a variation of the upper bound for fusion PT on the sequence of intervals between ST and RT can be classified into three categories:

1. Two or more object state observation sets can be fused between two prediction tasks;
2. One object state observation set can be fused between two prediction tasks; and
3. There are object state observation sets that cannot be fused between two prediction tasks and therefore have to be discarded.

Taking into account that the prediction CT is 40 *ms*, that the prediction requires $\frac{1}{3}\Delta t_{PT}^{fusISM}$, and that the fusion of an OOSM requires $\frac{3}{2}\Delta t_{PT}^{fusISM}$, the possible values for Δt_{PT}^{fusISM} can be classified as shown in table 4.1.

Table 4.1 shows that Δt_{PT}^{fusISM} in the state-of-the-art BUFF, the time-triggered BUFF, and the time-triggered synchronized configuration can be identically classified into the categories 1 and 2, with $\Delta t_{PT}^{fusISM} = \{2, 5, 10, 15\}$ *ms* belonging to category 1 and $\Delta t_{PT}^{fusISM} = \{20, 25\}$ *ms* belonging to category 2. Furthermore, table 4.1 shows that Δt_{PT}^{fusISM} in the state-of-the-art ADVA and the time-triggered unsynchronized ADVA configuration can be identically classified into the categories 1, 2, and 3, with $\Delta t_{PT}^{fusISM} = \{2, 5, 10\}$ *ms* belonging to category 1, $\Delta t_{PT}^{fusISM} = \{15, 20\}$ *ms* belonging

Configuration	Category	$\Delta t_{\text{PT}}^{\text{fusISM}}/ms$
State-of-the-art BUFF	1	{2, 5, 10, 15}
	2	{20, 25}
	3	
State-of-the-art ADVA	1	{2, 5, 10}
	2	{15, 20, 25(for ISM)}
	3	25(for OOSM)}
Time-triggered unsynchronized BUFF	1	{2, 5, 10, 15}
	2	{20, 25}
	3	
Time-triggered unsynchronized ADVA	1	{2, 5, 10}
	2	{15, 20, 25(for ISM)}
	3	25(for OOSM)}
Time-triggered synchronized	1	{2, 5, 10, 15}
	2	{20, 25}
	3	

Table 4.1.: ISM fusion PT, $\Delta t_{\text{PT}}^{\text{fusISM}}$, categories

to category 2, and $\Delta t_{\text{PT}}^{\text{fusISM}} = 25 ms$ belonging to category 2 if it represents an ISM and to category 3 if it represents an OOSM.

As the state-of-the-art BUFF and the state-of-the-art ADVA configuration have no deterministic schedule, the analysis starts with the time-triggered configurations and draws conclusions for the state-of-the-art configurations from the gained results as far as possible.

Figure 4.13 depicts the time-triggered unsynchronized BUFF configuration with $UB_{fus} = 20 ms$, which allows the fusion of one object state observation set between two consecutive prediction cycles. As a result, the prediction task is scheduled to align with the fusion task triggered by the incoming object state observation set of sensor 1. As the object state observation of sensor 1 is an OOSM, a previously received object state observation set from sensor 2 remains buffered until the OOSM from sensor 1 is fused. After the OOSM has been fused and the prediction task has been finished, the buffered object state observation from sensor 2 is fused. As the time required for fusion of the second object state observation set from sensor 2, which becomes available during the fusion of the object state observation from sensor 1, is too long for the fusion task to be executed before the next prediction task is scheduled, the fusion task is delayed until the second prediction task has been finished.

Due to the periodicity of the schedule, the sequence of intervals between ST and RT repeats a pattern of four prediction intervals $\Delta t_{\text{RT-ST}}$. If the upper bound for the fusion PT is longer than $UB_{fus} > 15 ms$, a variation of the upper bound for

the fusion PT will affect the prediction intervals in the pattern proportionally as the prediction task is directly aligned to the fusion task, which is scheduled in accordance with its upper bound.

If the upper bound for the fusion PT in Figure 4.13 increases, the prediction arrow is moved to the right and has to be prolonged in order to reach the square that travels along an axis defined by $t_{ST} = t_{RT}$.

If the upper bound for the fusion PT decreases from $UB_{fus} = 20 \text{ ms}$ to $UB_{fus} = 15 \text{ ms}$, the Δt_{RT-ST} pattern “jumps”, as the prediction task will be aligned to the second fusion task, which then directly follows the first fusion task. As a result, the first prediction arrow starts directly from the second ST level. Furthermore, the first interval between ST and RT of the Δt_{RT-ST} pattern experiences a jump decrease when the upper bound for the fusion PT changes from $UB_{fus} = 20 \text{ ms}$ to $UB_{fus} = 15 \text{ ms}$. However, it should be noted that the decrease of the first interval between ST and RT in the Δt_{RT-ST} pattern is partly compensated by the following intervals between ST and RT in the Δt_{RT-ST} pattern increasing by 10 ms due to the change in the PH of the prediction tasks.

When further decreasing the upper bound for the fusion PT, the sequence of intervals between ST and RT decreases twice as fast due to the execution of two fusion tasks before starting the prediction task.

In Figure 4.14, representing the time-triggered unsynchronized ADVA configuration, the prediction tasks are aligned to the fusion task triggered by the object state observation sets of sensor 1. As a result, a variation of the upper bound for the fusion PT leads to a 1.5 times greater variation of the sequence of intervals between ST and RT. Due to the schedule of the time-triggered unsynchronized ADVA configuration there is no possibility to fuse more than one object state observation set between two consecutive prediction tasks. If the upper bound for the fusion PT increases to $UB_{fus} = 25 \text{ ms}$, the object state observation sets of sensor 1 are discarded which, however, does not affect the sequence of intervals between ST and RT, as OOSMs have no effect on the ST of the object state image.

As a result, the time-triggered unsynchronized ADVA configuration reacts to a variation in the upper bound for the fusion PT from $UB_{fus} = 2 \text{ ms}$ to $UB_{fus} = 20 \text{ ms}$ with a proportional increase in the sequence of intervals between ST and RT.

Examining Figure 4.15, it becomes apparent that a variation of the upper bound for the fusion PT in the time-triggered synchronized configuration leads to a jump in the sequence of intervals between ST and RT, if the upper bound for the fusion PT changes from $UB_{fus} = 15 \text{ ms}$ to $UB_{fus} = 20 \text{ ms}$ and vice versa. Furthermore, a variation of the upper bound for the fusion PT leads to a twice as great variation in the sequence of intervals between ST and RT if the upper bound for the fusion PT ranges between $UB_{fus} = 2 \text{ ms}$ and $UB_{fus} = 15 \text{ ms}$ and to a proportional variation if the upper bound for the fusion PT ranges between $UB_{fus} = 20 \text{ ms}$ and $UB_{fus} = 25 \text{ ms}$.

With regard to the state-of-the-art BUFF and state-of-the-art ADVA configuration, it is clear that a variation of the upper bound for the fusion PT will lead to an over-proportional variation in the sequence of intervals between ST and RT. Furthermore, the sequences will not react with a jump as there is no scheduling concept that jumps nor is there a fixed fusion PT that might trigger a jump in a scheduling concept.

When summing up the aforesaid, the following conclusions can be drawn:

- The state-of-the-art BUFF configuration reacts to a variation of the upper bound for the fusion PT with an over-proportional variation in the sequence of intervals between ST and RT;
- The state-of-the-art ADVA configuration reacts to a variation of the upper bound for the fusion PT with an over-proportional variation in the sequence of intervals between ST and RT;
- The time-triggered unsynchronized BUFF configuration reacts to a variation of the upper bound for the fusion PT within $UB_{fus} = 2 \text{ ms}$ and $UB_{fus} = 15 \text{ ms}$ with a two times greater variation in the sequence of intervals between ST and RT. When the upper bound for the fusion PT becomes greater than $UB_{fus} = 15 \text{ ms}$, the sequence of intervals between ST and RT reacts with a jump increase. For an upper bound for the fusion PT being greater than $UB_{fus} > 15 \text{ ms}$, the sequence of intervals between ST and RT reacts with a proportional variation to a variation of the upper bound for the fusion PT;
- The time-triggered unsynchronized ADVA configuration reacts to an increasing upper bound for the fusion PT with a 1.5 times greater increase in the sequence of intervals between ST and RT; and
- The time-triggered synchronized configuration reacts to a variation of the upper bound for the fusion PT within $UB_{fus} = 2 \text{ ms}$ and $UB_{fus} = 15 \text{ ms}$ with a two times greater variation in the sequence of intervals between ST and RT. When the upper bound for the fusion PT becomes greater than $UB_{fus} = 15 \text{ ms}$, the sequence of intervals between ST and RT reacts with a jump increase. For an upper bound for the fusion PT being greater than $UB_{fus} > 15 \text{ ms}$, the sequence of intervals between ST and RT reacts with a proportional variation to a variation of the upper bound for the fusion PT.

Impact of the Upper Bound for the Fusion Processing Time on the Sequence of Object State State-Time Image Error Covariance Matrix Traces

It is clear that a variation of the upper bound for the fusion PT will not affect the ST image of object state ECMs in the value domain. However, it will affect the sequence of object state ST image ECM traces that are used to calculate the MP, as

the upper bound for the fusion PT has an effect on the schedule according to the configuration definition.

Further analysis of Figures 4.11, 4.12, 4.13, 4.14, and 4.15 shows that a sequence of object state ST image ECM traces will be unaffected by an increase or decrease of the upper bound for the fusion PT as long as the fusion of an object state observation is not shifted to before or after a prediction cycle or, an object state observation is discarded.

Therefore, the same classification into three categories can be used.

Having in mind the previous analysis, it is clear that the sequence of object state ST image ECM traces consists of a periodic pattern of four ST image object state ECMs.

A variation of the upper bound for the fusion PT in the time-triggered unsynchronized BUFF configuration will affect the first ST image object state ECM of the pattern if the upper bound for the fusion PT changes from $UB_{fus} = 15\text{ ms}$ to $UB_{fus} = 20\text{ ms}$ and vice versa. In contrast to the previous analysis where it was always favorable to fuse object state observations as this would lead to an decrease in the first prediction interval of the pattern, this conclusion cannot be drawn for the present situation, as the fusion of an object state observation consists of a prediction step, that increases uncertainty, and a fusion step, that decreases uncertainty. Therefore it is a question of prediction error and object state observation accuracy that determines whether the fusion of an object state observation is favorable.

In the time-triggered unsynchronized ADVA configuration, the pattern remains unaffected by a change in the upper bound for the fusion PT as long as the upper bound for the fusion PT is not greater than $UB_{fus} = 20\text{ ms}$. However if the upper bound for the fusion PT increases over $UB_{fus} = 20\text{ ms}$, object state observations of sensor 1 are discarded which results in a jump rise in the sequence of object state ST image ECM traces.

The time-triggered synchronized configuration can be seen analog to the time-triggered unsynchronized BUFF configuration where the first ST image object state ECM of the pattern changes if the upper bound for the fusion PT changes from $UB_{fus} = 15\text{ ms}$ to $UB_{fus} = 20\text{ ms}$ and vice versa. However, it is clear that the first ST image object state ECM of the pattern increases as the foregoing ST image object state ECM was updated with two object state observation sets with the same time-stamp.

How a variation of the upper bound for the fusion PT impacts the sequence of object state ST image ECM traces in the state-of-the-art BUFF or ADVA configuration cannot be determined analytically.

When summing up the aforesaid, the following conclusions can be drawn:

- The time-triggered unsynchronized BUFF configuration reacts to the upper bound for the fusion PT changing from $UB_{fus} = 15\text{ ms}$ to $UB_{fus} = 20\text{ ms}$ and vice versa with a change in the sequence of object state ST image ECM

traces, and stays stable for other variations of the upper bound for the fusion PT;

- The time-triggered unsynchronized ADVA configuration reacts to the upper bound for the fusion PT changing from $UB_{fus} = 20 \text{ ms}$ to $UB_{fus} = 25 \text{ ms}$ with a jump rise in the sequence of object state ST image ECM traces, and stays stable for other variations for the upper bound for the fusion PT; and
- The time-triggered synchronized configuration reacts to the upper bound for the fusion PT changing from $UB_{fus} = 15 \text{ ms}$ to $UB_{fus} = 20 \text{ ms}$ and vice versa with a change in the sequence of object state ST image ECM traces, and stays stable for other variations of the upper bound for the fusion PT. The change in the sequence is unfavorable, as before the change, the ST image of object states is updated by two object state observation sets with the same time stamp.

4.6.3. Process Noise Power Spectral Density

The process noise power spectral density impacts the sequence of object state ST image ECM traces through the matrix Q as shown in equation 2.6.

Accordingly, an increasing process noise power spectral density leads to an increase in the sequence of object state ST image ECM traces. This leads then to the question of how strong the sequence of object state ST image ECM traces reacts to a variation of the process noise power spectral density.

As there exists no analytical solution for the corresponding Riccati equation if the integrated process noise and the object state observation ECM are not constant, the impact cannot be determined analytically. However, when considering equations 2.6, 2.16, and 2.17, it is clear that an increase in the sequence of object state ST image ECM traces will lead to a gain matrix, K , which favors the object state observations and therefore, restricts the influence of the integrated process noise on the sequence of object state ST image ECM traces. Accordingly, an increase in the process noise power spectral density by a factor a increases, for example, the trace of the integrated process noise by the same factor a but does increase the sequence of object state ST image ECM traces by a factor that is smaller than a .

From the aforesaid, the following conclusion can be drawn:

- An increase in the process noise power spectral density leads to a under-proportional increase in the sequence of object state ST image ECM traces.

4.6.4. Interrelations

The sequence of integrated process noise traces is affected by the upper bound for the fusion PT and the process noise power spectral density, as the upper bound for

the fusion PT impacts the sequence of intervals between ST and RT and the process noise power spectral density acts as a “scaling factor” for the time matrices resulting from the sequence of intervals between ST to RT. As a result, the integrated process noise trace can be rewritten as

$$\text{trace}(Q(\Delta t_{\text{RT-ST}})) = q \cdot \text{trace}(Q'(\Delta t_{\text{RT-ST}})) \quad (4.23)$$

with

$$Q'(\Delta t_{\text{RT-ST}}) = \begin{pmatrix} \frac{(\Delta t_{\text{RT-ST}})^5}{20} & 0 & \frac{(\Delta t_{\text{RT-ST}})^4}{8} & 0 & \frac{(\Delta t_{\text{RT-ST}})^3}{6} & 0 \\ 0 & \frac{(\Delta t_{\text{RT-ST}})^5}{20} & 0 & \frac{(\Delta t_{\text{RT-ST}})^4}{8} & 0 & \frac{(\Delta t_{\text{RT-ST}})^3}{6} \\ \frac{(\Delta t_{\text{RT-ST}})^4}{8} & 0 & \frac{(\Delta t_{\text{RT-ST}})^3}{3} & 0 & \frac{(\Delta t_{\text{RT-ST}})^2}{2} & 0 \\ 0 & \frac{(\Delta t_{\text{RT-ST}})^4}{8} & 0 & \frac{(\Delta t_{\text{RT-ST}})^3}{3} & 0 & \frac{(\Delta t_{\text{RT-ST}})^2}{2} \\ \frac{(\Delta t_{\text{RT-ST}})^3}{6} & 0 & \frac{(\Delta t_{\text{RT-ST}})^2}{2} & 0 & \Delta t_{\text{RT-ST}} & 0 \\ 0 & \frac{(\Delta t_{\text{RT-ST}})^3}{6} & 0 & \frac{(\Delta t_{\text{RT-ST}})^2}{2} & 0 & \Delta t_{\text{RT-ST}} \end{pmatrix} \quad (4.24)$$

which can be simplified to

$$\text{trace}(Q(\Delta t_{\text{RT-ST}})) = q \cdot \left(2(\Delta t_{\text{RT-ST}}) + \frac{2(\Delta t_{\text{RT-ST}})^3}{3} + \frac{(\Delta t_{\text{RT-ST}})^5}{10} \right) \quad (4.25)$$

The impact of a variation of the upper bound for the fusion PT and the process noise power spectral density result from combining the analysis of subsection 4.6.2 with equation 4.25.

Accordingly, the following conclusions can be drawn:

- The state-of-the-art BUFF configuration reacts to a variation of the upper bound for the fusion PT with a more than two times greater variation in the sequence of integrated process noise traces;
- The state-of-the-art ADVA configuration reacts to a variation of the upper bound for the fusion PT with a more than two times greater variation in the sequence of integrated process noise traces;
- The time-triggered unsynchronized BUFF configuration reacts to a variation of the upper bound for the fusion PT within $UB_{fus} = 2 \text{ ms}$ and $UB_{fus} = 15 \text{ ms}$ with a four times greater variation in the sequence of integrated process noise traces. When the upper bound for the fusion PT becomes greater than $UB_{fus} = 15 \text{ ms}$ the sequence reacts with a jump increase. For an upper bound for the fusion PT being greater than $UB_{fus} > 15 \text{ ms}$ the sequence reacts with a two times greater variation to a variation of the upper bound for the fusion PT;

- The time-triggered unsynchronized ADVA configuration reacts to an increasing upper bound for the fusion PT with a three times greater increase in the sequence of integrated process noise traces;
- The time-triggered synchronized configuration reacts to a variation of the upper bound for fusion PT within $UB_{fus} = 2 \text{ ms}$ and $UB_{fus} = 15 \text{ ms}$ with a four times greater variation in the sequence of integrated process noise traces. When the upper bound for the fusion PT becomes greater than $UB_{fus} = 15 \text{ ms}$ the sequence reacts with a jump increase. For an upper bound for the fusion PT being greater than $UB_{fus} > 15 \text{ ms}$ the sequence reacts with a two times greater variation to a variation of UB_{fus} ; and
- The sequence of integrated process noise traces reacts to a variation of the process noise power spectral density with a proportional variation.

4.7. Discussion

Throughout this chapter, three aspects were described that could not be explained in a straightforward manner. However, through the preceding analysis, the following explanations can be provided:

1. The time-triggered synchronized configuration is able to fuse all object state observations but has greater values in the sequence of intervals between ST and RT compared to the state-of-the-art ADVA configuration. Accordingly, an increase in the process noise power spectral density which increases the sequence of integrated process noise traces to a greater extent than the sequence of object state ST image ECM traces is unfavorable for the time-triggered synchronized configuration, as the time-triggered synchronized configuration has the greater values in the sequence of intervals between ST and RT and therefore, the sequence of greater integrated process noise traces. The reason why the state-of-the-art ADVA configuration is outmatched by the time-triggered synchronized configuration for medium process noise power spectral density lies in the fact that the state-of-the-art ADVA configuration cannot fuse all object state observations of sensor 1. When the process noise power spectral density decreases, the influence of the integrated process noise traces is diminished and the focus shifts toward the sequence of object state ST image ECM traces. Here, the state-of-the-art BUFF configuration outmatches the time-triggered synchronized configuration due to the higher number of object state observation sets that are fused. The reason why this behavior is also observed for small lower bounds for sensor and fusion PTs is obvious when considering that the long times required to fuse an object state observation set and the

high number of uncoordinated object state observation sets may lead to fusion “jams”;

2. The time-triggered unsynchronized ADVA configuration has a sequence of object state ST image ECM traces that is unaffected by a variation in the upper bound for the fusion PT but reacts with a 1.5 times greater variation in the sequence of intervals between ST and RT. The time-triggered synchronized configuration experiences a jump in the sequence of object state ST image ECM traces for the upper bound for the fusion PT changing from $UB_{fus} = 15\text{ ms}$ to $UB_{fus} = 20\text{ ms}$. Furthermore, the sequence of intervals from ST to RT varies proportionally for $UB_{fus} \geq 20\text{ ms}$ and twice as great for $UB_{fus} \leq 15\text{ ms}$. Accordingly, the sequence of intervals between ST and RT increases stronger in the time-triggered synchronized configuration for the upper bound for the fusion PT $UB_{fus} \leq 15\text{ ms}$ and weaker for $UB_{fus} \geq 20\text{ ms}$ than the time-triggered unsynchronized configuration, which leads to the observed behavior; and
3. The observed interrelation stems from the influence of the sequence of integrated process noise traces that increase with increasing process noise power spectral density. In this regard, the jump in sequence of object state ST image ECM traces which react unfavorably to the upper bound for the fusion PT changing from $UB_{fus} = 15\text{ ms}$ to $UB_{fus} = 20\text{ ms}$ becomes greater and makes it impossible for the time-triggered synchronized configuration to be competitive.

4.8. Chapter Summary

In this chapter, a state-of-the-art model and a time-triggered model for multi-sensor ADASOTs are developed which consist of two sensors, a bus-system, an object tracking subsystem and a feature services subsystem.

In the state-of-the-art model, the sensor PHs are not controllable and the sensor CTs are equal to the sensor PTs which vary within a given range according to a Markov chain with given transition probability matrix. The object position observations are transmitted over an event-triggered bus system to the object tracking subsystem. On the object tracking subsystem, the object position observations are fused by a KF and predicted to RT. At predetermined points in time, the object tracking subsystem delivers the RT images of the object states to the feature service subsystem. The state-of-the-art model can be operated in two configurations, a state-of-the-art BUFF configuration, where object state observations are buffered and chronologically sorted before fusion and a state-of-the-art ADVA configuration that directly fuses OOSM using an ADVA approach.

In the time-triggered model, the sensor PHs are controllable, the sensor CTs are fixed and equal to the sensor worst case PTs. Furthermore, a time-triggered bus system with fixed transmission slots is used that transmits the object position observations from the sensors to the object tracking subsystem. After fusing the object position observations with a KF, the images of the object states are predicted to RT and delivered as RT images of the object states at predefined points in time to the feature service subsystem. The time-triggered model can be operated in various configurations from which three phase-aligned configurations are selected for further analysis, a time-triggered unsynchronized BUFF configuration, a time-triggered unsynchronized ADVA configuration, and a time-triggered synchronized configuration, where the object state observation sampling of both sensors is either unsynchronized or synchronized.

The results gained from simulating the tracking of an object by the state-of-the-art and the time-triggered configurations show that for the chosen parameter space, the state-of-the-art ADVA configuration yields the best results. However, the results also show that there are points in parameter space where the state-of-the-art ADVA configuration is outmatched by the state-of-the-art BUFF configuration, the time-triggered unsynchronized ADVA configuration or the time-triggered synchronized configuration.

When comparing the time-triggered unsynchronized ADVA configuration and the time-triggered synchronized configuration, the simulation results show that the time-triggered synchronized configuration outmatches the time-triggered unsynchronized ADVA configuration for slow and fast object tracking subsystems, but is outmatched by the time-triggered unsynchronized ADVA configuration for medium fast object tracking subsystems.

When comparing the best state-of-the-art configurations and the best time-triggered configurations, the simulation results show that the performance difference ranges from -15% to $+6\%$ of the mean RT ECM trace of the respective best time-triggered model configuration.

Furthermore, this chapter discusses aspects of the observed configuration behavior which cannot be explained in a straightforward manner. Heretofore the sequence of RT image object state ECM traces is decomposed into summands that relate to the sequence of prediction intervals and the sequence of object state ST image ECM traces. The sequence of prediction intervals and the sequence of object state ST image ECM traces is then analyzed for the different configurations. The gained results are used to explain the observed configuration behavior.

5. Case Study

The case study is being carried out on a Volkswagen Touran as depicted in Figure 5.1.

The Touran is equipped with a laser scanner, a stereo camera system, a PC104 hosting the object tracking subsystem and a feature service subsystem on a micro autobox as (schematically) shown in Figure 5.2 and Figure 5.3.

The laser scanner transmits its scans over a private CAN to an industrial PC (IPC) where the scans are preprocessed. From the IPC, the extracted object state observations are sent over a CAN/FlexRay gateway and the FlexRay bus to the PC104. The stereo camera system sends its frames over a private CAN to a PC based preprocessing unit. The preprocessing unit sends the extracted object state observations over a CAN/FlexRay gateway and the FlexRay bus to the PC104. After updating object states with associated object state observations, the RT images of the object states are delivered from the PC104 via FlexRay to the micro autobox, hosting the feature services.

The case study supports three configurations:

- The time-triggered unsynchronized BUFF configuration;
- The time-triggered unsynchronized ADVA configuration; and
- The time-triggered synchronized configuration.

5.1. Sensors

The stereo camera system is a “scabor” stereo vision system developed by the technical University of Cluj-Napoca and the laser scanner is an ALASCA laser scanner from Ibeo Automobile Sensor GmbH. Both sensors provide the possibility to be operated in a time-triggered mode [106] and have been developed for the tracking of objects in a road setting.

5.1.1. Scabor

The scabor stereo vision system consists of two cameras and an off-the-shelf PC which hosts the object state extraction algorithm. The PC is synchronized to the FlexRay time by a digital signal from the CAN/FlexRay gateway over RS232 and a corresponding CAN message containing the FlexRay time. The digital signal is

5. Case Study



Figure 5.1.: Volkswagen Touran test vehicle



Figure 5.2.: Test equipment

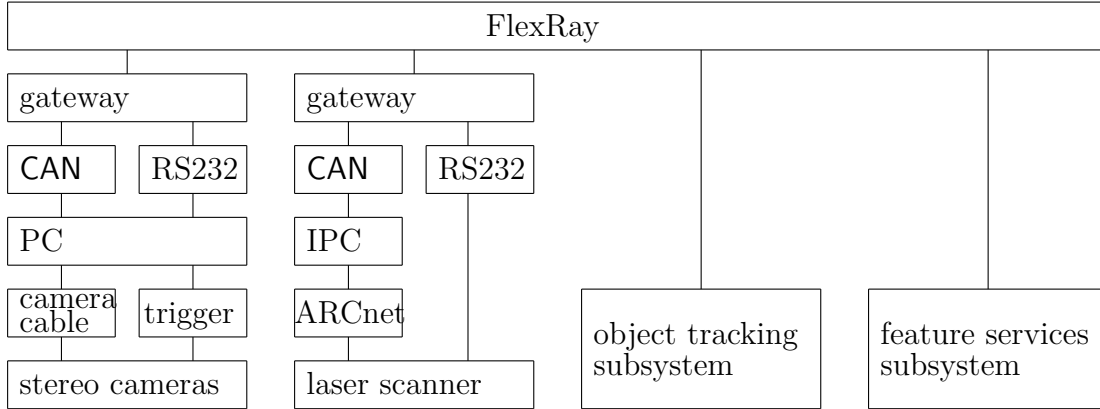


Figure 5.3.: Case study set-up

received by a frame-grabber that triggers the cameras. The cameras transmit the frames via camera cable to the PC where the object state observations are extracted and time-stamped with the time-stamp of the corresponding CAN message.

In the case study, the scabor stereo vision system delivers object coordinates (x, y) and object dimensions (w, l, h) .

$$\vec{z}_{t_k} = \begin{pmatrix} x \\ y \\ w \\ l \\ h \end{pmatrix} \quad (5.1)$$

The scabor stereo vision system is also able to deliver speed information, if an object has been tracked internally for some time. However, this information is not used in the case study in order to minimize the temporal correlation of errors. The object state observation ECM is computed online in a non-static manner from camera parameters and the location of the detected object.

The gateway triggers the frame grabber with 6.25 Hz .

5.1.2. Laser Scanner

The ALASCA laser scanner is based on LIDAR technology (LIght Detection And Ranging). It scans the surrounding environment by means of an infrared laser beam that is deflected by a rotating mirror. The laser emits short rapid-fire pulses that are reflected by objects. These reflections are then detected by the laser scanner which allows the pulses' response times to be measured. The distances to the objects can then be determined from the response time and the known velocity of light. Furthermore, the direction from which the reflected laser beam has been detected

is derived from the angular position of the rotating mirror that deflects the laser beam. The reflection points are then clustered to objects.

The laser scanner system consists of an ALASCA A0 scanner that emits four laser beams at different layers, an IPC which hosts the object extraction algorithm and a CAN/FlexRay gateway. The gateway sends a digital signal over RS232 to the IPC which in turn sends a digital signal to the scanner and synchronizes its internal clock to a time message that arrives over a private CAN. The rotating mirror in the laser scanner is triggered such that the laser beams point in the forward direction when the digital signal from the PC is received. The laser scanner transmits each scan via Arcnet to the IPC. In the IPC, the object state observations are extracted and time-stamped with the time stamp of the time message that was received from the CAN/FlexRay gateway.

The laser scanner system is able to determine the coordinates (x, y) of an object, its dimensions (w, l) depending on the shape of the object, and its speed (v_x, v_y) .

$$\vec{z}_{t_k} = \begin{pmatrix} x \\ y \\ w \\ l \\ v_x \\ v_y \end{pmatrix} \quad (5.2)$$

The laser scanner is triggered with 12.5 Hz .

5.2. Bus System

We use a FlexRay Star-Coupler to connect the sensors to the object tracking subsystem. The FlexRay round is repeated every 10 milliseconds.

The bus system also comprises two Decomsys NODE<ARM> CAN/FlexRay gateways that receive the CAN messages from the sensors. Thereon, the CAN messages are converted to FlexRay format and then transmitted over FlexRay to the object tracking subsystem.

5.3. Object Tracking Subsystem

The fusion is processed on a PC104 system running Linux/RTAI. The PC104 is connected to the FlexRay bus system by a TZM FlexCard. The received object state observations are processed by a KF as described in [51] and an ADVA as described in [9]. UB_{fus} has been determined to be 10 ms for the given hardware resources.¹ An update for the feature service subsystem is generated every 40 ms .

¹For a maximum number of 25 observed or tracked objects.

The object state is represented by vector 5.3 that consists of numerical values for the estimated Cartesian coordinates (x, y) , the direction in which the object moves (ψ), the dimensions (w, l, h) , the speed (v_x, v_y) , and the acceleration (a_x, a_y) .

$$\vec{x} = \begin{pmatrix} x \\ y \\ \psi \\ w \\ l \\ h \\ v_x \\ v_y \\ a_x \\ a_y \end{pmatrix} \quad (5.3)$$

The dynamical model is linear and restricted to the exact representation of objects that move with constant acceleration as represented by matrix 5.4 which is used to predict the evolution of the object state vector 5.3 from time instant t_k to time instant t_{k+1} with $\Delta t = t_{k+1} - t_k$.

$$F = \begin{pmatrix} 1 & 0 & \Delta t & 0 & 0 & 0 & 0 & 0 & \frac{\Delta t^2}{2} & 0 \\ 0 & 1 & 0 & \Delta t & 0 & 0 & 0 & 0 & 0 & \frac{\Delta t^2}{2} \\ 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & \Delta t & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & \Delta t \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 \end{pmatrix} \quad (5.4)$$

5.4. Scenarios

The first scenario is depicted in Figure 5.4. Therein, the vehicle bearing the sensors is stationary and one object accelerates in direction of the longitudinal axis of the vehicle.

In the second scenario, as depicted in Figure 5.5, the vehicle bearing the sensors is stationary and one object drives at a constant speed in the direction of the stationary vehicle and turns at about 10 m to 20 m ahead of it.

In both scenarios the vehicle bearing the sensors does not move in order to avoid a cumbersome correction of the ego-motion.

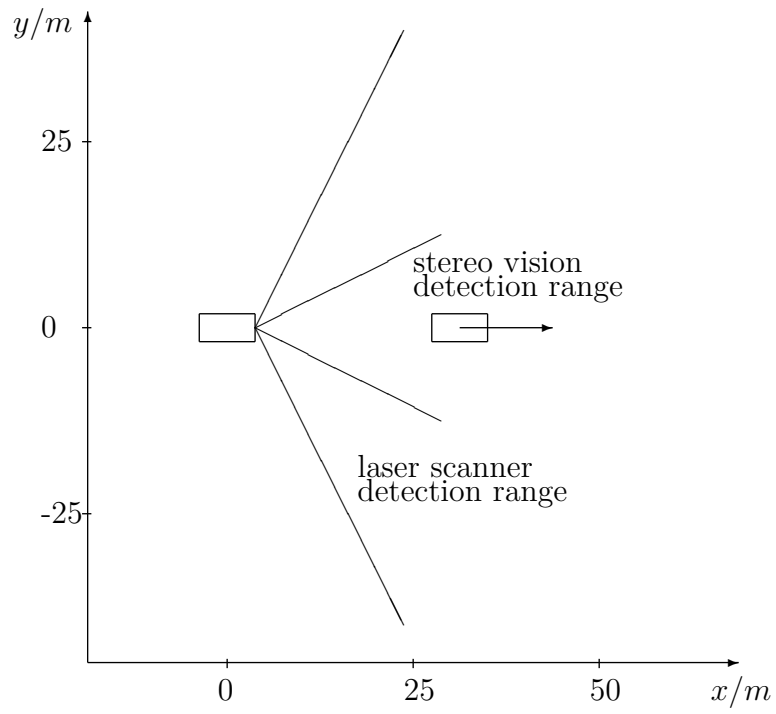


Figure 5.4.: Scenario 1

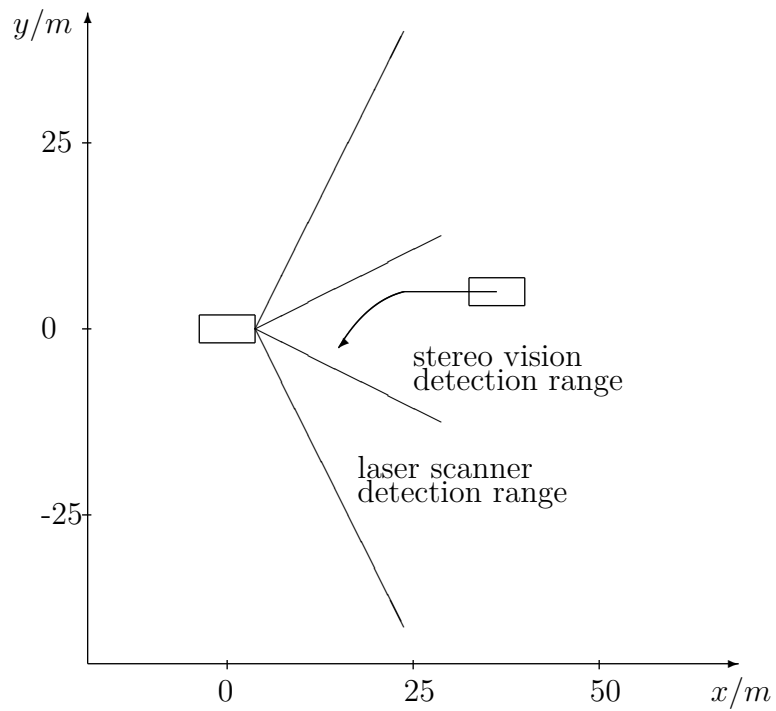


Figure 5.5.: Scenario 2

5.5. Differential Global Positioning System

In order to validate the trajectory of the object state vector representing the tracked object, said trajectory is directly measured by a DGPS system mounted on the tracked vehicle. The DGPS system is also deployed in order to measure the position of the vehicle bearing the sensors. For determining the direction in which the vehicle bearing the sensors was aligned, two DGPS measurements were taken at two spots which were about 50 *m* apart and where the vehicle was moved from one spot to the other with minimal steering movement.

5.6. Experimental Results

In this section, the results of four test drives are shown, comprising two test drives of the first scenario (scenario 1) and two test drives of the second scenario (scenario 2). The first test drive in each scenario represents the time-triggered unsynchronized BUFF and ADVA configuration, as both configurations use the same schedule. The second test drive represents the time-triggered synchronized configuration.

The results of the test drives are depicted in Figures 5.6, 5.7, 5.8, 5.9, 5.10, 5.11, 5.12, 5.13, 5.14, 5.14, 5.15, 5.16, and 5.17. Each figure consists of two subfigures, an RT *x* and an RT *y* plot. In each plot, object state observations from the DGPS, “dgps”, are indicated by circles, object state observations from the laser scanner, “lsc”, are indicated by crosses, object state observations from the stereo vision system, “scabor”, are indicated by triangles, and RT images of the object state, “fusion”, are indicated by squares.

Furthermore, the subfigures comprise arrows which point from object state observations to RT images of the object state. Every arrow indicates the points in time where the object state observations are contained in the RT image of the object states. Fusion points that will be discussed are referenced by boxed numbers.

Figure 5.6 and Figure 5.7 depict the time-triggered unsynchronized BUFF configuration in a scenario 1 test drive.

The test drive continues over 1500 *ms* within which the tracked vehicle moves from 30 *m* to 40 *m* in the positive *x* direction, which leads to an average speed of $v_x = 6.7 \frac{m}{s}$. Therein the graphs in the RT *x* plot cross each other at various points in time, and the graphs in the RT *y* plot vary around means that are vertically displaced by about 5 *cm* to 1 *m*. Between 0 *ms* and 750 *ms* in the RT *x* plot, the laser scanner and the scabor stereo vision system deliver object state observations that are similar in their deviation from the DGPS. The fusion in this region achieves a less accurate estimation of the object state observations than the sensors. Between 750 *ms* and 1500 *ms* in the RT *x* plot, however, the laser scanner yields better results than the scabor stereo vision system and the fusion results become comparable to the laser scanner results.

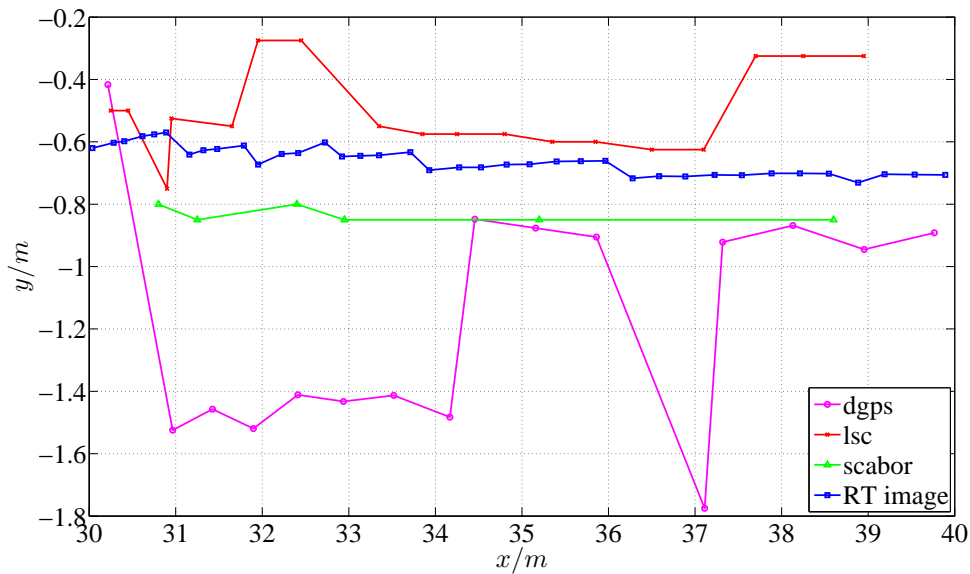


Figure 5.6.: Time-triggered unsynchronized BUFF configuration scenario 1, $x - y$ plot

The situation in the RT y plot is different in that the variations in the DGPS are greater than the variations in the sensors. Furthermore, it can be observed that the object state observations from the sensors rather underly a systematic error than a statistical zero mean error, at least for the duration visualized in Figure 5.7.

Figure 5.8 and Figure 5.9 depict the time-triggered unsynchronized ADVA configuration in a scenario 1 test drive.

As the BUFF and ADVA configuration have the same underlying schedule for the sensors and the bus system, Figure 5.9 is identical to Figure 5.7 in the DGPS and sensor plots. However, differences arise in the fusion schedule, which result in a different fusion plot, as can be recognized when, e.g., comparing the points indicated by reference signs [a](#) and [b](#).

Figure 5.10 and Figure 5.11 depict the time-triggered synchronized configuration in a scenario 1 test drive.

The test drive continues over 1000 ms within which the tracked vehicle moves from 30 m to 40 m in the positive x direction, which leads to an average speed of $v_x = 10 \frac{m}{s}$. Whereas the variations in the DGPS is much smaller than in the preceding figures, the variations in the sensors are about the same in size.

Figure 5.12 and Figure 5.13 depict the time-triggered unsynchronized BUFF configuration in a scenario 2 test drive.

The test drive continues over 1500 ms within which the tracked vehicle moves from 30 m to 17 m in the negative x direction respectively from 2 m to $-0.5 m$

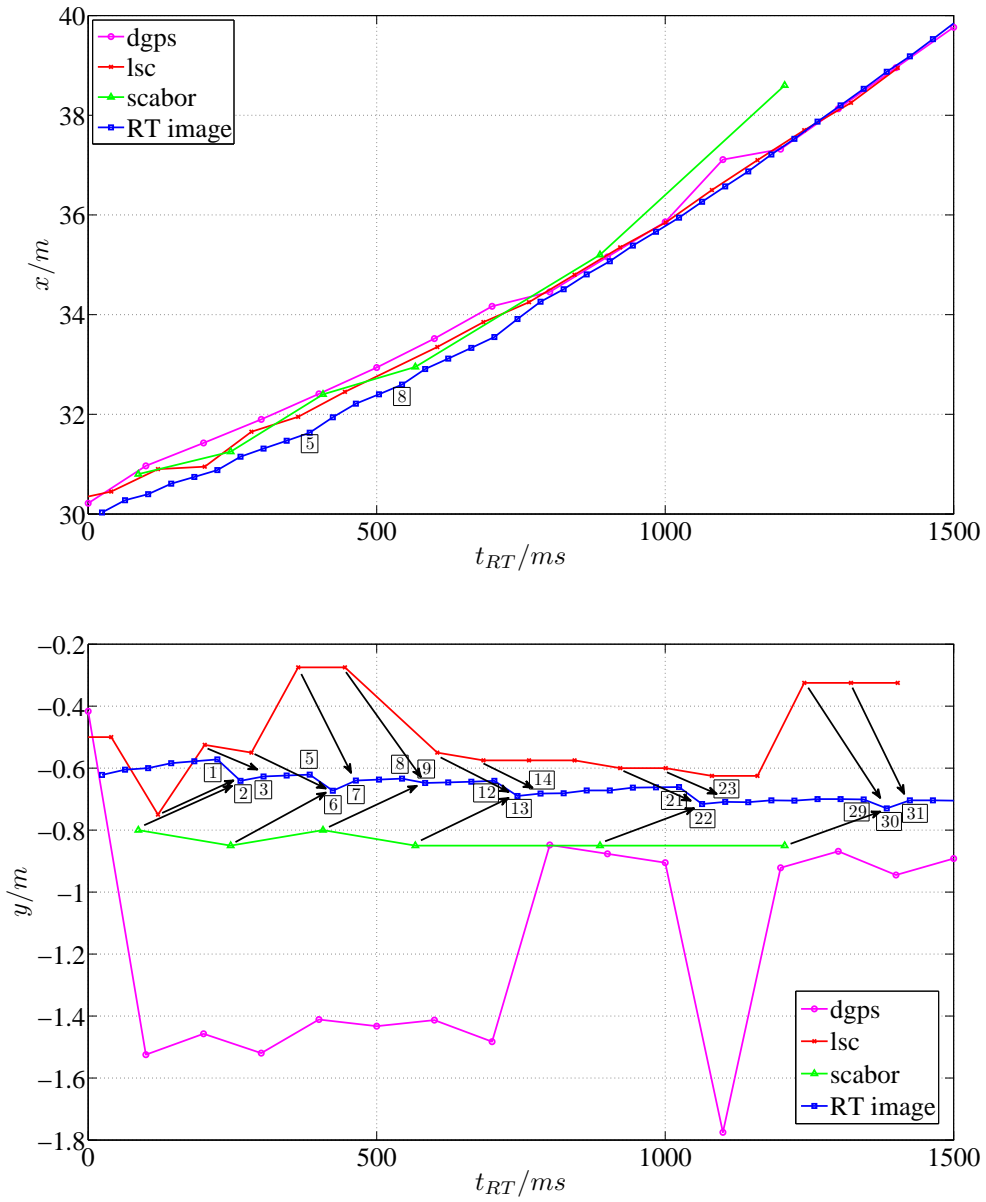


Figure 5.7.: Time-triggered unsynchronized BUFF configuration scenario 1, $RT - x$ plot, $RT - y$ plot

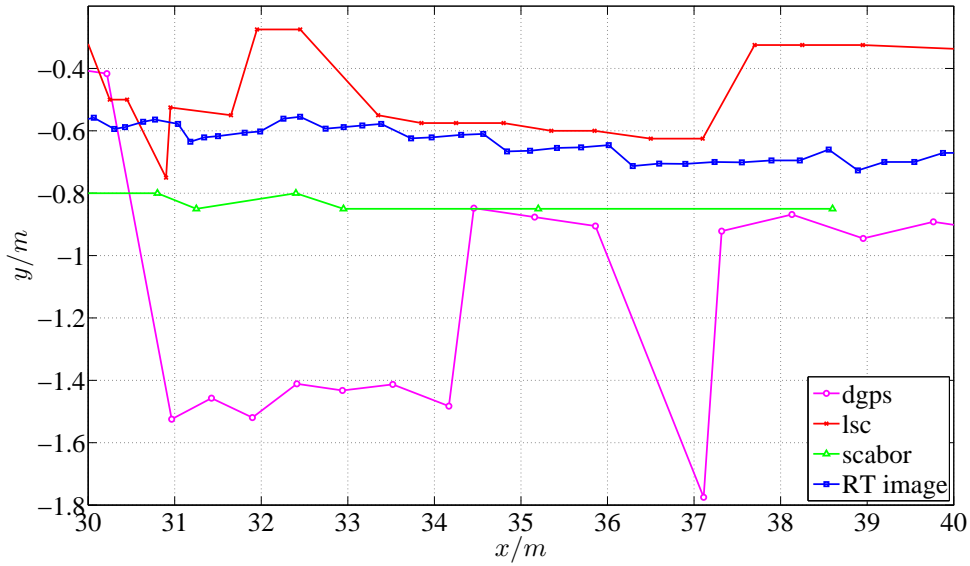


Figure 5.8.: Time-triggered unsynchronized ADVA configuration scenario 1, $x - y$ plot

in the negative y direction. This leads to an average speed of $v_x = -8.7 \frac{m}{s}$ and $v_y = -1.7 \frac{m}{s}$. In the RT x plot, the accuracy of the scabor stereo vision system and the laser scanner is about the same for the first $500 ms$ but then the laser scanner object state observation accuracy continuously degrades. In the RT y plot, the laser scanner yields better results than the scabor stereo vision system over the whole duration of the test drive.

Figure 5.14 and Figure 5.15 depict the time-triggered unsynchronized ADVA configuration in a scenario 2 test drive.

As the BUFF and ADVA configuration have the same underlying schedule for the sensors and the bus system, the DGPS and sensor plots in Figure 5.9 are identical to the respective plots in Figure 5.7. However, differences arise in the fusion schedule, which result in a different fusion plot, as can be recognized when, e.g., comparing the points indicated by reference signs [17] and [25].

Figure 5.16 and Figure 5.17 depict the time-triggered synchronized configuration in a scenario 2 test drive.

The test drive continues over $1500 ms$ within which the tracked vehicle moves from $30 m$ to $15 m$ in the negative x direction respectively from $2.5 m$ to $-0.5 m$ in the negative y direction. This leads to an average speed of $v_x = -10 \frac{m}{s}$ and $v_y = -2 \frac{m}{s}$. In the RT x plot, the accuracy of the scabor stereo vision system and the laser scanner is about the same. In the RT y plot, the scabor stereo vision system yields better results than the laser scanner over the whole duration of the

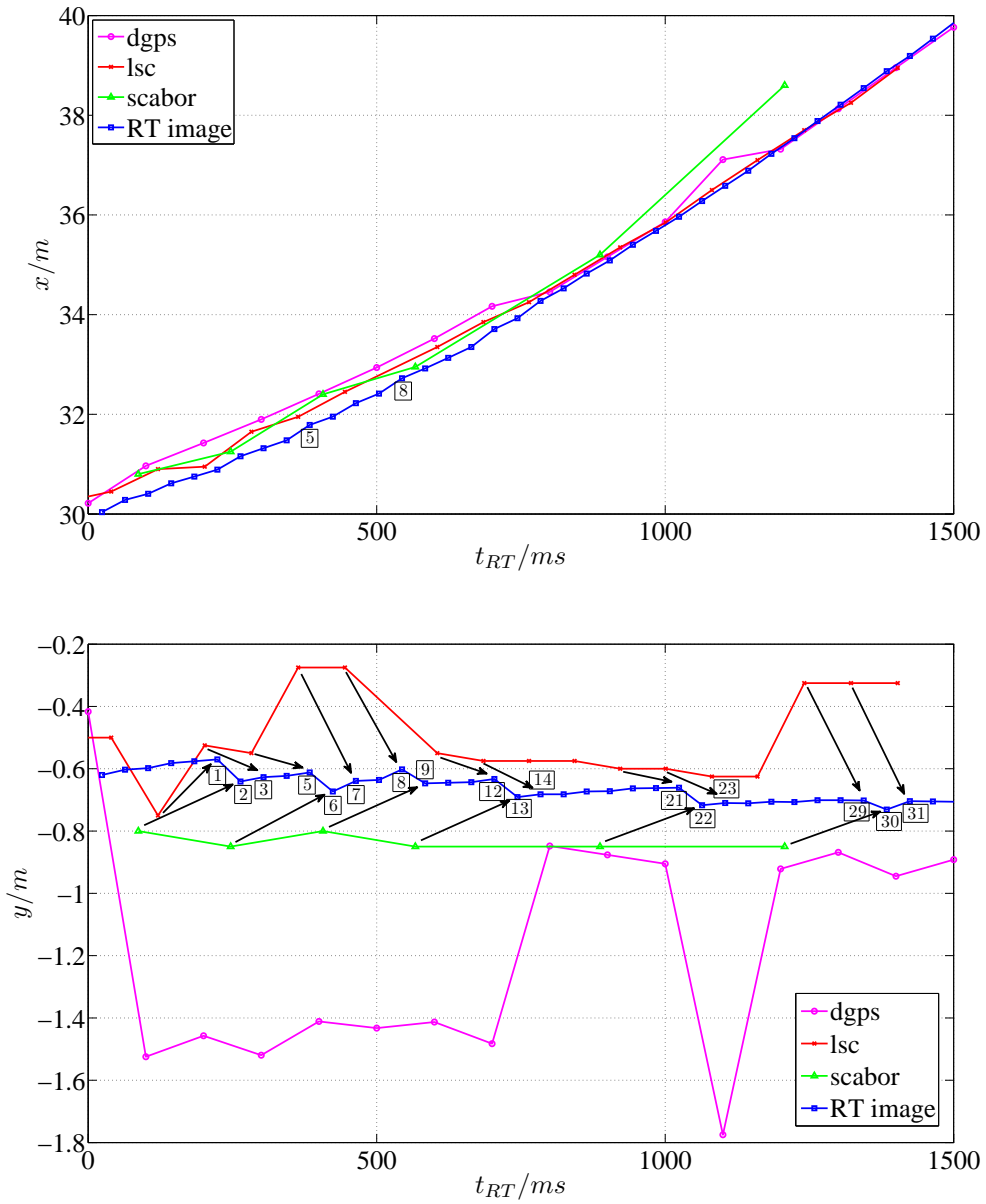


Figure 5.9.: Time-triggered unsynchronized ADVA configuration scenario 1, $RT - x$ plot, $RT - y$ plot

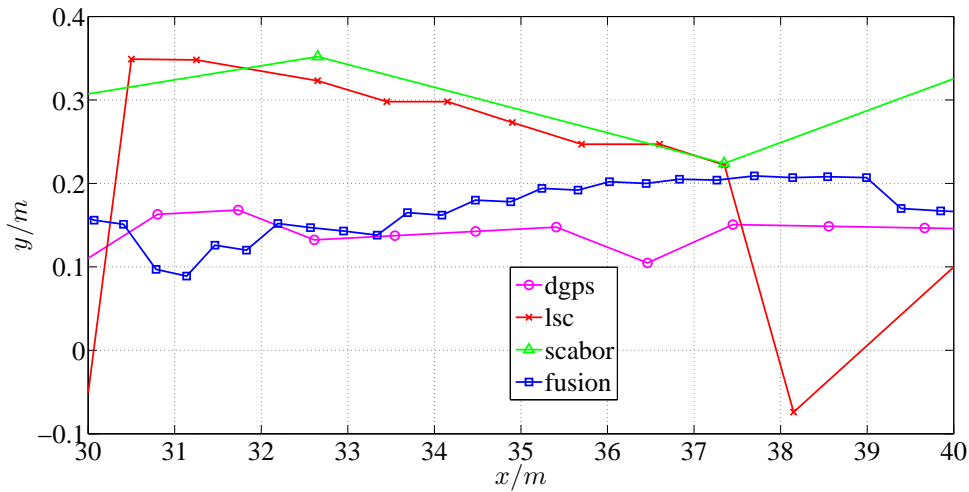


Figure 5.10.: Time-triggered synchronized configuration scenario 1, $x - y$ plot

test drive.

5.7. Discussion of Experimental Results

When analyzing the figures, it becomes clear that the test drives are not suited for comparing the MP of the configurations. This has several reasons:

- The laser scanner and the scabor trajectories do not seem to show a zero mean Gaussian distributed error over the selected test drives;
- The DGPS shows greater x and y variations than the sensors at least for a subset of the test drives which makes it impossible to use the DGPS as a reference;² and
- Even if the above problems could be overcome by using a large number of test drives, or tuning the sensors to the selected scenarios, it would remain unclear what number of test drives would enable a comparison where a MP difference in the range of 20% can be validated.

For these reasons the experimental results could not be used to validate the simulated results in the value domain, but only to plausibilize the configuration behavior.

²This problem is further complicated by the fact that speed and acceleration values have to be derived by calculating the time derivatives of the x and y trajectories which is problematic as the x and y trajectories are not continuously measured and random errors in the x and y trajectories might be augmented due to differentiating.

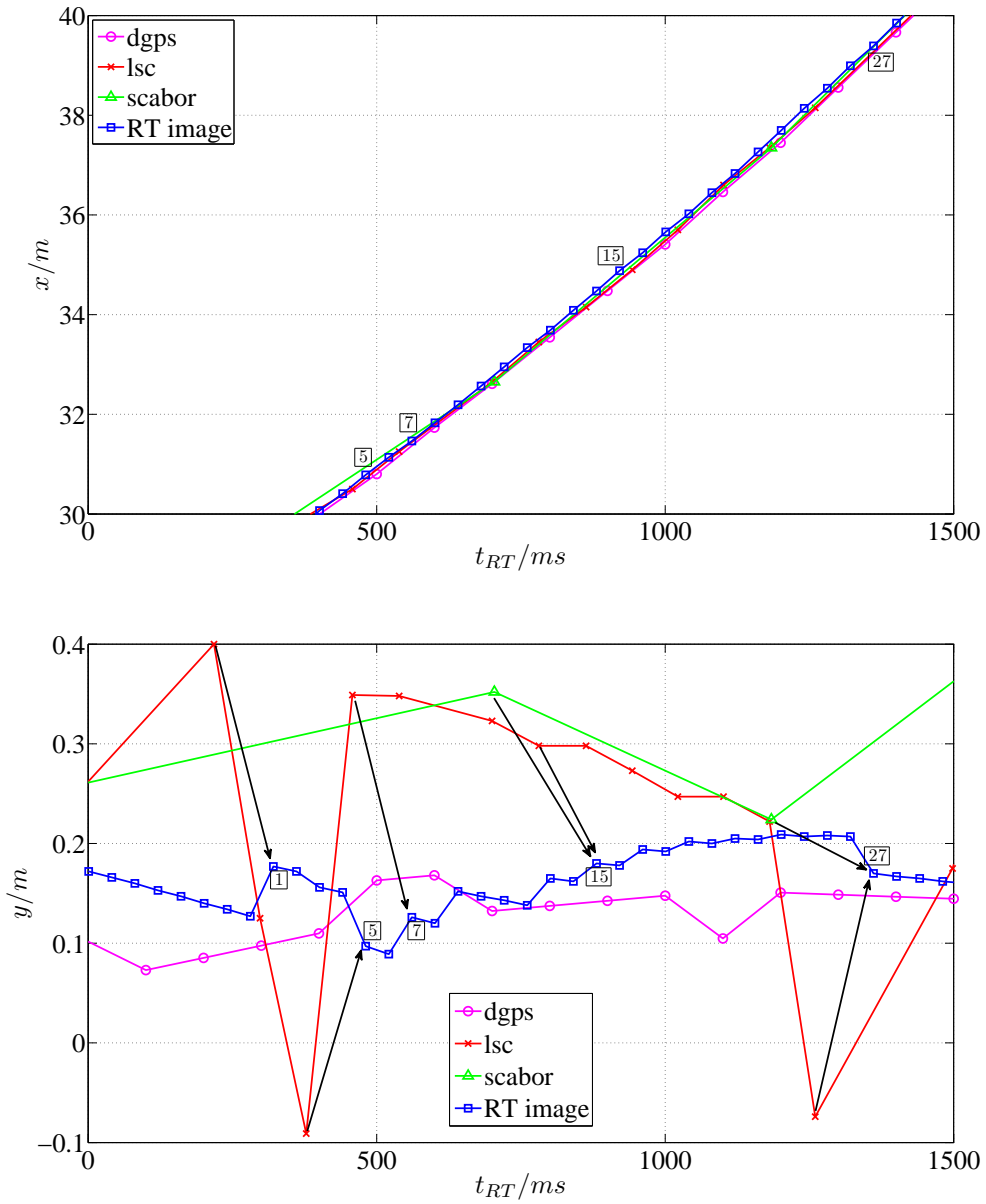


Figure 5.11.: Time-triggered synchronized configuration scenario 1, $RT - x$ plot, $RT - y$ plot

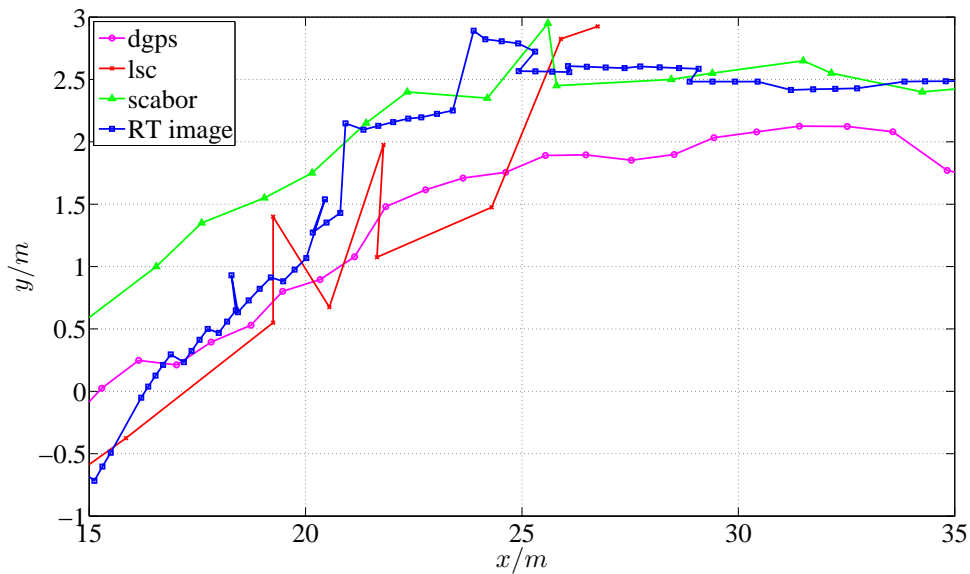


Figure 5.12.: Time-triggered unsynchronized BUFF configuration scenario 2, $x - y$ plot

5.7.1. Scenario 1

The time-triggered unsynchronized BUFF and the time-triggered unsynchronized ADVA configurations as depicted in Figures 5.7 and 5.9 differ in two aspects. The first aspect is the use of an advanced algorithm when fusing the object state observations of the scabor stereo vision system in the time-triggered unsynchronized ADVA configuration. The second aspect is the use of an object state observation buffer in the time-triggered unsynchronized BUFF configuration, which delays the fusion of object state observations of the laser scanner. Turning to the RT image of the object state indicated by reference sign [1] in Figure 5.9, said RT image comprises an object state observation from the laser scanner as indicated by the corresponding arrow. In Figure 5.7, said object state observation from the laser scanner is delayed by the need to chronologically sort the object state observations in the object state observation buffer. Accordingly said object state observation from the laser scanner is comprised in the RT image of the object state indicated by reference sign [2]. The difference can be best seen when regarding the RT images indicated by reference signs [5] and [8]. In the time-triggered unsynchronized ADVA configuration, said RT images of the object states are more directed to the y value of the laser scanner than to the y value of the scabor stereo vision system, as the most recent object state observation has been received from the laser scanner. When regarding the RT images indicated by reference signs [6] and [9], it becomes apparent that as soon as the same object state observations are fused with the image of the object states, the

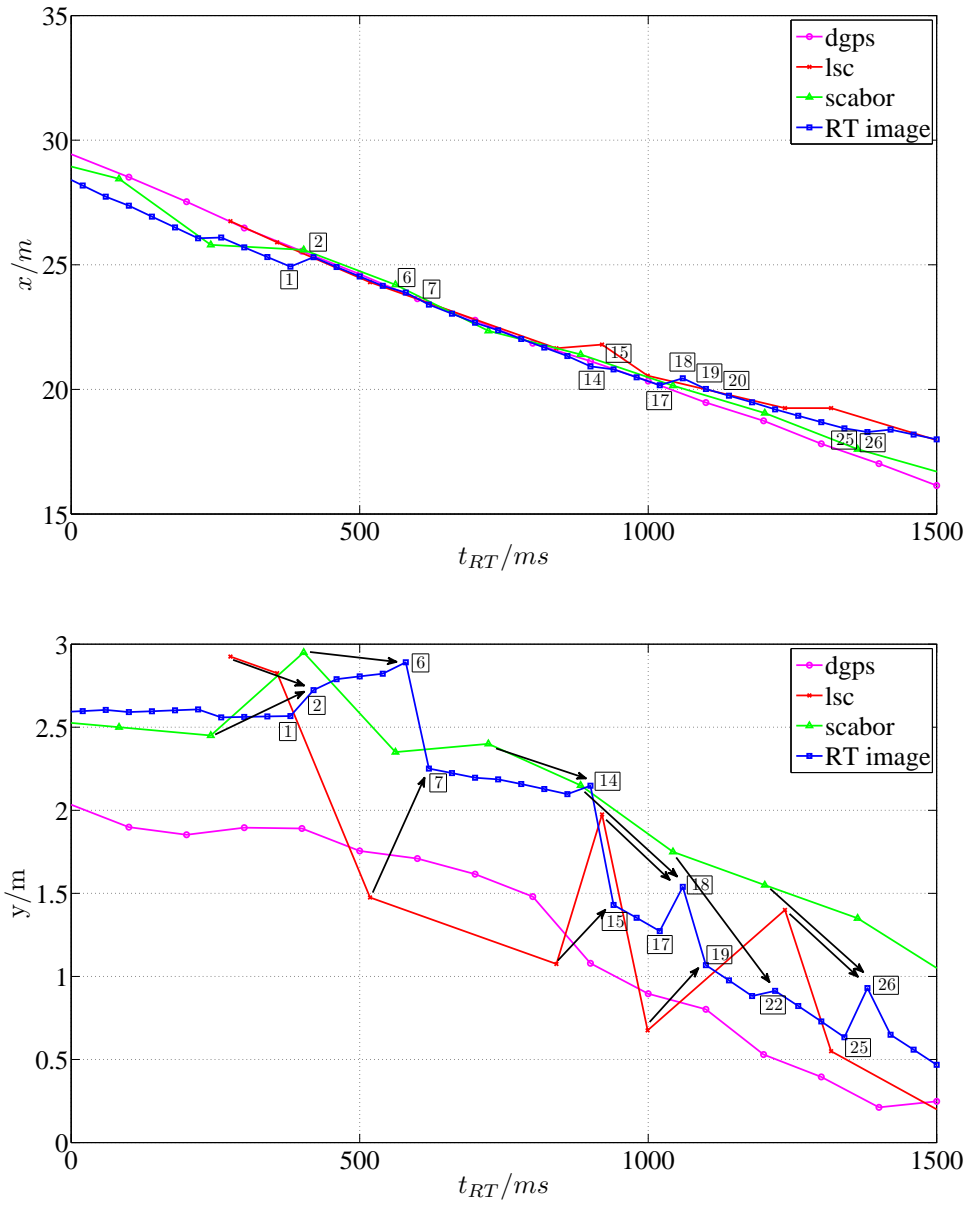


Figure 5.13.: Time-triggered unsynchronized BUFF configuration scenario 2, $RT - x$ plot, $RT - y$ plot

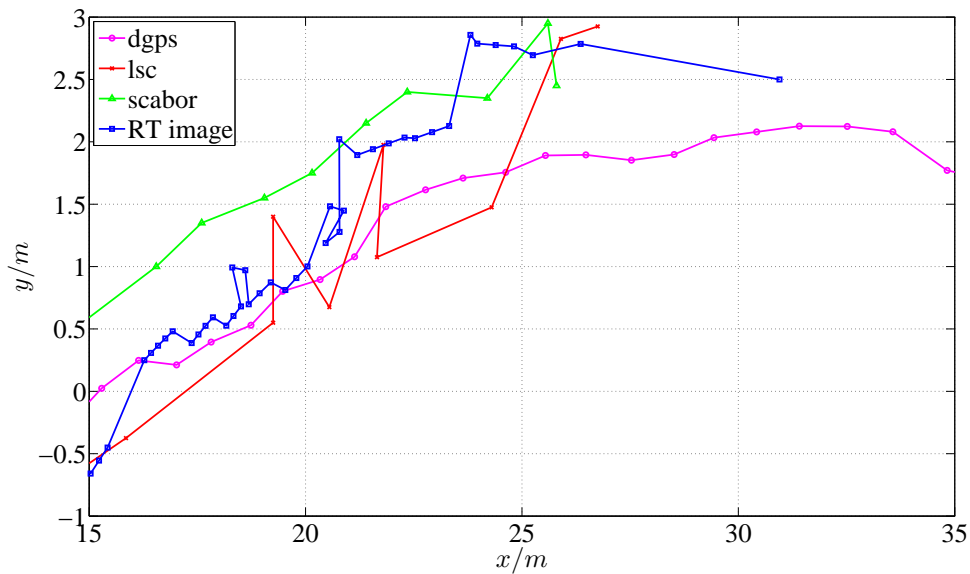


Figure 5.14.: Time-triggered unsynchronized ADVA configuration scenario 2, $x - y$ plot

predicted RT images of the object states are identical. This comparison therefore shows that the simulated superiority of the time-triggered unsynchronized ADVA configuration over the time-triggered unsynchronized BUFF configuration is the direct result of the time-triggered unsynchronized ADVA configuration being better reactive to dynamic changes as indicated by reference signs [5] and [8] in the RT x plot.

With regard to the time-triggered synchronized configuration as depicted in Figure 5.11 it should be noted that the reactivity of the time-triggered synchronized configuration is not much less than the time-triggered unsynchronized ADVA configuration as the difference in the prediction PH is very small compared to the sensor CTs. Furthermore, it might be interesting to consider that the initial estimate that the concentration of object state observations around certain points in time might stabilize the RT images of the object state is not correct, as can be seen when considering the RT images of the object state indicated by reference signs [1], [5], [7], [15] and [27]. There are two reasons for this. Firstly, an object may not be seen by every sensor in every scan or frame as indicated by reference signs [1], [5], and [7]. Secondly, even if two object state observations of the same object are provided by the sensors, the errors may have the same algebraic sign in the value domain as indicated by reference sign [15].

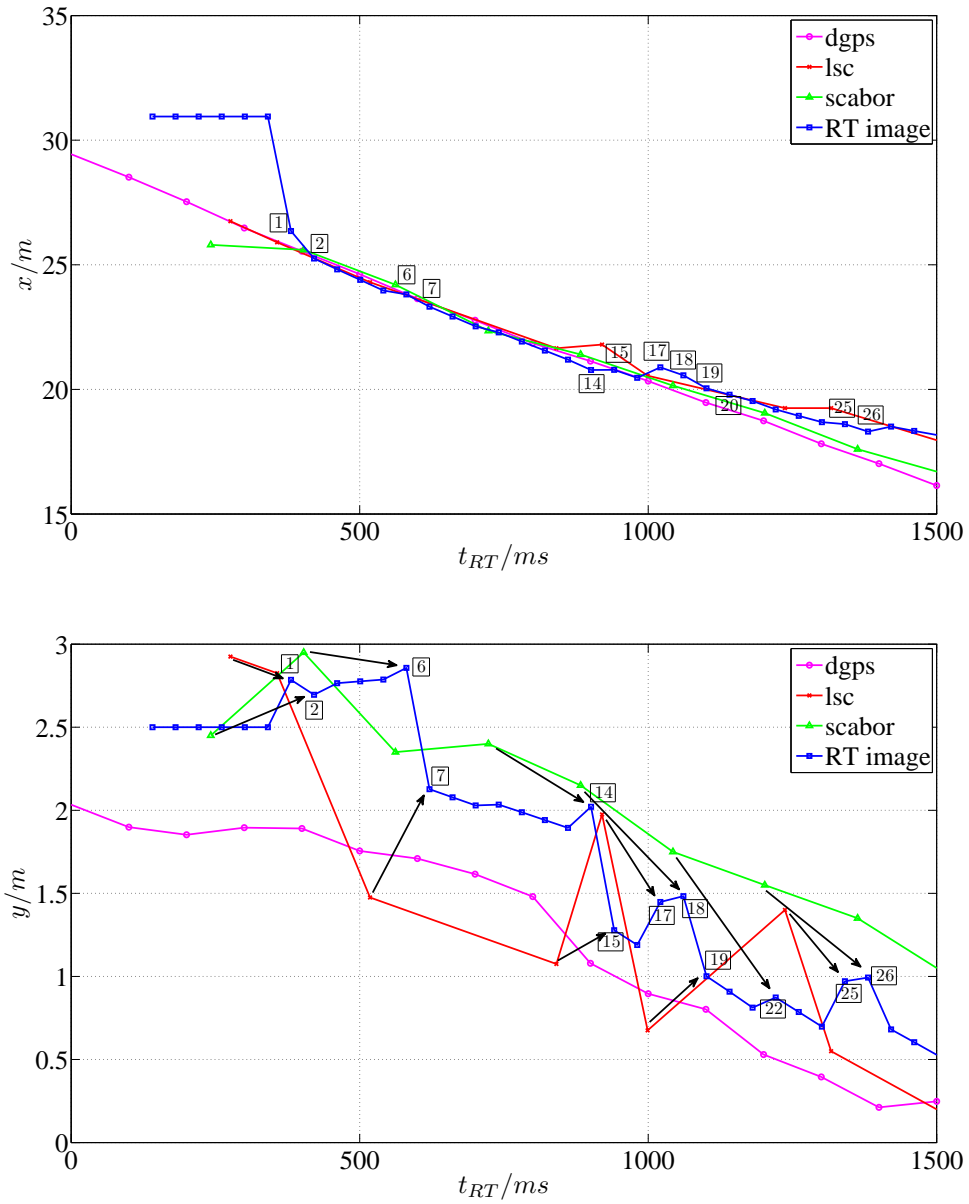


Figure 5.15.: Time-triggered unsynchronized ADVA configuration scenario 2, $RT - x$ plot, $RT - y$ plot

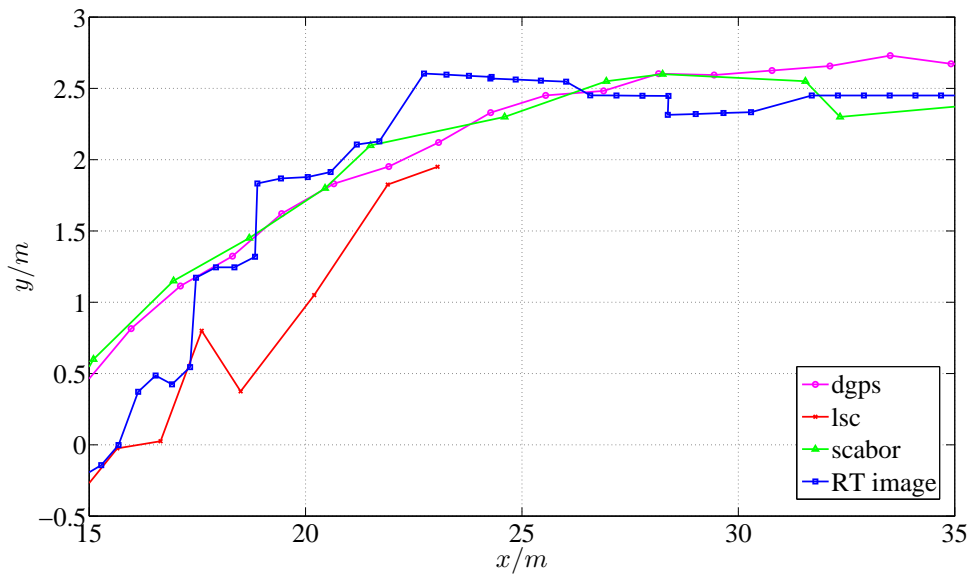


Figure 5.16.: Time-triggered synchronized configuration scenario 2, $x - y$ plot

5.7.2. Scenario 2

The time-triggered unsynchronized BUFF and the time-triggered unsynchronized ADVA configurations as depicted in Figures 5.13 and 5.15 show another aspect that can arise in a test drive and that has been neglected in the simulation. Thus, it may happen, that an object state observation of a sensor is not associated correctly. In the time-triggered unsynchronized BUFF configuration as depicted in Figure 5.13, the object is tracked during the full length of the plotted test drive, whereas in the time-triggered unsynchronized ADVA configuration as depicted in Figure 5.15, the object is detected during the first 500 ms of the plotted test drive. However, besides this difference the same aspects as mentioned for scenario 1 can be recognized when regarding reference signs [17] and [25], which show the higher reactivity of the time-triggered unsynchronized ADVA configuration to dynamic changes.

With regard to the time-triggered synchronized configuration as depicted in Figure 5.17, the test drive shows a similar situation as discussed for scenario 1. Again, there are only a subset of object state observation pairs, e.g. indicated by [1].

5.8. Discussion

When analyzing test drives for the time-triggered configurations, it became clear that the test drives are not suited for a validation of the MP comparison. This is due to sensors not showing a zero mean Gaussian distributed error over selected test

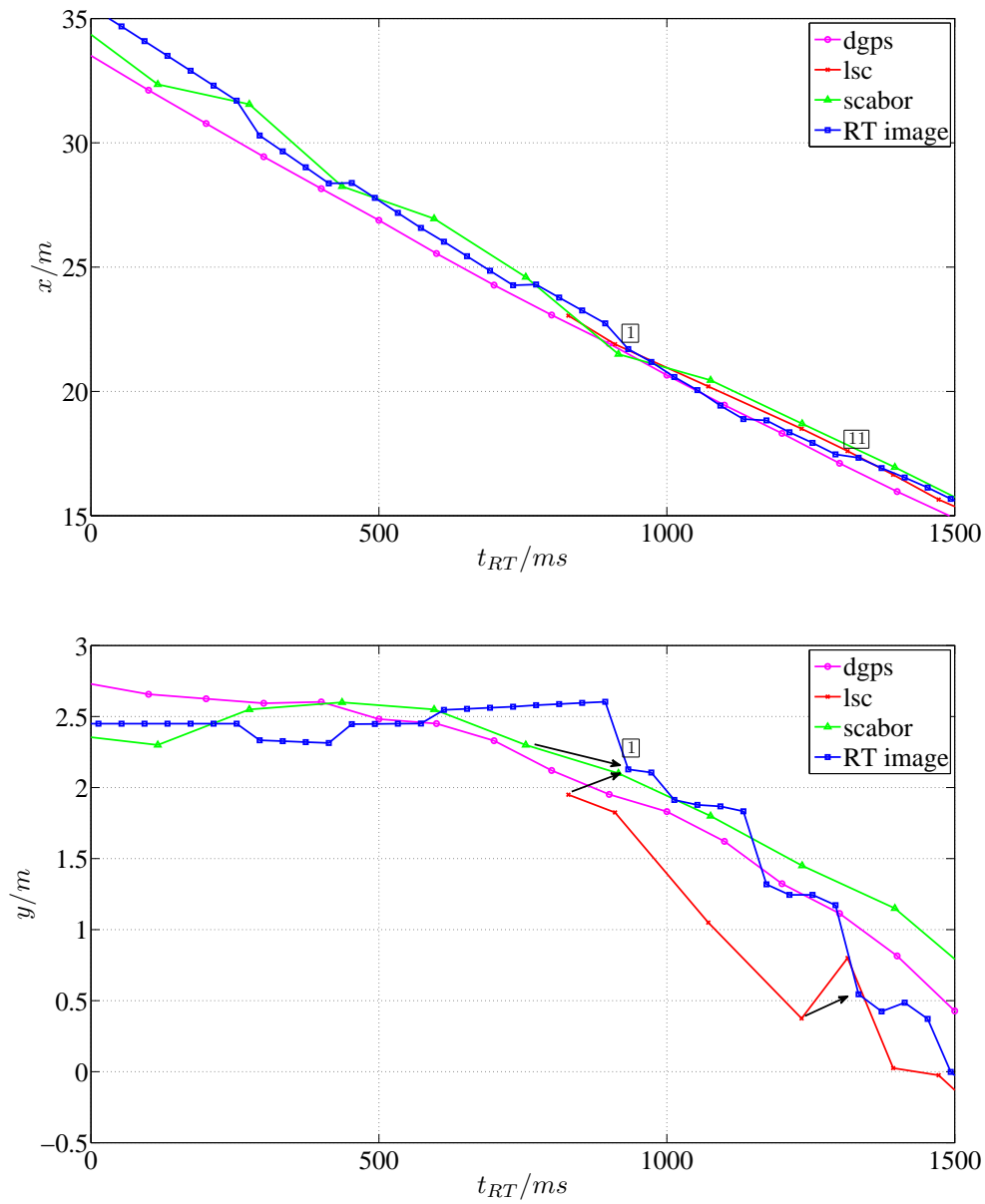


Figure 5.17.: Time-triggered synchronized configuration scenario 2, $RT - x$ plot, $RT - y$ plot

drives, the DGPS showing greater x and y variations than the sensors at least for a subset of the test drives making it impossible to use the DGPS as a reference or to determine what number of test drives might be necessary to achieve the desired precision.

When plausibilizing the simulation results with the test drives, it could be shown that the time-triggered unsynchronized ADVA configuration is more reactive to dynamic changes than the time-triggered unsynchronized BUFF configuration. Furthermore, the test drives show that the association which has been neglected in the simulation due to simplicity might have an effect on the MP comparison and should be investigated in the future.

5.9. Chapter Summary

In this chapter the case study set up has been described. With this set up, four test drives were performed. The test drives comprised of two test drives for the time-triggered unsynchronized BUFF and time-triggered unsynchronized ADVA configuration and two test drives for the time-triggered synchronized configuration, where the two test drives were performed in two different scenarios. The analysis of the test drives showed that they are unsuited to validate the MP comparison of the preceding chapter. However, as the time-triggered unsynchronized BUFF and the time-triggered unsynchronized ADVA configuration have the same underlying sensor and bus schedule, the test drives were used to plausibilize the configuration behavior observed in the simulation.

6. Discussion

The goal of this thesis was to analyze the feasibility of a paradigm shift toward time-triggered ADASOT. Heretofore a generic model of a state-of-the-art ADASOT and a generic model of a time-triggered ADASOT have been developed, taking into account the most recent research in object tracking. The time-triggered model has been developed following the time-triggered sensor fusion model proposed by Elmenreich and Pitzek [40]. The model comprises two heterogeneous sensors, an object tracking subsystem and a feature service subsystem. The aforementioned components are interconnected via a time-triggered bus system that synchronizes the clocks of all nodes and establishes a global time base, following the time-triggered architecture proposed by Kopetz et al. [79].

6.1. State-of-the-Art vs. Time-Triggered Configurations

The MP of both models has been simulated for different configurations, the configurations differing in the sensor and bus system schedules and the treatment of OOSMs. The schedules differ in that the sensors are either synchronized, i.e., the sensors produce object state observations with the same time-stamp, or unsynchronized, i.e., the sensors are scheduled to minimize times during which no measurement is taken. The treatment of OOSM is either realized by a buffer in which object state observations are chronologically sorted, or by an ADVA approach as proposed by Bar-Shalom [9].

The results show that for the chosen parameter space, the state-of-the-art ADVA configuration yields the best results. However, the results also show that there are points in parameter space where the state-of-the-art ADVA configuration is out-matched by the state-of-the-art BUFF configuration, the time-triggered unsynchronized ADVA configuration or the time-triggered synchronized configuration.

A detailed analysis has shown that the integrated process noise in combination with the sequence of ST to RT prediction intervals plays the dominant role in the factors that influence the MP measure. Accordingly, it is favorable to minimize the mean of the sequence of ST to RT prediction intervals but also to minimize the variance of the sequence as the influence is not restricted to the first order of the intervals. Furthermore, the influence of said sequence of intervals is further

strengthened by an increasing process noise power spectral density, as this linearly affects the integrated process noise.

Accordingly, the state-of-the-art configurations are favorable when the sensor PTs show very high variations. However, with decreasing sensor PT variation, the time-triggered configurations outmatch the state-of-the-art configurations for two reasons. The first reason is the increasing mean of the sequence between ST to RT. The second reason is that the time-triggered configurations show a smaller variation in the sequence between ST and RT which is advantageous when considering the higher order dependence of the mean trace of the integrated process noise.

As a result, the state-of-the-art configurations show weaknesses in situations of high risk potential, because such situations are characterized by a high number of objects which leads to low sensor PT variation and/or a fast changing environment which is represented by a high process noise power spectral density. Thus for linearized state-space models and for potentially dangerous scenarios with high dynamics in state space parameter derivatives which are not modeled, the time-triggered model becomes advantageous which promotes the paradigm shift toward time-triggered advanced driver assistance systems based on multi-sensor object tracking.

Furthermore, the analysis has provided insight into how the state-of-the-art configurations and the time-triggered configurations react to the parameter variations. The analysis shows that the state-of-the-art BUFF configuration is outmatched by the state-of-the-art ADVA configuration due to the higher sequence of ST to RT prediction intervals. The exception being very slow fusion systems where the additional processing resources required for the fusion of OOSM by an ADVA configuration overthrows the effect of the smaller sequence of ST to RT prediction intervals. Moreover, the simulation has shown that the time-triggered unsynchronized BUFF configuration is outmatched by the time-triggered unsynchronized ADVA configuration and/or the time-triggered synchronized configuration due to their lower sequence in ST to RT prediction intervals. In regard to the time-triggered unsynchronized ADVA configuration and the time-triggered synchronized configuration, the analysis shows that the time-triggered unsynchronized ADVA configuration is outmatched by the time-triggered synchronized configuration for slow and fast object tracking subsystems but outmatches the time-triggered synchronized configuration for an medium object tracking subsystem due to the schedule “switch” in the scheduling of the fusion tasks.

The time-triggered configurations have then been tested in a case study using a Volkswagen Touran passenger car. The gained results from test drives in two scenarios have shown that they are not suited to validate the results gained through analysis of the simulation results. However as the time-triggered unsynchronized BUFF configuration and the time-triggered unsynchronized ADVA configuration have the same sensor and bus schedule, the simulation results could at least in part be plausibilized with these test drives.

6.2. Outlook

As a future research objective, it would be interesting to incorporate the so far neglected association process in the analysis. This is especially interesting as the association might favor the time-triggered synchronized configuration. This might be due to the fact that an update with very precise object state observations might ameliorate the subsequent association of less accurate object state observations which have the same time-stamp.

A further interesting topic might be the possibility to establish an adaptive scheduling concept for the object tracking subsystem and possibly also for the sensors and the bus system where the schedule switches in order to support the most favorable configuration, depending on an estimated process noise power spectral density and an interpolated fusion PT.

Finally, it should be evaluated in which situations the absolute expected performance differences between the configurations would really lead to a appreciable difference in the feature service performance.

A. Observation Preprocessing Variance Modeling

The following state and state transition matrices are used for modeling the sensor PTs and fusion PTs for $c = 0.6$,

$$m_{c=0.6}^{\vec{}} = \begin{pmatrix} \Delta t_{PT}^{sens1}/ms & \Delta t_{PT}^{sens2}/ms & \Delta t_{PT}^{fus}/ms \\ 96 & 48 & \{1, 3, 6, 9, 12, 15\} \\ 104 & 52 & \{1, 4, 6, 10, 13, 17\} \\ 112 & 56 & \{1, 4, 7, 11, 14, 18\} \\ 120 & 60 & \{1, 4, 7, 11, 15, 19\} \\ 128 & 64 & \{2, 4, 8, 12, 16, 20\} \\ 136 & 68 & \{2, 5, 8, 13, 17, 22\} \\ 144 & 72 & \{2, 5, 9, 14, 18, 23\} \\ 152 & 76 & \{2, 5, 9, 14, 19, 24\} \\ 160 & 80 & \{2, 5, 10, 15, 20, 25\} \end{pmatrix} \quad (A.1)$$

$$M_{c=0.6} = \begin{pmatrix} 0.5 & 0.5 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0.25 & 0.5 & 0.25 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0.25 & 0.5 & 0.25 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0.25 & 0.5 & 0.25 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0.25 & 0.5 & 0.25 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0.25 & 0.5 & 0.25 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0.25 & 0.5 & 0.25 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0.25 & 0.5 & 0.25 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0.5 & 0.5 \end{pmatrix} \quad (A.2)$$

for $c = 0.7$,

$$m_{c=0.7}^{\vec{}} = \begin{pmatrix} \frac{\Delta t_{\text{PT}}^{\text{sens1}}}{ms} & \frac{\Delta t_{\text{PT}}^{\text{sens2}}}{ms} & \frac{\Delta t_{\text{PT}}^{\text{fus}}}{ms} \\ 112 & 56 & \{1, 4, 7, 11, 14, 18\} \\ 120 & 60 & \{1, 4, 7, 11, 15, 19\} \\ 128 & 64 & \{2, 4, 8, 12, 16, 20\} \\ 136 & 68 & \{2, 5, 8, 13, 17, 22\} \\ 144 & 72 & \{2, 5, 9, 14, 18, 23\} \\ 152 & 76 & \{2, 5, 9, 14, 19, 24\} \\ 160 & 80 & \{2, 5, 10, 15, 20, 25\} \end{pmatrix} \quad (\text{A.3})$$

$$M_{c=0.7} = \begin{pmatrix} 0.5 & 0.5 & 0 & 0 & 0 & 0 & 0 \\ 0.25 & 0.5 & 0.25 & 0 & 0 & 0 & 0 \\ 0 & 0.25 & 0.5 & 0.25 & 0 & 0 & 0 \\ 0 & 0 & 0.25 & 0.5 & 0.25 & 0 & 0 \\ 0 & 0 & 0 & 0.25 & 0.5 & 0.25 & 0 \\ 0 & 0 & 0 & 0 & 0.25 & 0.5 & 0.25 \\ 0 & 0 & 0 & 0 & 0 & 0.5 & 0.5 \end{pmatrix} \quad (\text{A.4})$$

for $c = 0.8$, and

$$m_{c=0.8}^{\vec{}} = \begin{pmatrix} \frac{\Delta t_{\text{PT}}^{\text{sens1}}}{ms} & \frac{\Delta t_{\text{PT}}^{\text{sens2}}}{ms} & \frac{\Delta t_{\text{PT}}^{\text{fus}}}{ms} \\ 128 & 64 & \{2, 4, 8, 12, 16, 20\} \\ 136 & 68 & \{2, 5, 8, 13, 17, 22\} \\ 144 & 72 & \{2, 5, 9, 14, 18, 23\} \\ 152 & 76 & \{2, 5, 9, 14, 19, 24\} \\ 160 & 80 & \{2, 5, 10, 15, 20, 25\} \end{pmatrix} \quad (\text{A.5})$$

$$M_{c=0.8} = \begin{pmatrix} 0.5 & 0.5 & 0 & 0 & 0 \\ 0.25 & 0.5 & 0.25 & 0 & 0 \\ 0 & 0.25 & 0.5 & 0.25 & 0 \\ 0 & 0 & 0.25 & 0.5 & 0.25 \\ 0 & 0 & 0 & 0.5 & 0.5 \end{pmatrix} \quad (\text{A.6})$$

for $c = 0.9$.

$$m_{c=0.9}^{\vec{}} = \begin{pmatrix} \frac{\Delta t_{\text{PT}}^{\text{sens1}}}{ms} & \frac{\Delta t_{\text{PT}}^{\text{sens2}}}{ms} & \frac{\Delta t_{\text{PT}}^{\text{fus}}}{ms} \\ 144 & 72 & \{2, 5, 9, 14, 18, 23\} \\ 152 & 76 & \{2, 5, 9, 14, 19, 24\} \\ 160 & 80 & \{2, 5, 10, 15, 20, 25\} \end{pmatrix} \quad (\text{A.7})$$

$$M_{c=0.9} = \begin{pmatrix} 0.5 & 0.5 & 0.0 \\ 0.25 & 0.5 & 0.25 \\ 0 & 0.5 & 0.5 \end{pmatrix} \quad (\text{A.8})$$

B. Process Noise of Linearized State Evolution

In order to calculate the process noise of a linearized state evolution, formula B.1 for the propagation of the ECM $P(t_k, t_{k-1})$ in an EKF is regarded.

$$\begin{aligned}
P(t_k|t_{k-1}) &:= E \left[\left(\hat{\vec{x}}(t_k|t_{k-1}) - \vec{x}(t_k) \right) \cdot \left(\hat{\vec{x}}(t_k|t_{k-1}) - \vec{x}(t_k) \right)^T \right] = \\
&= E \left[\left(\hat{\vec{x}}(t_{k-1}|t_{k-1}) + \hat{\vec{f}}(\vec{x} = \hat{\vec{x}}(t_{k-1}|t_{k-1}), t_{k-1}) \cdot (t_k - t_{k-1}) - \right. \right. \\
&\quad \left. \left. - \vec{x}(t_{k-1}) - \int_{t_{k-1}}^{t_k} \vec{f}(\vec{x}(t), t) dt \right) \cdot \right. \\
&\quad \left. \cdot \left(\hat{\vec{x}}(t_{k-1}|t_{k-1}) + \hat{\vec{f}}(\vec{x} = \hat{\vec{x}}(t_{k-1}|t_{k-1}), t_{k-1}) \cdot (t_k - t_{k-1}) - \right. \right. \\
&\quad \left. \left. - \vec{x}(t_{k-1}) - \int_{t_{k-1}}^{t_k} \vec{f}(\vec{x}(t), t) dt \right)^T \right] + Q_{t_{k-1}} \tag{B.1}
\end{aligned}$$

Expanding $\int_{t_{k-1}}^{t_k} \vec{f}(\vec{x}(t), t) dt$ around t_{k-1}

$$\int_{t_{k-1}}^{t_k} \vec{f}(\vec{x}(t), t) dt = \vec{f}(\vec{x}(t_{k-1}), t_{k-1}) \cdot (t_k - t_{k-1}) + \int_{t_{k-1}}^{t_k} \vec{f}'(\vec{x}(t), t) (t - t_{k-1}) dt \tag{B.2}$$

leads to formula B.3

$$\begin{aligned}
P(t_k|t_{k-1}) = E & \left[\left(\hat{\vec{x}}(t_{k-1}|t_{k-1}) + \hat{\vec{f}}(\vec{x} = \hat{\vec{x}}(t_{k-1}|t_{k-1}), t_{k-1}) \cdot (t_k - t_{k-1}) - \right. \right. \\
& - \vec{x}(t_{k-1}) - \vec{f}(\vec{x}(t_{k-1}), t_{k-1}) \cdot (t_k - t_{k-1}) - \\
& \left. \left. - \int_{t_{k-1}}^{t_k} \vec{f}'(\vec{x}(t), t) (t - t_{k-1}) dt \right) \cdot \right. \\
& \cdot \left(\hat{\vec{x}}(t_{k-1}|t_{k-1}) + \hat{\vec{f}}(\vec{x} = \hat{\vec{x}}(t_{k-1}|t_{k-1}), t_{k-1}) \cdot (t_k - t_{k-1}) - \right. \\
& - \vec{x}(t_{k-1}) - \vec{f}(\vec{x}(t_{k-1}), t_{k-1}) \cdot (t_k - t_{k-1}) - \\
& \left. \left. - \int_{t_{k-1}}^{t_k} \vec{f}'(\vec{x}(t), t) (t - t_{k-1}) dt \right)^T \right] + Q_{t_{k-1}} \tag{B.3}
\end{aligned}$$

Expanding $\vec{f}(\vec{x}(t_{k-1}), t_{k-1})$ around $\hat{\vec{x}}(t_{k-1}|t_{k-1})$

$$\begin{aligned}
\vec{f}(\vec{x}(t_{k-1}), t_{k-1}) = & \vec{f}(\hat{\vec{x}}(t_{k-1}|t_{k-1}), t_{k-1}) + \\
& + \left. \frac{\partial \vec{f}}{\partial \vec{x}} \right|_{\vec{x}=\hat{\vec{x}}(t_{k-1}|t_{k-1})} (\vec{x}(t_{k-1}) - \hat{\vec{x}}(t_{k-1}|t_{k-1})) + \\
& + \int_{\hat{\vec{x}}(t_{k-1}|t_{k-1})}^{\vec{x}(t_{k-1})} \frac{\partial^2 \vec{f}}{\partial \vec{x}^2} (\vec{x} - \hat{\vec{x}}(t_{k-1}|t_{k-1})) d\vec{x} \tag{B.4}
\end{aligned}$$

with $\tilde{\vec{x}}(t_{k-1}) = \hat{\vec{x}}(t_{k-1}|t_{k-1}) - \vec{x}(t_{k-1})$ as well as $\Delta t = t_k - t_{k-1}$ leads to formula B.5

$$\begin{aligned}
P(t_k|t_{k-1}) = E & \left[\left(\tilde{\vec{x}}(t_{k-1}) + \hat{\vec{f}}(\vec{x} = \hat{\vec{x}}(t_{k-1}|t_{k-1}), t_{k-1}) \cdot \Delta t - \right. \right. \\
& - \left(\vec{f}(\hat{\vec{x}}(t_{k-1}|t_{k-1}), t_{k-1}) - \left. \frac{\partial \vec{f}}{\partial \vec{x}} \Big|_{\vec{x}=\hat{\vec{x}}(t_{k-1}|t_{k-1})} \cdot \tilde{\vec{x}}(t_{k-1}) \right) \cdot \Delta t - \right. \\
& - \int_{t_{k-1}}^{t_k} \vec{f}'(\vec{x}(t), t) (t - t_{k-1}) dt - \\
& \left. - \Delta t \cdot \int_{\hat{\vec{x}}(t_{k-1}|t_{k-1})}^{\vec{x}(t_{k-1})} \frac{\partial^2 \vec{f}}{\partial \vec{x}^2} (\vec{x} - \hat{\vec{x}}(t_{k-1}|t_{k-1})) d\vec{x} \right) \cdot \\
& \cdot \left(\tilde{\vec{x}}(t_{k-1}) + \hat{\vec{f}}(\vec{x} = \hat{\vec{x}}(t_{k-1}|t_{k-1}), t_{k-1}) \cdot \Delta t - \right. \\
& - \left(\vec{f}(\hat{\vec{x}}(t_{k-1}|t_{k-1}), t_{k-1}) - \left. \frac{\partial \vec{f}}{\partial \vec{x}} \Big|_{\vec{x}=\hat{\vec{x}}(t_{k-1}|t_{k-1})} \cdot \tilde{\vec{x}}(t_{k-1}) \right) \cdot \Delta t - \right. \\
& - \int_{t_{k-1}}^{t_k} \vec{f}'(\vec{x}(t), t) (t - t_{k-1}) dt - \\
& \left. - \Delta t \cdot \int_{\hat{\vec{x}}(t_{k-1}|t_{k-1})}^{\vec{x}(t_{k-1})} \frac{\partial^2 \vec{f}}{\partial \vec{x}^2} (\vec{x} - \hat{\vec{x}}(t_{k-1}|t_{k-1})) d\vec{x} \right)^T \Big] + Q_{t_{k-1}} \quad (\text{B.5})
\end{aligned}$$

With

$$\tilde{\vec{f}}(\vec{x} = \hat{\vec{x}}(t_{k-1}|t_{k-1}), t_{k-1}) := \hat{\vec{f}}(\vec{x} = \hat{\vec{x}}(t_{k-1}|t_{k-1}), t_{k-1}) - \vec{f}(\vec{x} = \hat{\vec{x}}(t_{k-1}|t_{k-1}), t_{k-1}) \quad (\text{B.6})$$

and

$$\left(I + F(\hat{\vec{x}}(t_{k-1}|t_{k-1}), t_{k-1}) \cdot \Delta t \right) \cdot \tilde{\vec{x}}(t_{k-1}) := \tilde{\vec{x}}(t_{k-1}) + \frac{\partial \vec{f}}{\partial \vec{x}} \Big|_{\vec{x}=\hat{\vec{x}}(t_{k-1}|t_{k-1})} \cdot \tilde{\vec{x}}(t_{k-1}) \cdot \Delta t \quad (\text{B.7})$$

equation B.5 is simplified to

$$\begin{aligned}
P(t_k|t_{k-1}) = E & \left[\left(\left(I + F(\hat{\vec{x}}(t_{k-1}|t_{k-1}), t_{k-1}) \cdot \Delta t \right) \cdot \tilde{\vec{x}}(t_{k-1}) + \right. \right. \\
& + \tilde{\vec{f}}(\hat{\vec{x}}(t_{k-1}|t_{k-1}), t_{k-1}) \cdot \Delta t - \int_{t_{k-1}}^{t_k} \vec{f}'(\vec{x}(t), t) (t - t_{k-1}) dt - \\
& - \left. \left. \frac{\partial \tilde{\vec{f}}}{\partial \vec{x}} \Big|_{\vec{x}=\hat{\vec{x}}(t_{k-1}|t_{k-1})} \cdot \tilde{\vec{x}}(t_{k-1}) \cdot \Delta t - \right. \right. \\
& \left. \left. - \Delta t \cdot \int_{\hat{\vec{x}}(t_{k-1}|t_{k-1})}^{\vec{x}(t_{k-1})} \frac{\partial^2 \vec{f}}{\partial \vec{x}^2} (\vec{x} - \hat{\vec{x}}(t_{k-1}|t_{k-1})) d\vec{x} \right) \cdot \right. \\
& \cdot \left(\left(I + F(\hat{\vec{x}}(t_{k-1}|t_{k-1}), t_{k-1}) \cdot \Delta t \right) \cdot \tilde{\vec{x}}(t_{k-1}) + \right. \\
& + \tilde{\vec{f}}(\hat{\vec{x}}(t_{k-1}|t_{k-1}), t_{k-1}) \cdot \Delta t - \int_{t_{k-1}}^{t_k} \vec{f}'(\vec{x}(t), t) (t - t_{k-1}) dt - \\
& - \left. \left. \frac{\partial \tilde{\vec{f}}}{\partial \vec{x}} \Big|_{\vec{x}=\hat{\vec{x}}(t_{k-1}|t_{k-1})} \cdot \tilde{\vec{x}}(t_{k-1}) \cdot \Delta t - \right. \right. \\
& \left. \left. - \Delta t \cdot \int_{\hat{\vec{x}}(t_{k-1}|t_{k-1})}^{\vec{x}(t_{k-1})} \frac{\partial^2 \vec{f}}{\partial \vec{x}^2} (\vec{x} - \hat{\vec{x}}(t_{k-1}|t_{k-1})) d\vec{x} \right)^T \right] + Q_{t_{k-1}} \quad (\text{B.8})
\end{aligned}$$

With $Q'_{t_{k-1}}$ defined as

$$\begin{aligned}
Q'_{t_{k-1}} := & E \left[\left(\left(I + F \left(\hat{\vec{x}}(t_{k-1}|t_{k-1}), t_{k-1} \right) \cdot \Delta t \right) \cdot \tilde{\vec{x}}(t_{k-1}) \right) \cdot \right. \\
& \cdot \left(\tilde{f} \left(\hat{\vec{x}}(t_{k-1}|t_{k-1}), t_{k-1} \right) \cdot \Delta t - \int_{t_{k-1}}^{t_k} \vec{f}'(\vec{x}(t), t) (t - t_{k-1}) dt - \right. \\
& \left. \left. - \frac{\partial \tilde{f}}{\partial \vec{x}} \Bigg|_{\vec{x}=\hat{\vec{x}}(t_{k-1}|t_{k-1})} \cdot \tilde{\vec{x}}(t_{k-1}) \cdot \Delta t - \Delta t \cdot \int_{\hat{\vec{x}}(t_{k-1}|t_{k-1})}^{\vec{x}(t_{k-1})} \frac{\partial^2 \tilde{f}}{\partial \vec{x}^2} \left(\vec{x} - \hat{\vec{x}}(t_{k-1}|t_{k-1}) \right) d\vec{x} \right)^T \right] + \\
& + E \left[\left(\tilde{f} \left(\hat{\vec{x}}(t_{k-1}|t_{k-1}), t_{k-1} \right) \cdot \Delta t - \int_{t_{k-1}}^{t_k} \vec{f}'(\vec{x}(t), t) (t - t_{k-1}) dt - \right. \right. \\
& \left. \left. - \frac{\partial \tilde{f}}{\partial \vec{x}} \Bigg|_{\vec{x}=\hat{\vec{x}}(t_{k-1}|t_{k-1})} \cdot \tilde{\vec{x}}(t_{k-1}) \cdot \Delta t - \Delta t \cdot \int_{\hat{\vec{x}}(t_{k-1}|t_{k-1})}^{\vec{x}(t_{k-1})} \frac{\partial^2 \tilde{f}}{\partial \vec{x}^2} \left(\vec{x} - \hat{\vec{x}}(t_{k-1}|t_{k-1}) \right) d\vec{x} \right) \cdot \right. \\
& \left. \cdot \left(\left(I + F \left(\hat{\vec{x}}(t_{k-1}|t_{k-1}), t_{k-1} \right) \cdot \Delta t \right) \cdot \tilde{\vec{x}}(t_{k-1}) \right)^T \right] + \\
& + E \left[\left(\tilde{f} \left(\hat{\vec{x}}(t_{k-1}|t_{k-1}), t_{k-1} \right) \cdot \Delta t - \int_{t_{k-1}}^{t_k} \vec{f}'(\vec{x}(t), t) (t - t_{k-1}) dt - \right. \right. \\
& \left. \left. - \frac{\partial \tilde{f}}{\partial \vec{x}} \Bigg|_{\vec{x}=\hat{\vec{x}}(t_{k-1}|t_{k-1})} \cdot \tilde{\vec{x}}(t_{k-1}) \cdot \Delta t - \Delta t \cdot \int_{\hat{\vec{x}}(t_{k-1}|t_{k-1})}^{\vec{x}(t_{k-1})} \frac{\partial^2 \tilde{f}}{\partial \vec{x}^2} \left(\vec{x} - \hat{\vec{x}}(t_{k-1}|t_{k-1}) \right) d\vec{x} \right) \cdot \right. \\
& \left. \cdot \left(\tilde{f} \left(\hat{\vec{x}}(t_{k-1}|t_{k-1}), t_{k-1} \right) \cdot \Delta t - \int_{t_{k-1}}^{t_k} \vec{f}'(\vec{x}(t), t) (t - t_{k-1}) dt - \right. \right. \\
& \left. \left. - \frac{\partial \tilde{f}}{\partial \vec{x}} \Bigg|_{\vec{x}=\hat{\vec{x}}(t_{k-1}|t_{k-1})} \cdot \tilde{\vec{x}}(t_{k-1}) \cdot \Delta t - \Delta t \cdot \int_{\hat{\vec{x}}(t_{k-1}|t_{k-1})}^{\vec{x}(t_{k-1})} \frac{\partial^2 \tilde{f}}{\partial \vec{x}^2} \left(\vec{x} - \hat{\vec{x}}(t_{k-1}|t_{k-1}) \right) d\vec{x} \right)^T \right] + \\
& + Q_{t_{k-1}} \tag{B.9}
\end{aligned}$$

an expression similar to the KF is gained.

$$P(t_k|t_{k-1}) = \left(I + F(\hat{\vec{x}}(t_{k-1}|t_{k-1}), t_{k-1}) \cdot \Delta t \right) P(t_{k-1}|t_{k-1}) \cdot \quad (\text{B.10})$$

$$\cdot \left(I + F(\hat{\vec{x}}(t_{k-1}|t_{k-1}), t_{k-1}) \cdot \Delta t \right)^T + Q'_{t_{k-1}} \quad (\text{B.11})$$

However it should be noted that $Q'_{t_{k-1}}$ is not equal to $Q_{t_{k-1}}$ which leads to an increased process noise.

References

- [1] R. Abou-Jaoude. Acc radar sensor technology, test requirements, and test solutions. *IEEE Transactions on Intelligent Transportation Systems*, 4(3):115–122, Sept. 2003.
- [2] A.T. Alouani and W.D. Blair. Use of a kinematic constraint in tracking constant speed, maneuvering targets. In *Proc. 30th IEEE Conference on Decision and Control*, pages 2055–2058, 11–13 Dec. 1991.
- [3] Y. Anxi, L. Diannong, H. Weidong, and D. Zhen. A unified out-of-sequence measurements filter. In *Proc. IEEE International Radar Conference*, pages 453–458, 2005.
- [4] M.S. Arulampalam, S. Maskell, N. Gordon, and T. Clapp. A tutorial on particle filters for online nonlinear/non-gaussian bayesian tracking. *IEEE Transactions on Signal Processing*, 50(2):174–188, 2002.
- [5] D. Avitzour and S.R. Rogers. Optimal measurement scheduling for prediction and estimation. *IEEE Transactions on Acoustics, Speech and Signal Processing*, 38(10):1733–1739, 1990.
- [6] Y. Bar-Shalom. *Multitarget-Multisensor Tracking: Applications and Advances*, volume 3 of *Multitarget-Multisensor Tracking*. Artech House, 2000.
- [7] Y. Bar-Shalom. Update with out-of-sequence measurements in tracking: exact solution. *IEEE Transactions on Aerospace and Electronic Systems*, 38(3):769–777, July 2002.
- [8] Y. Bar-Shalom and H. Chen. Imm estimator with out-of-sequence measurements. *IEEE Transactions on Aerospace and Electronic Systems*, 41(1):90–98, Jan. 2005.
- [9] Y. Bar-Shalom, H. Chen, and M. Mallick. One-step solution for the multi-step out-of-sequence-measurement problem in tracking. *IEEE Transactions on Aerospace and Electronic Systems*, 40(1):27–37, Jan. 2004.
- [10] Y. Bar-Shalom and T. Fortman. *Tracking and Data Association*. Academic Press, New York, 1988.

- [11] Y. Bar-Shalom and X.R. Li. *Estimation and Tracking: Principles, Techniques and Software*. Artech House, 1993.
- [12] Y. Bar-Shalom and X.R. Li. *Multitarget-Multisensor Tracking: Principles and Techniques*. YBS Publishing, 3. edition, 1995.
- [13] Y. Bar-Shalom, M. Mallick, H. Chen, and R. Washburn. One-step solution for the general out-of-sequence-measurement problem in tracking. In *Proc. IEEE Aerospace Conference*, volume 4, pages 1551–1559, 2002.
- [14] Y. Bar-Shalom and E. Tse. Tracking in a cluttered environment with probabilistic data association. *Automatica*, 2:451–460, September 1975.
- [15] M. Bertozzi, A. Broggi, P. Grisleri, T. Graf, and M.-M. Meinecke. Pedestrian detection in infrared images. In A. Broggi, editor, *Proc. IEEE Intelligent Vehicles Symposium*, pages 662–667, 2003.
- [16] E. Besada-Portas, J.A. Lopez-Orozco, and J.M. de la Cruz. Multisensor out-of-sequence data fusion for estimating the state of dynamic systems. In *Proc. Information, Decision and Control Conference*, pages 348–353, 2007.
- [17] S. Blackman. *Multiple Target Tracking with Radar Applications*. Norwood, MA, 1986.
- [18] S. Blackman and R. Popoli. *Design and Analysis of Modern Tracking Systems*. Artech House, 1999.
- [19] W.D. Blair and G.A. Watson. Interacting multiple bias model algorithm with application to tracking maneuvering targets. In *Proc. 31st IEEE Conference on Decision and Control*, pages 3790–3795, 16–18 Dec. 1992.
- [20] R.R. Brooks and S.S. Iyengar. *Multi-Sensor Fusion: Fundamentals and Applications*. Prentice Hall, New Jersey, 1998.
- [21] Statistisches Bundesamt. Verkehrsunfälle 2007. *Fachserie 8 Reihe 7*, November 2007.
- [22] G. Cena and A. Valenzano. On the properties of the flexible time division multiple access technique. *IEEE TRANSACTIONS ON INDUSTRIAL INFORMATICS*, 2(2):86–94, 2006.
- [23] V. Cerone, M. Milanese, and D. Regruto. Experimental results on combined automatic lane keeping and driver’s steering. In *Proc. American Control Conference*, pages 3126–3131, 9–13 July 2007.

-
- [24] S. Challa, R. Evans, and X. Wang. A bayesian solution and its approximations to out-of-sequence measurement problems. *Information Fusion*, 4:pages 185–199, 2003.
- [25] S. Challa, R. Evans, X. Wang, and J. Legg. A fixed-lag smoothing solution to out-of-sequence information fusion problems. *Communications in Information and Systems*, 2(4):325–348, Dec. 2002.
- [26] S. Challa and J. Legg. Track-to-track fusion of out-of-sequence tracks. In *Proc. Fifth International Conference on Information Fusion*, volume 2, pages 919–926, Annapolis, July 2002.
- [27] K.C. Chang, Z. Tian, S. Mori, and C.Y. Chong. Map track fusion performance evaluation. In *Proc. Fifth International Conference on Information Fusion*, volume 1, pages 512–519, Annapolis, 2002.
- [28] H. Chen, T. Kirubarajan, and Y. Bar-Shalom. Performance limits of track-to-track fusion versus centralized estimation: theory and application [sensor fusion]. *IEEE Transactions on Aerospace and Electronic Systems*, 39(2):386–400, 2003.
- [29] H. Chen, X.R. Li, and Y. Bar-Shalom. On joint track initiation and parameter estimation under measurement origin uncertainty. *IEEE Transactions on Aerospace and Electronic Systems*, 40(2):675–694, 2004.
- [30] C.-Y. Chong, S. Mori, W.H. Barker, and K.-C. Chang. Architectures and algorithms for track association and fusion. *IEEE Aerospace and Electronic Systems Magazine*, 15(1):5–13, Jan. 2000.
- [31] G. Dantzig. *Linear Programming and Extensions*. Princeton University Press, 1998.
- [32] D. Danu, A. Sinha, T. Kirubarajan, M.F. Farooq, and D. Peters. Performance evaluation of multi-platform distributed data fusion methods for multi-target tracking. In *Proc. IEEE Aerospace Conference*, pages 1–14, 2007.
- [33] J. Dezert, A. Tchamova, T. Semerdjiev, and P. Konstantinova. Performance evaluation of fusion rules for multitarget tracking in clutter based on generalized data association. In *Proc. 8th International Conference on Information Fusion*, volume 2, Philadelphia, July 2005.
- [34] E. Dincmen and T. Acarman. Application of vehicle dynamics’ active control to a realistic vehicle model. In *Proc. American Control Conference ACC ’07*, pages 200–205, 2007.

- [35] A. Discant, A. Rogozan, C. Rusu, and A. Benschair. Sensors for obstacle detection - a survey. In A. Rogozan, editor, *Proc. th International Spring Seminar on Electronics Technology*, pages 100–105, 2007.
- [36] Z. Duan, C. Han, and X. Rong Li. Comments on "unbiased converted measurements for tracking". *IEEE Transactions on Aerospace and Electronic Systems*, 40(4):1374–, 2004.
- [37] H. F. Durrant-Whyte. Sensor models and multisensor integration. *International Journal of Robotics Research*, 7(6):97–113, Dec. 1988.
- [38] W. Elmenreich. *Sensor Fusion in Time-Triggered Systems*. PhD thesis, Technical University of Vienna, Oct. 2002.
- [39] W. Elmenreich, editor. *Embedded Systems Engineering*. Vienna University of Technology, Austria, Vienna, Austria, 2009.
- [40] W. Elmenreich and S. Pitzek. The time-triggered sensor fusion model. In *Proc. 5th IEEE International Conference on Intelligent Engineering Systems*, pages 297–300, Helsinki, Sep. 2001.
- [41] W. Elmenreich and S. Pitzek. Using sensor fusion in a time-triggered network. In *Proc. 27th Annual Conference of the IEEE Industrial Electronics Society*, volume 1, pages 369–374, Nov.Dec. 2001.
- [42] T. Engel, B. Rech, and R. Schmidt. Time-triggered network-architecture for vehicle surrounding detection. In *Proc. 13. Aachener Kolloquium Fahrzeug- und Motorentechnik*, Aachen, Oct. 2004.
- [43] A. Ewald and V. Willhoeft. Laser scanners for obstacle detection in automotive applications. In V. Willhoeft, editor, *Proc. IEEE Intelligent Vehicles Symposium IV 2000*, pages 682–687, 2000.
- [44] J. Feldges. Panel discussion on introduction of intelligent vehicles into society: technical, mental and legal aspects. future driver assistance systems-product liability and driver’s responsibility. In *Proc. IEEE Intelligent Vehicles Symposium*, pages 319–322, Sept. 1996.
- [45] FlexRay Consortium. Flexray communications system. Protocol specification, FlexRay Consortium, December 2005. Version 2.1 Revision A.
- [46] F. Folster and H. Rohling. Data association and tracking for automotive radar networks. *IEEE TRANSACTIONS ON INTELLIGENT TRANSPORTATION SYSTEMS*, 6(4):370–377, Dec. 2005.
- [47] M. Fox. *Intelligent Scheduling*. Morgan Kaufmann, San Francisco, 1994.

-
- [48] U. Franke, S. Mehring, A. Suissa, and S. Hahn. The daimler-benz steering assistant: a spin-off from autonomous driving. In *Proc. IEEE Intelligent Vehicles Symposium*, pages 120–124, Oct. 1994.
- [49] K. Fuerstenberg, P. Baraud, G. Caporaletti, S. Citelli, Z. Eitan, U. Lages, and C. Lavergne. Development of a pre-crash sensorial system - the chameleon project. In *Proc. Joint VDI/VW Congress Vehicle Concepts for the 2nd Century of Automotive Technology*, Nov. 2001.
- [50] A. Gage and R.R. Murphy. Sensor scheduling in mobile robots using incomplete information via min-conflict with happiness. *IEEE Transactions on Systems, Man and Cybernetics, Part B*, 34(1):454–467, Feb. 2004.
- [51] A. Gelb, J. Kasper, R. Nash, C. Price, and A. Sutherland. *Applied Optimal Estimation*. The M.I.T. Press, 1974.
- [52] Robert Bosch GmbH. *Sicherheits- und Komfortsysteme*. Vieweg, October 2004.
- [53] I. Goodman, R. Mahler, and H. Nguyen. *MATHEMATICS OF DATA FUSION*, volume 37 of *Mathematical and Statistical Methods*. Kluwer Academic Publishers, Dordrecht / Boston / London, 1997.
- [54] J. Gunnarsson, L. Svensson, L. Danielsson, and F. Bengtsson. Tracking vehicles using radar detections. In L. Svensson, editor, *Proc. IEEE Intelligent Vehicles Symposium*, pages 296–302, 2007.
- [55] A. Hagiescu, U.D. Bordoloi, S. Chakraborty, P. Sampath, P.V.V. Ganesan, and S. Ramesh. Performance analysis of flexray-based ecu networks. In U.D. Bordoloi, editor, *Proc. 44th ACM/IEEE Design Automation Conference DAC '07*, pages 284–289, 2007.
- [56] I. Harre. A standardized algorithm for the determination of position errors by the example of gps with and without selective availability. *International Hydrographic Review*, 2(1):2001, June 2001.
- [57] F. Hartwich, B. Müller, T. Führer, and R. Hugel. Time triggered communication on CAN. In *Proc. 7th International CAN Conference*, Amsterdam, 2000.
- [58] G.M. Hoffmann, C.J. Tomlin, M. Montemerlo, and S. Thrun. Autonomous automobile trajectory tracking for off-road driving: Controller design, experimental validation and racing. In C.J. Tomlin, editor, *Proc. American Control Conference ACC '07*, pages 2296–2301, 2007.

- [59] I. Hogan and W. Manning. The use of vehicle dynamic control systems for automotive collision mitigation. In W. Manning, editor, *Proc. 3rd Institution of Engineering and Technology Conference on Automotive Electronics*, pages 1–10, 2007.
- [60] L. Hong, S. Cong, and D. Wicker. Distributed multirate interacting multiple model (dmrimm) filtering with out-of-sequence gmti data. In *Proc. Fifth International Conference on Information Fusion*, volume 2, pages 1054–1061, Annapolis, July 2002.
- [61] L. Hong, S. Cong, and D. Wicker. Multirate interacting multiple model (MRIMM) filtering with out-of-sequence GMTI data. In *IEE Proceedings on Radar, Sonar and Navigation*, volume 150, pages 333–343, Oct. 2003.
- [62] L. Hong, S. Cong, and D. Wicker. Distributed multirate interacting multiple model fusion (dmrimmf) with application to out-of-sequence gmti data. *IEEE Transactions on Automatic Control*, 49(1):102–107, Jan. 2004.
- [63] Z.J. Ill, X. DongJi, K.J. Wan, and K.Y. Bae. A study of autonomous parking for a 4-wheel driven mobile robot. In *Proc. Chinese Control Conference*, pages 179–184, 2007.
- [64] W.D. Jones. Keeping cars from crashing. *IEEE Spectrum*, 38(9):40–45, 2001.
- [65] R. Jordan, B. Lucas, M. Randler, and U. Wilhelm. Safety application specific requirements on the data processing of environmental sensors. In B. Lucas, editor, *Proc. IEEE Intelligent Vehicles Symposium*, pages 907–912, 2004.
- [66] S.J. Julier and J.K. Uhlmann. Corrections to unscented filtering and nonlinear estimation. *Proceedings of the IEEE*, 92(12):1958–1958, Dec. 2004.
- [67] S.J. Julier and J.K. Uhlmann. Unscented filtering and nonlinear estimation. *Proceedings of the IEEE*, 92(3):401–422, March 2004.
- [68] S.J. Julier and J.K. Uhlmann. Fusion of time delayed measurements with uncertain time delays. In *Proc. American Control Conference the 2005*, pages 4028–4033, 8–10 June 2005.
- [69] N. Kaempchen and K. Dietmayer. Data synchronization strategies for multi-sensor fusion. In *Proc. IEEE Conference on Intelligent Transportation Systems*, Oct. 2003.
- [70] N. Kaempchen, U. Franke, and R. Ott. Stereo vision based pose estimation of parking lots using 3d vehicle models. In *Proc. IEEE Intelligent Vehicle Symposium*, volume 2, pages 459–464, June 2002.

-
- [71] N. Kaempchen, K. Weiss, M. Schaefer, and K.C.J. Dietmayer. Imm object tracking for high dynamic driving maneuvers. In *Proc. IEEE Intelligent Vehicles Symposium*, pages 825–830, 14–17 June 2004.
- [72] L.M. Kaplan, W.D. Blair, and Y. Bar-Shalom. Simulations studies of multi-sensor track association and fusion methods. In *Proc. IEEE Aerospace Conference*, 2006.
- [73] J. Kaszubiak, M. Tornow, R.W. Kuhn, and B. Michaelis. Real-time, 3-d-multi object position estimation and tracking. In *Proc. 17th International Conference on Pattern Recognition*, 2004.
- [74] J. Kibbel, W. Justus, and K. Furstenberg. Lane estimation and departure warning using multilayer laserscanner. In *Proc. IEEE Intelligent Transportation Systems*, pages 607–611, 13–15 Sept. 2005.
- [75] H. Kopetz. Event-triggered versus time-triggered real-time systems. Research Report 8/1991, Technische Universität Wien, Institut für Technische Informatik, Treitlstr. 1-3/182-1, 1040 Vienna, Austria, 1991.
- [76] H. Kopetz. *Real-Time Systems. Design Principles for Distributed Embedded Applications*. Kluwer Academic Publisher, 1997.
- [77] H. Kopetz. The time-triggered model of computation. In *Proc. 19th IEEE Real-Time Systems Symposium*, pages 168–177, 1998.
- [78] H. Kopetz. Fault containment and error detection in the time-triggered architecture. In *Proc. Sixth International Symposium on Autonomous Decentralized Systems*, pages 139–146, 2003.
- [79] H. Kopetz and G. Bauer. The time-triggered architecture. *Proc. IEEE Special Issue on Modeling and Design of Embedded Software*, 91:112–126, January 2003.
- [80] H. Kopetz et al. Specification of the TTP/A protocol. Research Report 61/2002, Vienna, Austria, Sep. 2002. Version 2.00.
- [81] M. Koplin and W. Elmerreich. Analysis of kalman filter based approaches for fusing out-of-sequence measurements corrupted by systematic errors. In *Proc. IEEE International Conference on Multisensor Fusion and Integration for Intelligent Systems*, August 2008.
- [82] R. Kotake, K. Watanabe, and K. Kobayashi. Vision-based two-phase strategy for autonomous lane navigation. In *Proc. International Joint Conference SICE-ICASE*, pages 2808–2811, Oct. 2006.

- [83] C.M. Kreucher and A. O. Hero. Monte carlo methods for sensor management in target tracking. In *Proc. IEEE Nonlinear Statistical Signal Processing Workshop*, pages 232–237, Sept. 2006.
- [84] R. Labayrade, C. Royere, and D. Aubert. A collision mitigation system using laser scanner and stereovision fusion and its assessment. In *Proc. IEEE Intelligent Vehicles Symposium*, pages 441–446, 6–8 June 2005.
- [85] R. Labayrade, C. Royere, and D. Aubert. Experimental assessment of the rescue collision-mitigation system. *IEEE Transactions on Vehicular Technology*, 56(1):89–102, Jan. 2007.
- [86] R.P. Lang and D.B. Freitag. Programmable digital vehicle control system. *IEEE Transactions on Vehicular Technology*, 28(1):80–87, Feb. 1979.
- [87] T.D. Larsen, N.A. Andersen, O. Ravn, and N.K. Poulsen. Incorporation of time delayed measurements in a discrete-time kalman filter. In *Proc. 37th IEEE Conference on Decision and Control*, volume 4, pages 3972–3977, Dec. 1998.
- [88] S.L. Lauritzen. Propagation of probabilities, means and variances in mixed graphical models. *Journal of the American Statistical Association*, 87(420):1098–1108., 1992.
- [89] J.-P. Le Cadre. Scheduling active and passive measurements [tracking systems]. In *Proc. Third International Conference on Information Fusion*, volume 2, pages 22–29, Paris, July 2000.
- [90] R. Leland. An alternate calculation of the discrete-time kalman filter gain and riccati equation solution. *IEEE Transactions on Automatic Control*, 41(12):1817–1819, 1996.
- [91] J. Leohold and C. Schmidt. Communication requirements of future driver assistance systems in automobiles. In *Proc. IEEE International Workshop on Factory Communication Systems*, pages 167–174, Sept. 2004.
- [92] Y. Li, L.W. Krakow, E.K.P. Chong, and K.N. Groom. Dynamic sensor management for multisensor multitarget tracking. In *Proc. 40th Annual Conference on Information Sciences and Systems*, pages 1397–1402, March 2006.
- [93] Y. Liu, We. Wu, and J. Lin. Research of information fusion evaluation methodology based on dominance relation. In *Proc. Eighth ACIS International Conference on Software Engineering, Artificial Intelligence, Networking, and Parallel/Distributed Computing*, volume 1, pages 788–793, 2007.

-
- [94] X. Lu, H.S. Zhang, W. Wang, and K.L. Teo. Kalman filtering for multiple time-delay systems. *AUTOMATICA*, 41(8):1455 – 1461, 2005.
- [95] M. Mallick, S. Coraluppi, and C. Carthel. Advances in asynchronous and decentralized estimation. In *Proc. IEEE Aerospace Conference*, volume 4, pages 1873–1888, March 2001.
- [96] M. Mallick, T. Kirubarajan, and S. Arulampalam. Out-of-sequence measurement processing for tracking ground target using particle filters. In *Proc. IEEE Aerospace Conference*, volume 4, pages 1809–1818, 2002.
- [97] M. Mallick, J. Krant, and Y. Bar-Shalom. Multi-sensor multi-target tracking using out-of-sequence measurements. In *Proc. Fifth International Conference on Information Fusion*, volume 1, pages 135–142, Annapolis, July 2002.
- [98] M. Mallick and A. Marrs. Comparison of the kf and particle filter based out-of-sequence measurement filtering algorithms. In *Proc. Sixth International Conference of Information Fusion*, volume 1, pages 422–429, Cairns, July 2003.
- [99] T. Marita, F. Oniga, S. Nedeveschi, and T. Graf. Calibration accuracy assessment methods for stereovision sensors used in vehicles. In *Proc. IEEE International Conference on Intelligent Computer Communication and Processing*, pages 111–118, 6–8 Sept. 2007.
- [100] T. Marita, F. Oniga, S. Nedeveschi, T. Graf, and R. Schmidt. Camera calibration method for far range stereovision sensors used in vehicles. In *Proc. IEEE Intelligent Vehicles Symposium*, pages 356–363, 13–15 June 2006.
- [101] M. Mauthner. Out-of-sequence measurements treatment in sensor fusion applications: Optimal sensor fusion speed and synchronization. In *Proc. 4th International Workshop on Intelligent Transportation*, Hamburg, March 2007.
- [102] M. Mauthner, R. Altendorfer, W. Elmenreich, and A. Kirchner. Optimization of sensor, bus, and fusion schedules of a time-triggered sensor fusion system. In *Proc. IEEE Intelligent Vehicles Symposium*, pages 570–575, Istanbul, June 2007.
- [103] M. Mauthner, W. Elmenreich, and A. Kirchner. Analysis of sensor and fusion schedules of a time-triggered sensor fusion system. In *Proc. 10th International Conference on Information Fusion*, pages 1–5, Quebec City, July 2007.
- [104] M. Mauthner, W. Elmenreich, A. Kirchner, and D. Boesel. Out-of-sequence measurements treatment in sensor fusion applications: Buffering versus advanced algorithms. In *Proc. 4. Workshop Fahrerassistenzsysteme*, Loewenstein/Hoesslinsuelz, Oct. 2006.

- [105] R. Mehra. Optimization of measurement schedules and sensor designs for linear dynamic systems. *IEEE Transactions on Automatic Control*, 21(1):55–64, 1976.
- [106] P. Milbredt. Synchronisation der messdatenaufnahme in einem zeitgesteuerten netzwerk. Master’s thesis, Technical University of Dresden, Germany, 2005.
- [107] A.I. Mourikis and S.I. Roumeliotis. Optimal sensor scheduling for resource-constrained localization of mobile robot formations. *IEEE Transactions on Robotics*, 22(5):917–931, Oct. 2006.
- [108] N. Navet, Y. Song, F. Simonot-Lion, and C. Wilwert. Trends in automotive communication systems. *Proceedings of the IEEE*, 93(6):1204–1223, 2005.
- [109] S. Nedeveschi, R. Danescu, D. Frentiu, T. Marita, F. Oniga, C. Pocol, R. Schmidt, and T. Graf. High accuracy stereo vision system for far distance obstacle detection. In *Proc. IEEE Intelligent Vehicles Symposium*, pages 292–297, 14–17 June 2004.
- [110] E.W. Nettleton and H. Durrant-Whyte. Delayed and asequent data in decentralized sensing networks. In *Proc. of SPIE Conference on Sensor Fusion and Decentralized Control in Robotic Systems IV*, volume 4571, pages 1–9, 2001.
- [111] R. Obermaisser. Can emulation in a time-triggered environment. In *Proc. IEEE International Symposium on Industrial Electronics*, volume 1, pages 270–275 vol.1, 2002.
- [112] M. Orton and A. Marrs. A bayesian approach to multi-target tracking and data fusion with out-of-sequence measurements. In *Proc. IEE Target Tracking: Algorithms and Applications*, volume 1, pages 15/1–15/5, Oct. 2001.
- [113] M. Orton and A. Marrs. Incorporation of out-of-sequence measurements in non-linear dynamic systems using particle filters. Technical report, University of Cambridge: Department of Engineering, 2001.
- [114] M. Orton and A. Marrs. Particle filters for tracking with out-of-sequence measurements. *IEEE Transactions on Aerospace and Electronic Systems*, 41(2):693–702, April 2005.
- [115] D. Reid. An algorithm for tracking multiple targets. *IEEE Transactions on Automatic Control*, 24(6):843–854, Dec. 1979.
- [116] H. Rohling and M.-M. Meinecke. Waveform design principles for automotive radar systems. In *Proc. CIE International Conference on Radar*, pages 1–4, 2001.

-
- [117] X. Rong Li and V.P. Jilkov. Survey of maneuvering target tracking. part iii: Measurement models. In *Proc. SPIE Conference on Signal and Data Processing of Small Targets*, July 2001.
- [118] X. Rong Li and V.P. Jilkov. Survey of maneuvering target tracking. part i: Dynamic models. *IEEE Transactions on Aerospace and Electronic Systems*, 39(4):1333–1364, Oct. 2003.
- [119] S. Schemmer and E. Nett. Achieving reliable and timely task execution in mobile embedded applications. In *Proc. Ninth IEEE International Workshop on Object-Oriented Real-Time Dependable Systems*, Oct. 2003.
- [120] D. Schrage and P.G. Gonsalves. Sensor scheduling using ant colony optimization. In *Proc. Sixth International Conference of Information Fusion*, volume 1, pages 379–385, Cairns, 2003.
- [121] J.C. Spall. Improved methods for monte carlo estimation of the fisher information matrix. In *Proc. American Control Conference*, pages 2395–2400, 11–13 June 2008.
- [122] A. Stevens and A. Parkes. Advisors - a strategic approach to adas deployment. In *Proc. International Conference on Advanced Driver Assistance Systems*, pages 1–5, Sept. 2001.
- [123] A. Stevens and D. Strang. Maximising usability and minimising liability - the response project. In *Proc. International Conference on Advanced Driver Assistance Systems*, pages 11–14, Sept. 2001.
- [124] D. Stromberg, M. Andersson, and F. Lantz. On platform-based sensor management. In *Proc. Fifth International Conference on Information Fusion*, volume 1, pages 600–607, Annapolis, July 2002.
- [125] D. Stüker. *Heterogene Sensordatenfusion zur robusten Objektverfolgung im automobilen Straßenverkehr*. PhD thesis, Carl von Ossietzky-Universität Oldenburg, September 2004.
- [126] Z. Sun, G. Bebis, and R. Miller. On-road vehicle detection: a review. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 28(5):694–711, May 2006.
- [127] S. Suranthiran and S. Jayasuriya. Optimal fusion of multiple nonlinear sensor data. *IEEE Sensors Journal*, 4(5):651–663, Oct. 2004.
- [128] D. K. Tasoulis, N. M. Adams, and D. J. Hand. Should delayed measurements always be incorporated in filtering? In *Proc. 15th International Conference on Digital Signal Processing*, pages 264–267, 2007.

- [129] S.C.A. Thomopoulos and L. Zhang. Distributed filtering with random sampling and delay. In *Proc. 27th IEEE Conference on Decision and Control*, pages 2348–2353, Dec. 1988.
- [130] S. Thrun. Winning the darpa grand challenge: A robot race through the mo-jave desert. In *Proc. 21st IEEE/ACM International Conference on Automated Software Engineering ASE '06*, pages 11–11, 18–22 Sept. 2006.
- [131] TTAGroup. *Specification of the TTP/C Protocol*. TTAGroup, 2003. Available at www.ttagroup.org.
- [132] M. Tucker, A. Heenan, and A. Buchanan. Real time embedded sensor fusion for driver assistance. In A. Heenan, editor, *Proc. IEEE Intelligent Transportation Systems*, pages 596–601, 2005.
- [133] J.K. Uhlmann. Covariance consistency methods for fault-tolerant distributed data fusion. *Information Fusion*, 4(3):201–215, Sept. 2003.
- [134] T. Vaa, M. Penttinen, and I. Spyropoulou. Intelligent transport systems and effects on road traffic accidents: state of the art. In *Proc. 13th World Congress on Intelligent Transport Systems and Services*, volume 1, pages 81–88, June 2007.
- [135] G. van Keuk and S.S. Blackman. On phased-array radar tracking and parameter control. *IEEE Transactions on Aerospace and Electronic Systems*, 29(1):186–194, 1993.
- [136] W. van Norden, J. de Jong, F. Bolderheij, and L. Rothkrantz. Intelligent task scheduling in sensor networks. In *Proc. 8th International Conference on Information Fusion*, volume 2, Philadelphia, July 2005.
- [137] H. Van Trees. *Detection, Estimation and Modulation Theory*. Wiley, New York, 1968.
- [138] K.A.P.C. van Wees. Vehicle safety regulations and adas: tensions between law and technology. In *Proc. IEEE International Conference on Systems, Man and Cybernetics*, volume 4, pages 4011–4016, Oct. 2004.
- [139] P.K. Varshney. Multisensor data fusion. *Electronics & Communication Engineering Journal*, 9(6):245–253, Dec. 1997.
- [140] X. Wang and S. Challa. Augmented state imm-pda for oosm solution to maneuvering target tracking in clutter. In *Proc. International Radar Conference*, pages 479–485, Sept. 2003.

- [141] K. Weiss. *Interpretation von Fahrumgebungen für Fahrerassistenzsysteme*. PhD thesis, University of Rostock, 2007.
- [142] K. Weiss, H. Philipps, T.-B. To, and A. Kirchner. Environmental perception and situation assessment for an advanced highway assistant. In *Proc. IEEE Intelligent Vehicles Symposium*, pages 472–477, 2005.
- [143] C. Wen, X. Tang, and Q. Ge. The recursive fusion algorithm with out-of-sequence measurements for sensor networks. In *Proc. 3rd International Workshop on Signal Design and Its Applications in Communications*, pages 368–372, 2007.
- [144] Wikipedia, the free Encyclopedia. Sensor fusion. Wikimedia Foundation, May 11, 2009. http://en.wikipedia.org/wiki/Sensor_fusion.
- [145] K. Zhang and X. Rong Li. Optimal update with out-of-sequence measurements for distributed filtering. In *Proc. Fifth International Conference on Information Fusion*, volume 2, pages 1519–1526, Annapolis, July 2002.
- [146] K. Zhang, X. Rong Li, and H. Chen. Multi-sensor multi-target tracking with out-of-sequence measurements. In *Proc. Sixth International Conference of Information Fusion*, volume 1, pages 672–679, Cairns, 2003.
- [147] K. Zhang, X. Rong Li, and Y. Zhu. Optimal update with out-of-sequence measurements. *IEEE Transactions on Signal Processing*, 53(6):1992–2004, June 2005.
- [148] X.B. Zhang, C.Y. Qin, and R. Gao. The research on multisensor information fusion with out-of-sequence measurements. *Chinese Journal of sensors and actuators*, 19(4):1310–1312, 2006.
- [149] Z. Zhao and X. Rong Li. Two classes of relative measures of estimation performance. In *Proc. 10th International Conference on Information Fusion*, pages 1–7, Quebec City, July 2007.

List of Publications

M. Koplin (née Mauthner) and W. Elmenreich. Analysis of kalman filter based approaches for fusing out-of-sequence measurements corrupted by systematic errors. In *Proc. IEEE International Conference on Multisensor Fusion and Integration for Intelligent Systems*, August 2008.

M. Mauthner, W. Elmenreich, and A. Kirchner. Analysis of sensor and fusion schedules of a time-triggered sensor fusion system. In *Proc. 10th International Conference on Information Fusion*, pages 1–5, Quebec City, July 2007.

M. Mauthner, R. Altendorfer, W. Elmenreich, and A. Kirchner. Optimization of sensor, bus, and fusion schedules of a time-triggered sensor fusion system. In *Proc. IEEE Intelligent Vehicles Symposium*, pages 570–575, Istanbul, June 2007.

M. Mauthner. Out-of-sequence measurements treatment in sensor fusion applications: Optimal sensor fusion speed and synchronization. In *Proc. 4th International Workshop on Intelligent Transportation*, Hamburg, March 2007.

M. Mauthner, W. Elmenreich, A. Kirchner, and D. Boesel. Out-of-sequence measurements treatment in sensor fusion applications: Buffering versus advanced algorithms. In *Proc. 4. Workshop Fahrerassistenzsysteme*, Loewenstein/Hoesslinsuelz, Oct. 2006.

K. Weiss, H. Philipps, and M. Mauthner. Gütekriterien und Testprozeduren für die Sensordatenfusion für Fahrerassistenzsysteme. In *Konferenzband: Elektronik im Kraftfahrzeug*, Baden-Baden, Oct. 2005.

Curriculum Vitae

86/87 – 89/90	Volksschule Zeitlarn
90/91 – 98/99	Albrecht-Altendorfer-Gymnasium Regensburg
18.08.95 – 15.11.95	Forrest Lawn Highschool, Calgary, Kanada
06.01.97 – 10.04.97	Lycée Jean Macé, Niort, Frankreich
20.10.99 – 09.10.00	Studium der Physik Technische Universität München
10.10.00 – 30.09.04	Studium der Physik Universität Regensburg
11.08.03 – 31.07.04	Siemens VDO Automotive AG, Regensburg: Diplomand (Thema: Fahrzustandsanalyse mit Hilfe von Sensor- datenfusion)
08.11.04 – 07.11.07	Volkswagen AG, Wolfsburg: Doktorand (Thema: Time- Triggered Object Tracking for Advanced Driver Assis- tance Systems)
seit 01.04.05	Studium der Wirtschaftswissenschaften (BWL) FernUniversität Hagen
seit 21.04.05	Doktorstudium der technischen Informatik Technische Universität Wien
01.11.07 – 31.11.2008	Patentanwaltskanzlei Grünecker Kinkeldey Stockmair & Schwanhäusser, München: Patentingenieur
seit 01.12.08	Advanced Realtime Tracking GmbH, Weilheim i. Ob.: Softwareentwickler