SINADRA: Situation-Aware Dynamic Risk Assessment for Safe and Permissive Autonomous Systems

Thesis approved by
the Department of Computer Science
RPTU University Kaiserslautern-Landau
for the award of the Doctoral Degree

Doctor of Engineering (Dr.-Ing.)

to

Jan Reich, M.Sc.

Fraunhofer Institute for Experimental Software Engineering
(Fraunhofer IESE)

Reviewers:	Prof. Dr. Dr. h.c. Karsten Berns Prof. Dr. habil. Mario Trapp
Dean:	Prof. Dr. Christoph Garth
Date of Defense:	27th March 2026

DE-386

Copyright and licence notice

© 2026 Jan Reich

Except where otherwise indicated, this dissertation is licensed under the Creative Commons Attribution–NonCommercial–NoDerivatives 4.0 International License (CC BY-NC-ND 4.0): https://creativecommons.org/licenses/by-nc-nd/4.0/

This licence does not apply to third-party material or previously published material identified by a separate source, copyright, licence, or permission notice. In particular, figures and tables carrying individual credit lines remain subject to the terms stated in those credit lines. Uses beyond those terms may require permission from the respective rights holder.

"The art of life lies in a constant readjustment to our surroundings."

Kakuzō Okakura
The Book of Tea, 1906

Abstract

Runtime safety monitors are a key safety measure for autonomous systems, but they are often designed for worst-case behavior of other agents and the environment. This keeps safety arguments simple, yet it can make systems overly conservative, with unnecessary stops, large distances, and reduced throughput. The core problem addressed in this dissertation is how to increase monitor permissiveness without weakening safety justification: when a monitor relaxes an environment assumption, engineers must be able to argue why this relaxation is safe in the relevant hazardous situations and within the stated operational design domain.

This dissertation proposes **Situation-Aware Dynamic Risk Assessment (SINADRA)**, an engineering method that extends hazard analysis and risk assessment (HARA) work products so that they support *monitorable low-risk situation hypotheses* for adaptive constraint relaxation. SINADRA targets risk variability that stems from external agent behavior like human workers in warehouses or human drivers in traffic. The method has three main elements. First, **Structured Environment Models (SEM)** scope each hazardous situation into an explicit, risk-driven environment view, including initiator scenarios and environment factors that matter for risk. Second, **Critical Behavior Cause Trees (CBCT)** refine hazardous situations into behavior risk scenarios and relate these scenarios to *risk implausibility hypotheses* – monitorable situation conditions indicating low risk – and other safety measures from the safety concept like system behavior restrictions. CBCTs form a Boolean model that supports minimal cut set analysis as reviewable verification evidence that, within the modeled scope, risk scenarios are sufficiently blocked by hypotheses and safety measures. Third, SINADRA provides a **runtime model transformation** that turns argued hypotheses into executable inference components, using either Boolean logic or Bayesian networks (BNs).

SINADRA is evaluated in two case studies. A warehouse robotics study executes the full design-time chain, integrates Boolean runtime models into a ROS-based safety monitoring system, and demonstrates hypothesis-driven mode switching with worst-case as the default. An analytic comparison reports estimated corridor travel-time gains of about 2% for passing and 21% for overtaking under stated assumptions. An urban automated-driving study demonstrates component-level runtime feasibility in CARLA by inferring behavior intent likelihoods with BNs and computing an intent-weighted collision-risk signal that differs from a worst-case baseline in uncritical situations. Overall, SINADRA provides a traceable path from HARA artifacts to runtime adaptation logic for safer, situation-dependent permissiveness.

Acknowledgements

This dissertation started in 2018, but the idea behind it grew out of work that was already taking place around me at Fraunhofer IESE. At that time, my colleague Patrik Feth was working on dynamic risk assessment in his doctoral thesis. His work made me think about one aspect that I felt deserved more attention: a detailed understanding of the environment in which a system operates. As a human driver, I was fascinated by how naturally we adapt our behavior to a situation, interpret small cues, form expectations about others, and adjust our actions accordingly. This raised a simple question for me: if humans can act in such a situation-aware manner, why should we not try to teach machines to do the same? I started studying concepts of human decision making from cognitive science and tried to connect these ideas with established concepts from system safety engineering. Looking back, one of the most important lessons of this PhD is that having an idea for solving a problem is the easy part. Showing in a scientifically sound way that the proposed solution actually addresses the problem is much harder.

First and foremost, I would like to thank Mario Trapp, who was my operative supervisor throughout the entire PhD. From 2018 until submission, we met almost every month. I am very grateful for this continuity and for his exceptional ability as an academic supervisor. Mario was an invaluable sparring partner who challenged my ideas, asked the right questions, and gave direction without simply presenting a solution. I came to appreciate that this is an important part of PhD supervision: allowing you to struggle with difficult questions while helping you find your own answers. His support also went beyond scientific discussions and included the less visible parts of PhD life, such as managing stress and difficult phases. Thank you, Mario, for your time, patience, and trust over all these years.

I would also like to thank Karsten Berns, my supervising professor at RPTU Kaiserslautern-Landau. In the later stages of the thesis, he supported me with formal matters and the submission process. His extensive experience in robotics also provided an important second perspective on my work. His questions pushed me to think about SINADRA not only as a safety engineering method, but also from the practical point of view of autonomous robotic systems, which I believe improved the final approach.

At Fraunhofer IESE, many colleagues influenced this work. I am particularly grateful to Rasmus Adler, with whom I discussed many of the concepts that became part of SINADRA. I especially value his expertise in risk modeling and the importance he places on industry standards as a basis for new safety

engineering methods. I would also like to thank Daniel Schneider, who was my department head until 2024. I greatly appreciated the freedom he gave me to develop and test ideas in public research projects such as DEIS, SESAME, and V&V Methods, as well as in industry projects with Hitachi and Intel Labs.

A special thank you goes to the students and student assistants who contributed to this work over the years. Their project, Bachelor, and Master theses helped me test ideas, reflect on them from different angles, and turn parts of SINADRA into working prototypes. In particular, I would like to thank Daniel Seifert, Andreas Betz, Sebastian Fuchs, Daniel Hillen, Michael Wittemaier, Marc Lorenz, Corey Kreiniker, and Sophie Kadel. I am also very grateful to Sonnhild Namingha for her excellent proofreading of this dissertation.

This thesis also benefited greatly from long-term collaboration with industry. I am especially thankful to Hitachi Ltd. in Japan. Our close collaboration from 2021 until today gave me the opportunity to further develop SINADRA and evaluate it in different industrial settings, from warehouse robotics to autonomous vehicles. I would particularly like to thank Satoshi Otsuka and Takehito Ogata for their valuable industry insights and continued support. I am equally grateful to Fabian Oboril and Kay-Ulrich Scholl from Intel Labs. In 2020, they gave me the opportunity to implement a SINADRA library for the CARLA simulator and evaluate an early version of the runtime mechanism. Looking back, their feedback had a significant influence on my later decision to use Bayesian networks as a runtime formalism.

During the challenging final writing phase, I greatly appreciated the company of fellow writers in the collaborative writing rooms of *Coachingzonen Wissenschaft* and *The Writing Academic*. Knowing that others were working through similar phases at the same time made many long writing sessions much easier.

Finally, I would like to thank my family. Without the continued support of my parents, pursuing a PhD would not have been possible. My father sparked my interest in science and the desire to understand why things are the way they are. My mother always had an open ear, especially when work became stressful. I would also like to thank my life partner, Ines, for her patience, encouragement, and support throughout the long eight years this project took to be completed.

Table of Contents

1 Introduction 1
1.1 Motivation 1
1.2 Research Problem 2
1.3 Solution 3
1.4 Thesis Contributions 5
1.5 Significance 6
1.6 Thesis Structure 7

2 State of the Art 9
2.1 Background 9
2.1.1 System safety terminology in this thesis 10
2.1.2 Runtime safety monitoring, safety constraints, dynamic risk assessment 11
2.1.3 Assumptions, situation hypotheses, SINADRA artifacts 13
2.2 State of the Practice: Criteria for justified adaptive runtime monitors 14
2.3 Assumption adaptation based on data 16
2.3.1 From naturalistic data to RSS parameter values 17
2.3.2 Naturalistic data ranges for IEEE 2846 assumptions 18
2.3.3 Simulation-supported calibration 18
2.3.4 Weaknesses of data-based approaches 19
2.4 Assumption adaptation with motion-based situation hypotheses 20
2.4.1 Safety constraints with dynamic object hypotheses: Risk-Aware RSS and RSS+ 20
2.4.2 Formal rule composition with environmental assumptions: goal-aware, compositional RSS 21
2.4.3 Prediction-centered behavior planning with predictive risk maps: Honda risk-based framework 22
2.4.4 Set-based reachability runtime verification with dynamic constraints 23
2.4.5 Weaknesses of motion-based approaches 24
2.5 Assumption adaptation with risk-based situation hypotheses 25
2.5.1 Dynamic risk management 26
2.5.2 Runtime representation of hazardous events 27
2.5.3 Tree-based HARA refinement for variability 29
2.5.4 Safety supervisor 29
2.5.5 ASIL-inspired dynamic tactical safety 30
2.5.6 µODDs: Parameter classes per observable situation subset 30
2.5.7 Precautionary safety and the quantitative risk norm 31
2.5.8 Weaknesses of risk-based approaches 32

2.6 Research Gap and Research Question 33
2.7 Design Science Research Process 35

3 Solution Overview 37
3.1 SINADRA Solution Idea at Runtime 37
3.2 SINADRA Engineering Method 40
3.2.1 From HARA to CBCT: refining hazardous situations 42
3.2.2 Achieving environment coverage in CBCTs with SEM 43
3.2.3 Runtime model transformation: from RIH to monitor logic. 44
3.2.4 Safety justification chain and criteria coverage 45
3.3 Example: Pedestrian Crosswalk 46

4 Method 49
4.1 Structured Environment Modeling (SEM) 49
4.1.1 From HARA to risk-driven environment models 49
4.1.2 Engineering method 51
4.2 Critical Behavior Cause Trees (CBCT) 59
4.2.1 Concept and formal model definition 60
4.2.2 Engineering method 65
4.3 Runtime Model Transformation 77
4.3.1 SINADRA runtime concept 78
4.3.2 From risk implausibility hypotheses to observable features 79
4.3.3 From features to runtime inference 80
4.3.4 From runtime model to parameter value adaptation 85
4.4 Prototypical tool support 86
4.4.1 CBCT modeling in safeTbox 89
4.4.2 Modeling Bayesian networks in GeNIe and runtime component generation 92
4.4.3 Usage of prototype tool in this thesis 94
4.5 Chapter Summary 95

5 Application and Evaluation 97
5.1 Case Study 1: Co-located human-robot operation in warehouses 101
5.1.1 Structured environment modeling 103
5.1.2 Critical behavior cause tree modeling 106
5.1.3 Runtime model transformation 113
5.1.4 Simulative feasibility demonstration 114
5.1.5 Analytic permissiveness evaluation 116
5.1.6 Evidence assessment and limitations 120
5.2 Case Study 2: Urban ADS Driving 121
5.2.1 Case study setup and produced artifacts 121
5.2.2 Situation class scoping and actor role classification 124
5.2.3 Runtime model operationalization as Bayesian networks . 126
5.2.4 Motion models and trajectory distributions 130
5.2.5 Risk computation and intent-weighted aggregation 131
5.2.6 Runtime feasibility demonstration 132

5.2.7 Evidence assessment and limitations 137
5.3 Chapter Summary 138

6 Discussion and Conclusion 141
6.1 Key Findings 141
6.2 Threats to Validity 141
6.3 Methodological Limitations 142
6.4 Contributions 145
6.5 Impact 146
6.6 Future Research Directions 147
6.7 Closing Remarks 148

Appendix A Supplementary Method Material 151
A.1 Example Initiator Scenarios 151
A.2 SINADRA Capability Deviation Analysis Guidewords 152

Appendix B Supplementary Toolchain Material 155

Appendix C Supplementary Case Study Material 159
C.1 Warehouse Case Study 159
C.2 ADS Case Study 162

References 165

List of Figures

Figure 2.1 Illustration of the working principles of Risk-Aware RSS (top) and RSS⁺ (bottom). Top: Reprinted, with permission, from Fig. 8 in (Oboril & Scholl, 2020). © 2020 IEEE. Bottom: Reprinted, with permission, from Fig. 3 in (Oboril & Scholl, 2021). © 2021 IEEE. 21

Figure 2.2 Overview of the situation-based risk evaluation and behavior planning framework. Reproduced from Fig. 1 in (Eggert, Klingelschmitt, & Damerow, 2015), licensed under CC BY 4.0. ... 22

Figure 2.3 Set-based reachability runtime verification framework. Reproduced from Fig. 6 in (Pek et al., 2020), with permission from Springer Nature. © Springer Nature 2020. 24

Figure 2.4 Dynamic Risk Management – conceptual overview. 26

Figure 2.5 Example hazardous situation refinement. Composite adapted from Table 2, Fig. 4, and Table 3 in (Warg et al., 2016). Reproduced with permission from Springer Nature. © Springer International Publishing Switzerland 2016. 28

Figure 2.6 µODDs condition safety envelope definitions on specific situation subsets, where different sets of parameter values lead to the maximum permissiveness in a µODD and can be justified from a safety point of view. Reproduced from Fig. 3 in (Koopman et al., 2019) with permission of the authors. 31

Figure 2.7 Design Science Research Process mapped to the chapters of this dissertation. Adapted from (Peffers et al., 2006); chapter mapping added by the author. Reprinted by permission of Informa UK Limited, trading as Taylor & Francis Group. 36

Figure 3.1 Situation-Aware Dynamic Risk Assessment (SINADRA) solution idea at runtime. 38

Figure 3.2 Situation-Aware Dynamic Risk Assessment (SINADRA) engineering method overview..... 41

Figure 4.1 The Structured Environment Modeling (SEM) concept enables a detailed analysis of the operational environment relevant to risks originating from the behavior of other traffic participants. The model uses initiator sources, initiator scenarios, and agent capabilities as conceptual elements to express potential causes of critical behavior relevant to assumptions the ego system makes regarding that behavior. 50

Figure 4.2 Top: Six-layer model for structuring the operational environment, reproduced from Table 1 in (Scholtes et al., 2021) under CC BY 4.0. Bottom: Selected A.U.T.O. class hierarchies from Layers 1

and 4, displayed in Protégé; author-created screenshot based on A.U.T.O. v2.0.0, © 2022 Lukas Westhofen, MIT License. 55
Figure 4.3 Agent behavior model for structuring the agent capability analysis step..... 56
Figure 4.4 Agent capability model for initiator scenario “Driver gives way to an approaching emergency vehicle from the rear”. 57
Figure 4.5 Conceptual traceability between HARA context and the structured environment model elements..... 58
Figure 4.6 Critical Behavior Cause Trees represent behavior risk scenarios together with behavior risk measures, so that reviewers can check whether the modeled risk paths are blocked by at least one measure within the declared scope. 60
Figure 4.7 Typed elements representing the causal structure of behavior risk scenarios (left) and behavior risk measures in CBCTs. 61
Figure 4.8 Conceptual traceability between behavior risk scenarios and behavior risk measures as the essential contents of critical behavior cause trees and the Structured Environment Model concretizing the elements of operational situations in the HARA. 63
Figure 4.9 Illustration of modeling elements of Critical Behavior Cause Trees (CBCT). 65
Figure 4.10 Example capability model for initiator scenario “Driver reacts to pedestrian entering the crosswalk”. 66
Figure 4.11 CBCT modular top-level structure derived from SEM. 67
Figure 4.12 CBCT structure derived from the SEM in the running example. 68
Figure 4.13 Structure of an initiator scenario CBCT. The causal structure of capability deviations consists of agent deficiencies combined with situation deviation triggers that cause deviations to occur or propagate. To identify deviations and deficiencies, HAZOP-inspired guide words are used as guidance. To identify situation deviation triggers, an operational environment ontology is used as guidance..... 69
Figure 4.14 Adding risk implausibility hypotheses (RIH) and other safety measures to the CBCT. 73
Figure 4.15 SINADRA runtime concept: Argued risk implausibility hypotheses are concretized to ego-observable features and transformed into a runtime inference model, which infers the hypothesis validity based on perceived features and switches a runtime safety monitor’s environment parameter to a pre-defined permissiveness-increasing value. 78
Figure 4.16 SINADRA runtime model as a Bayesian network. Top: evidence of a clear crosswalk, leading to a high probability that RIHcw-clear is valid. Bottom: evidence of a critical pedestrian situation, leading to a low validity probability and a high probability that

the hypothesis is invalid. Figure created using GeNIe modeler, a product of BayesFusion, LLC. 83
Figure 4.17 SINADRA Toolchain Prototype Architecture. Blue nodes are artifacts (models, files, software code), green nodes are scripts, algorithms or tool features. Abbreviations: EDDI = Executable Digital Dependability Identity, EGL = Epsilon Generation Language, ROS = Robot Operating System, GSN = Goal Structuring Notation, V&V = Verification & Validation. 88
Figure 4.18 CBCT modeling toolbox and example model in safeTbox. The basic elements of the component fault tree modeling features in safeTbox are adapted to CBCT semantics by using typed basic events. 89
Figure 4.19 Traceability between safety concept measures and GSN safety concepts in safeTbox. 90
Figure 4.20 Example output excerpt of the automated EDDI Export of the example CBCT from safeTbox into an ODE::FailureLogic Package. 92
Figure 4.21 ODE::BayesianNetwork metamodel in the Open Dependability Exchange (ODE) format 93
Figure 4.22 Intermediate Transformation Results of BN Runtime Code Generation Pipeline. 94
Figure 5.1 Context of the warehouse use case. Reproduced from Fig. 1 in (Reich et al., 2022) with permission from Springer Nature. © 2022, under exclusive licence to Springer Nature Switzerland AG. 101
Figure 5.2 Exemplary cross-role task decomposition for warehouse operation. AMR = Autonomous Mobile Robot, AFL = Autonomous Forklift, Worker = Human worker. 105
Figure 5.3 Top-Level Decomposition of Human Worker CBCT with the output deviation mode “ODM_HS1: Unsafe distance HW-AMR in narrow corridor”. The hazardous situation occurs if a critical human worker behavior occurs in either of two initiator scenarios. The realization of the SubCBCT *IS-PASS* is located in Figure 5.4, the realization of SubCBCT *IS-OVTK* is located in Figure C.1 in Appendix C.1. 107
Figure 5.4 Realization of SubCBCT *IS-PASS* consisting of the output deviation mode (ODM_IS-PASS), capability SubCBCTs, capability deviations (CD_x), scenario presence indicators (PI_x), situation deviation triggers (SDT_x), risk implausibility hypotheses (RIH_x), safety concept measures (M_x). Agent deficiencies (AD_x) are not displayed, because they are part of the realization of the capability SubCBCTs in Figure 5.5. 109
Figure 5.5 Realization of four SubCBCTs *Focus, Perceive, Understand, Act* of initiator scenario CBCT *IS-PASS*. 110

Figure 5.6 Differences in the realization of initiator SubCBCT *IS-OVTK* and its *Focus* and *Perceive* capability SubCBCTs. 111

Figure 5.7 Boolean runtime models to operationalize SINADRA risk implausibility hypotheses related to the human worker in corridor passing (left) and overtaking (right) scenarios. The models define a conservative risk area width w_C and a reduced width w_0. The risk area width is switched based on the risk implausibility hypothesis validity dependent on the presence of situation features. Figure adapted from (Reich et al., 2022), with permission from Springer Nature. © 2022 The Author(s), under exclusive licence to Springer Nature Switzerland AG. 113

Figure 5.8 Integration architecture of the SINADRA runtime software component into the Symbiotic Safety System Architecture. Adapted from Fig. 5 in (Reich et al., 2022), with permission from Springer Nature. © 2022, under exclusive licence to Springer Nature Switzerland AG. Change: “Dynamic Risk / Capability Area Prediction” was renamed to “SINADRA Dynamic Risk Area Prediction.” 115

Figure 5.9 SINADRA runtime feasibility demonstration for passing (top) and overtaking (bottom) scenarios. The scenes on the left show the situations where the risk implausibility hypotheses are not confirmed leading to the worst-case size of the risk area around the human worker. In consequence, the AMR needs to stop until the human worker has passed (passing scenario) and needs to stay behind the human worker operating at a lower speed (overtaking scenario). The scenes on the right show the situations where the risk implausibility hypotheses are confirmed based on the SINADRA runtime inference. In consequence, the risk area width is decreased in both scenarios leading to more efficient passing and overtaking. 116

Figure 5.10 Passing (left) and overtaking (right) scenarios evaluated in the case study. Adapted from Fig. 6 in (Reich et al., 2022), with permission from Springer Nature. © 2022 The Author(s), under exclusive licence to Springer Nature Switzerland AG. 117

Figure 5.11 SINADRA demonstrator architecture, showing design-time inputs and the runtime chain from ego localization and actor-role detection through intent inference and trajectory generation to ego-actor risk estimation. Adapted from Fig. 1 in (Reich et al., 2021), with permission from Springer Nature. © 2021 Springer Nature Switzerland AG. 122

Figure 5.12 Component structure of the CARLA-coupled SINADRA Python client. Originally published in the SINADRA software repository © 2021 Jan Reich. 123

Figure 5.13 Situation class decomposition of the operational design domain, exemplified for the CARLA Town03 map. The ODD is decomposed

into situation classes to scope relevant actor roles, for which critical behavior intents driving the risk of hazardous situations can be specified in a concrete situational context. Adapted from the figure originally published in the SINADRA software repository. © 2021 Jan Reich. 125

Figure 5.14 Situation class detection and actor role classification with intent-model selection. Ego localization in a situation class sets the context for searching and classifying relevant actors. Classified roles are mapped to Bayesian network identifiers that determine which intent models are evaluated for which actors. Adapted from the figure originally published in the SINADRA software repository. © 2021 Jan Reich. 126

Figure 5.15 Qualitative causal influences for front vehicle braking behavior. 127

Figure 5.16 Bayesian network model for front vehicle critical braking behavior intent prediction. The model was created with the GeNIe modeling software. Adapted from the figure originally published in the SINADRA software repository. © 2021 Jan Reich. 129

Figure 5.17 Process of semi-automatically transforming a design-time Bayesian network model in the GeNIe modeling tool into an executable Python inference component based on the pgmpy library..... 130

Figure 5.18 Transformation from behavior intent to trajectory distribution. Behavior-specific motion models generate distributions rather than single trajectories by sampling probabilistic parameters (e.g., deceleration limits or time gaps) to represent variability in intent execution. Originally published in the SINADRA software repository. © 2021 Jan Reich. 131

Figure 5.19 Behavior-specific risk computation based on inferred behavior intents from the SINADRA BN and generated trajectory distributions for each intent. The final situation-aware dynamic risk estimate is produced by weighting the behavior-specific risk scores with the estimated behavior likelihoods. 132

Figure 5.20 Example Bayesian network for SINADRA-based critical braking behavior intent prediction for the red front vehicle considering the slower green FV front vehicle. Adapted from Fig. 2 in (Reich et al., 2021), with permission from Springer Nature. © 2021 Springer Nature Switzerland..... 133

Figure 5.21 Situation-aware front vehicle braking dynamic risk assessment in uncritical scenario phase. Reproduced from Fig. 3 in (Reich et al., 2021), with permission from Springer Nature. © 2021 Springer Nature Switzerland. 134

Figure 5.22 Situation-aware front vehicle braking dynamic risk assessment in critical scenario phase. Reproduced from Fig. 3 in (Reich et al., 2021), with permission from Springer Nature. © 2021 Springer Nature Switzerland..... 135
Figure B.1 Illustration of formal traceability between CBCT output deviation modes and hazard elements in the safeTbox HARA model. 155
Figure B.2 safeTbox minimal cut set analysis result dialog with the possibility to filter and process CBCT minimal cut sets by using the formal CBCT element types. 156
Figure B.3 ODE::SINADRA Metamodel in the Open Dependability Exchange (ODE) Format. 157
Figure C.1 SubCBCT *IS-OVTK* for the overtaking scenario in the warehouse case study..... 159
Figure C.2 Realization of four SubCBCTs Focus, Perceive, Understand, Act of initiator scenario IS-OVTK. 160
Figure C.3 Qualitative causal influences for side-vehicle lane change intent. 162
Figure C.4 Bayesian network model for side vehicle critical cut-in behavior intent prediction. 162
Figure C.5 Intermediate formats of semi-automated BN runtime inference Python components. 163

List of Tables

Table 2.1 State-of-the-art assessment and research gap. Safety justification criteria: C1 traceability from risk and hazardous situations to assumption values, C2 hazardous situation coverage, C3 system safety reasoning, and C4 systematic engineering and assurance process. Symbols in C1–C4: + = largely addressed, o = partially addressed, - = not addressed. 34

Table 4.1 Example results of the initiator scenario analysis for the hazardous situation “Front vehicle braking in highway car-following exceeds the ego assumption.” 53

Table 4.2 Concretized HAZOP guideword set to support the identification of capability deviations and agent deficiencies. Full set with examples in Appendix A.2. 70

Table 5.1 Scoping of evaluation focus for the primary case studies and produced evidence for evaluation criteria. 100

Table 5.2 Structured Environment Model (SEM) excerpt for the hazardous situation “unsafe distance between a human worker and an AMR during co-located operation in narrow corridors.” 103

Table 5.3 Warehouse Ontology Excerpt. 104

Table 5.4 Four minimal cut sets of “ODM-IS-PASS: Unsafe distance HW-AMR in passing scenario”. 112

Table C.1 Four minimal cut sets of “ODM-IS-OVTK: Unsafe distance HW-AMR in overtaking scenario”. 161

1 Introduction

1.1 Motivation

Autonomous systems such as automated driving systems (ADS) or autonomous mobile robots can yield substantial societal and economic benefits by improving operational efficiency, enhancing travel comfort, and addressing labor shortages (Kubasáková et al., 2024; Wang, 2024). Their deployment, however, is subject to one non-negotiable condition: sufficient safety, as demanded by applicable standards and regulations.

Today's engineers guarantee safety across diverse environments by designing systems that can handle every worst-case situation. This strategy keeps safety assurance tractable, because if the system withstands the worst case, it is safe in all possible situations. However, the downside is a loss of permissiveness – the ability to operate flexibly and efficiently under real-world conditions. Overly conservative behavior triggers unnecessary interventions in the vast majority of non-worst-case situations, producing long delays, extended queues, premature stops, abrupt braking, and excessive headways that can cascade into congestion (Almusawi et al., 2024; Chen et al., 2025; Li et al., 2023). In the extreme, the system deems every path falsely unsafe and ends up in a "freezing-robot" standstill (Trautman & Krause, 2010).

The core open problem is therefore how to assure safety without sacrificing permissiveness – the safety-permissiveness trade-off. Improving this trade-off requires moving from worst-case assumptions to a more complex scheme in which the system reasons dynamically about the present *risks in each individual situation at runtime*. Through such *dynamic risk assessment* (DRA), permissiveness can be increased case by case while safety remains demonstrable.

Because accidents arise from undesired system behavior within a specific environmental context, DRA must evaluate how environmental factors influence risk. In complex operational settings like road traffic or warehouses, worst-case environmental assumptions are especially restrictive. Human drivers perform DRA intuitively: They read contextual risk cues – especially about other road users' behavior – and adjust their actions to the momentary risk. Current autonomous systems, though equipped with runtime safety monitors, still lack the ability to interpret and respond to these nuanced risk variations, especially those introduced by environmental conditions.

Consequently, increasing permissiveness *safely* calls for safety monitors to be capable of *demonstrably safe* and *situation-aware* dynamic risk assessment at runtime. Delivering this capability is the central objective of the *Situation-Aware Dynamic Risk Assessment* (SINADRA) framework presented in this dissertation.

The remainder of this introduction situates existing work, identifies the research gap, outlines the SINADRA solution, and previews its evaluation, contributions, and significance.

1.2 Research Problem

We begin by examining how state-of-the-art safety monitors incorporate environmental assumptions in DRA and why, so far, relaxing these assumptions to boost permissiveness is not justifiable from a safety assurance standpoint. Although conventional monitors do track both the system state and the surrounding environment at runtime, they still rely on static worst-case assumptions about environmental conditions, such as anticipating emergency braking by others at all times. To mitigate the resulting loss of permissiveness, recent research proposes monitors that adapt environmental assumptions dynamically. These approaches fall into three broad categories:

(1) **Data-based methods** extract assumptions from statistical analysis of observed behavior but suffer from data limitations and a lack of causal clarity. Because the correlations are not embedded in a formal risk model and usually not integrated with the system safety concept, justifying the adapted assumptions across the operational design domain (ODD) is difficult.

(2) **Motion-based methods** adapt safety constraints at runtime by predicting trajectories and subsequently inferring which assumptions are currently justifiable given the prediction. Full motion prediction depends on many factors unrelated to risk, so the models become complex and hard to interpret. This complexity increases the risk of prediction errors, which can cascade to incorrect risk estimates. Consequently, verifying motion-based assumptions against safety requirements remains problematic.

(3) **Risk-based methods** establish the relationship between situation factors and safety constraints directly from hazard analysis and risk assessment (HARA) artifacts. This approach is promising because it aligns with established safety engineering practice that is required by industry standards and avoids the need to predict general motion. Instead, it exclusively identifies and predicts those factors that influence risk. By limiting adaptation to factors already considered in safety analyses, these methods offer a clearer path to safety justification. However, one major challenge remains: There is no systematic engineering method for translating high-level hazardous situations into specific, verifiable, and runtime-monitorable situation hypotheses enabling a dynamic assumption adaptation.

To date, safety monitors cannot be made more permissive *safely* with data-based and motion-based approaches because the causal chain linking environmental conditions, their impact on hazardous situation risk, and specific relaxed assumptions remains implicit. This opacity undermines safety argumentation: Without an explicit cause-and-effect rationale, it is difficult to supply compelling evidence that any deviation from worst-case assumptions is still safe. In contrast, risk-based methods do follow the safe-by-design principle, but have not yet been operationalized to deliver more permissive monitors.

This thesis develops the structured **Situation-Aware Dynamic Risk Assessment (SINADRA) engineering methodology** to identify low-risk situations based on HARA artifacts and to support runtime adaptation of safety monitor assumptions, thereby enhancing system permissiveness while providing justification evidence. Because SINADRA's primary goal is to make more permissive safety monitors defensible from a safety assurance standpoint, the engineering framework adopts risk-based methods as its methodological foundation.

The **research question** guiding this work is:

"How can hazard analysis and risk assessment artifacts be extended to systematically identify monitorable low-risk situations that justify adaptive permissiveness-increasing constraint relaxation in runtime safety monitors for autonomous systems?"

Permissiveness is the degree to which an autonomous system may relax conservative assumptions to operate efficiently without breaching residual risk limits. A *low-risk situation* is a set of environmental conditions where a high-risk hazardous situation is implausible, because the conditions that would make it relevant and risky are unlikely to occur at that moment.

To develop and evaluate SINADRA, this dissertation follows the *Design Science Research Process* (Peffers et al., 2006). The resulting artifact is the SINADRA engineering method.

1.3 Solution

A central design ambition of SINADRA is to make assumption relaxation an engineering object rather than an ad hoc runtime heuristic. Data-based and motion-based methods often adapt assumptions without explicit derivation from, or traceability to, risk models and the system safety concept, which complicates review and safety argumentation. SINADRA addresses this by grounding the relaxation logic in design-time safety artifacts and by producing explicit trace links from hazardous situations to monitorable low-risk hypotheses and their runtime validity conditions. This does not make the

relaxation “safe by itself”, but it makes the admissibility conditions and the supporting evidence inspectable and auditable within the stated scope, with conservative fallback when validity cannot be established.

To achieve this, SINADRA closes a key gap between design-time safety work products and runtime adaptation. It links hazardous situations, their situational dependencies, and relevant safety measures that influence risk to the concrete runtime logic that relaxes assumptions. This trace chain supports a structured safety argument for when non-worst-case assumptions are admissible, based on stepwise checks of SINADRA’s intermediate models against the HARA work products and the stated modeling scope. The framework builds on, and extends, concepts from three families of industry standards so that relaxed environmental assumptions become justifiable:

(1) **HARA frameworks** like those proposed in ISO 26262 (International Organization for Standardization, 2018), IEC 61508 (International Electrotechnical Commission, 2010), or MIL-STD-882E (U.S. Department of Defense, 2023) provide a structured way to analyze hazardous situations and their risks. These are the starting point for decomposing risk variability due to environmental conditions in SINADRA.
(2) **Environment modeling concepts** in ISO 21448 (International Organization for Standardization, 2022) and ISO 34503 (International Organization for Standardization, 2023) define how to characterize and constrain the operational environment. These provide the foundation for a detailed analysis of the relationship between environmental conditions and risk.
(3) **Assumption analysis concepts** in IEEE 2846-2022 (Institute of Electrical and Electronics Engineers, 2022) and IEEE 3321-2024 (Institute of Electrical and Electronics Engineers, 2025) help specify and evaluate assumptions on reasonably foreseeable behavior of other road users in safety-related models. Since safety monitors use such models, and assumptions are the central artifacts of interest in SINADRA, these standards are an important input for this thesis.

SINADRA does *not* claim to improve the quality of functional safety HARA or SOTIF analyses per se. Rather, it leverages their concepts and techniques as the best available foundation for justifying permissiveness-improving assumption relaxation at runtime, which is a concern not yet covered normatively.

The SINADRA engineering methodology introduces three solution elements:

(1) **Structured Environment Models (SEM).** Guide phrases, coupled with an environment ontology, help safety engineers to systematically identify environment factors that influence hazardous situations and their risks. The resulting structured environment model forms the architectural basis for analyzing risk variability with Critical Behavior Cause Trees.

(2) **Critical Behavior Cause Trees (CBCTs).** As the core of SINADRA, these novel modeling structures support structured checks, within the modeled scope, that assumption relaxation is only applied under explicitly justified conditions. The unified CBCT model establishes traceability between three previously unrelated aspects: (1) the cause-effect chain from environment conditions to hazardous situation risks, (2) SINADRA risk implausibility hypotheses – monitorable combinations of environment conditions indicating low-risk situations, and (3) other safety measures such as operational design domain (ODD) restrictions or behavior constraints that also influence risks. Minimal cut set analysis on this Boolean model is used to check whether every modeled risk scenario is covered. A step-wise, guide-phrase-driven engineering process supports systematic construction, review, and critique of the CBCT model and its assumptions.

(3) **Runtime Risk Implausibility Detection Models**. CBCTs are *design-time* artifacts that provide safety evidence. Runtime adaptation, however, requires executable rules suggesting that assumptions change given the current environment conditions. In the final SINADRA step, the relevant parts of each CBCT are extracted and converted into a runtime model deployable alongside the safety monitor – either as a Bayesian network or a Boolean model, depending on ODD complexity. Because this conversion is algorithmic, traceability is achieved automatically and consistency between design-time and runtime artifacts supports consistency arguments between design-time artifacts and runtime checks, and reduces the risk of ad hoc mismatches.

These solution elements were instantiated in two case studies in distinct operational contexts. The warehouse case study demonstrates an end-to-end SINADRA application with explicit monitor-driven behavior change based on risk implausibility hypothesis validity signals. The urban ADS case study demonstrates runtime feasibility at component level by integrating SINADRA Bayesian network (BN) runtime inference models with collision risk computation and showing runtime risk-output differentiation relative to a worst-case intent baseline. Across both studies, increased permissiveness is treated as admissible only under explicitly argued conditions and only when the corresponding runtime validity signal can be established from available observations. Otherwise, a conservative fallback to worst-case assumptions applies.

1.4 Thesis Contributions

This thesis contributes a method and supporting artifacts for engineering assumption adaptation in runtime safety monitors with explicit trace links to design-time safety work products and conservative fallback when validity cannot be established. It makes the following contributions:

(1) The Situation-Aware Dynamic Risk Assessment (SINADRA) engineering method for justified adaptation of environmental assumptions in runtime safety monitors (Chapter 3), comprising

(1.1) Structured environment modeling that refines hazardous situations into a hazard-specific environment model with explicit coverage checks (Section 4.1).
(1.2) Critical Behavior Cause Trees (CBCTs) that refine hazardous situations into behavior risk scenarios, relate them to risk implausibility hypotheses and other safety measures, and enable verification evidence via minimal cut set analysis (Section 4.2).
(1.3) Runtime model transformation that maps argued risk implausibility hypotheses to observable features and operationalizes them with runtime inference capabilities (Section 4.3).

(2) Supporting tool and implementation artifacts that demonstrate partial model-based operationalization of the method, including a CBCT modeling frontend and the generation of Boolean and Bayesian runtime software components based on the *Digital Dependability Identity* (DDI) framework (Schneider et al., 2015; Zeller et al., 2023) (Section 4.4). The toolchain is presented as feasibility support rather than as an independent research contribution.

(3) Two case studies that apply SINADRA in different operational design domains and provide evidence for methodological feasibility and runtime feasibility within the stated scope (Sections 5.1 and 5.2).

1.5 Significance

This thesis matters because it turns assumption adaptation in runtime safety monitors into an explicit, reviewable object of safety engineering. SINADRA provides a concrete artifact chain that links design-time safety work products to runtime admissibility conditions for non-worst-case assumptions. This improves how such adaptation can be reviewed, communicated, and argued in a safety case: the assumptions, their context conditions, and their relation to other safety measures are made explicit and can be inspected rather than remaining implicit in tuning decisions or embedded in code.

The contribution is relevant to several communities. For runtime assurance and safety monitor engineers, SINADRA offers a method blueprint and artifact set for structuring and documenting assumption adaptation decisions in a way that supports systematic review. For data-driven approaches, SINADRA provides a way to focus empirical work on risk-relevant dependencies: the CBCT structure can be used to narrow down which variables and relations should be measured and tested, reducing model search space and helping to

target validation effort where it matters for risk. For safety case and assessment stakeholders, SINADRA yields intermediate artifacts that can be used as review anchors, because they expose the reasoning chain from hazardous situations to admissibility conditions and to runtime checks within the stated scope. It also provides reusable patterns that can inform emerging practice on explicit behavioral assumptions in safety-related models, including work in the IEEE 2846 and IEEE 3321 context.

1.6 Thesis Structure

This thesis is structured as follows. Chapter 2 reviews the state of the art and derives criteria for justified assumption adaptation in runtime safety monitors, leading to the research gap and research question. Chapter 3 introduces the SINADRA concept and provides a solution overview. Chapter 4 defines the SINADRA engineering method in detail, covering structured environment modeling, environment safety analysis based on Critical Behavior Cause Trees (CBCT), and the transformation of argued risk implausibility hypotheses into executable runtime logic, including an overview of the implemented tool prototype. Chapter 5 applies SINADRA in two case studies and evaluates the method against the stated evaluation criteria, including a cross-case summary. Chapter 6 discusses the findings, threats to validity, limitations, contributions, impact, future work, and concludes the dissertation.

2 State of the Art

Chapter 1 motivated the need for runtime safety monitors to support justified, context-dependent permissiveness without relying on worst-case assumptions in all situations.

This chapter advances the argument in three respects. First, it establishes a shared understanding of how this thesis uses key safety terms such as hazard, hazardous situation, risk, safety measures, and runtime monitors. Second, it derives criteria for when non-worst-case assumptions in runtime monitors can be considered justified. Third, it assesses how far current work meets these criteria and substantiates the existence of a scientific gap.

Chapter 2 is structured as follows. Section 2.1 introduces background concepts. Section 2.2 analyzes practice and standards guidance and derives criteria for justified use of non-worst-case assumptions in runtime safety monitors. Sections 2.3 to 2.5 review data-based, motion-based, and risk-based assumption adaptation approaches and assess them against these criteria. Section 2.6 summarizes the findings as a research gap and states the research question. Section 2.7 introduces the Design Science Research Process used to structure the development and evaluation in later chapters.

2.1 Background

At this point, the thesis has introduced the research problem and given an overview of the proposed solution. The role of this section is to provide the background concepts and terminology that will be assumed as known in later chapters.

The background explains three things: (1) how this thesis understands safety, hazards, hazardous situations, and risk in a general system safety sense with automotive examples, (2) how runtime safety monitoring, safety constraints, and dynamic risk assessment fit into this safety view, and (3) how assumptions regarding other agents and the environment, situational hypotheses, and the main artifacts of SINADRA relate to each other. Together, these elements provide the basis for the criteria for justifiably safe non-worst-case assumptions in Section 2.2 and for the method described in Chapters 3 and 4.

2.1.1 System safety terminology in this thesis

This subsection defines the terminology and basic concepts of hazard-based system safety engineering used in this thesis.

System safety in this thesis follows the dependability taxonomy introduced in (Avizienis et al., 2004). A system provides a service to users in a given environment. Dependability groups properties such as reliability, availability, and safety. Safety is the property that the system does not cause unacceptable risk of harm. A failure is a deviation of the service from its correct behavior. An error is a part of the system state that can lead to a failure. A fault is the cause of an error. This view is not restricted to the automotive domain. It is used across many safety-critical domains such as aviation, rail, and robotics, and this thesis adopts it as a general basis.

On top of this general view, this thesis uses a hazard-based way to reason about safety. A **hazard** is a potential source of harm that arises from the combination of a function and its environment. For road vehicles, a hazard is often described by an item, a function, and a traffic context. A **hazardous situation** is a state of the system and its environment in which the hazard can lead to harm. Examples include a crossing situation where another road user could be hit or a following situation where a rear-end collision is possible. Risk is the combination of the severity of possible harm, the exposure of the situation, and the controllability by the involved persons. Different standards use different labels for these elements, but the underlying idea is consistent.

In this thesis, the term hazardous situation is used instead of hazardous event. The term hazardous event is common in automotive standards, where it often combines a hazard with an operational situation and emphasizes events inside the item that can change risk. SINADRA has a different focus. It analyzes how the behavior of agents in the environment changes the risk that certain hazards will lead to harm. For this purpose, it is more natural to speak of situations and how they evolve than of events originating in the system. The term hazardous situation fits this focus better because it makes the dependence on the environmental situation explicit.

Risk assessment in this thesis follows standard practice in system safety engineering. For each hazard and each operational situation in scope, the engineering process aims for a residual risk level that is considered acceptable under the chosen risk acceptance criteria. Safety measures are then selected and designed so that the residual risk remains at or below these criteria. This thesis does not define a new risk metric. It relies on the idea that acceptable residual risk is constrained by agreed risk acceptance targets for the hazards and situations in scope, and that these targets guide the design of functions and safety measures. More specifically, SINADRA makes explicit how risk

varies with environmental context and uses this variability as an input to justified assumption adaptation in runtime safety monitors.

The link between hazards, hazardous situations, risk, and safety measures is usually organized as a chain. The intended function is analyzed to identify hazards in relevant situations. The hazard analysis and risk assessment (HARA) process groups hazards, assesses risk, and derives safety goals or safety objectives. These goals are refined into safety requirements. Safety measures are allocated to these requirements to achieve acceptable residual risk. In the automotive domain, the artifacts in this chain are commonly described as the HARA and the safety concept. In other domains, different terms may be used, but the overall structure is similar.

The later chapters of this thesis build directly on this chain. Hazard analysis and risk assessment provide the starting point for SINADRA's Structured Environment Model and the Critical Behavior Cause Tree. These artifacts preserve the link to hazardous situations and risk and refine it into a form that can be used for justified assumption adaptation.

2.1.2 Runtime safety monitoring, safety constraints, dynamic risk assessment

Based on this hazard-focused view of system safety, this subsection introduces runtime safety monitoring and explains how safety constraints and dynamic risk assessment use situational context to keep residual risk in hazardous situations within the defined risk acceptance criteria.

In the dependability view, safety is achieved by a combination of **safety measures**. The term safety measures refers to means that help to avoid, remove, or handle faults and failures in order to keep the risk of harm acceptable. Examples include design rules, verification and validation activities, and protective functions during operation. In the automotive domain, standards such as ISO 26262 use the term **safety mechanism** for technical means that detect faults or limit their impact. Safety mechanisms are therefore one class of safety measures. This thesis adopts this classification and places runtime safety monitoring within it.

A central concept for this thesis is the **safety constraint**. A safety constraint is a rule on system behavior that must hold to keep the residual risk of the hazards in scope acceptable in a given operational situation. Typical examples include bounds on speed, distance, acceleration, or other quantities that influence the likelihood and severity of harmful outcomes. The set of all defined safety constraints is called the **safety envelope**. If the system stays inside this envelope, residual risk is intended to remain within the defined acceptance criteria, under the stated assumptions.

A **runtime safety monitor** is a safety mechanism that evaluates relevant safety constraints during operation and helps maintain their satisfaction. It observes relevant aspects of the system and its environment and checks whether a constraint holds or is about to be violated. If the monitor detects that a constraint would be violated, it triggers a reaction. This reaction can be a limitation of the nominal function, a switch to a safer mode, or fallback behavior. In this way, runtime safety monitoring helps keep the system inside the design-time safety envelope derived from hazard analysis and risk assessment.

For runtime safety monitoring to be meaningful, the monitor must not only check a constraint against internal signals. It must also check whether the constraint still matches the situation the system is currently in. This motivates dynamic risk assessment. **Dynamic risk assessment** means that, during operation, the system repeatedly assesses whether the current behavior and situation keep the residual risk of the hazards of interest within the defined risk acceptance criteria. The monitor plays a key role in this by using sensor data and other observations to determine whether the specified safety constraint is still appropriate for the current situation or whether a different constraint set should apply.

Situational context has a strong influence on both the plausible behavior of other agents and the resulting risk. Examples include traffic density, road type, weather and visibility, and the presence of vulnerable road users. In dense urban traffic, closer distances between vehicles may be expected and acceptable, while the same distances may be risky at high speed on a highway. This means that the **assumptions about other agents and the environment** that underlie safety constraints depend on the operational situation. When the context changes, assumptions that were appropriate before may become too optimistic, or assumptions that were overly conservative may no longer be needed.

Runtime safety monitoring therefore requires **adaptation of assumptions**. The constraints enforced by a monitor are not only numeric rules. They embed assumptions about how other agents behave, how the environment can change, and how the system itself performs. To keep the safety envelope both acceptable and useful under changing context, these assumptions may need to change. This adaptation should remain traceable to the hazards and risk acceptance criteria defined in the safety concept. One main goal of this thesis is to support such adaptation in a way that remains justifiable.

In this setting, **permissiveness and utility** capture different aspects of performance. Permissiveness describes how little the monitor constrains the system when extra restriction is not needed to satisfy the stated risk acceptance criteria. A highly permissive monitor allows the nominal function to operate with wide parameter ranges and richer behavior when this is

justified. Utility refers to resulting benefit for users and stakeholders, such as comfort, travel time, or service availability. Utility is influenced by permissiveness but also by the use case and context. SINADRA focuses on the permissiveness of runtime safety monitors and provides means to argue that monitors can be less restrictive in many situations while still keeping residual risk acceptable within the defined criteria.

The notion of low-risk situations links these ideas back to hazardous situations and risk. A **low-risk situation**, in the sense of this thesis, is a situation in which, under stated and checkable conditions, certain hazardous situations are implausible. In such situations, the system can operate with more permissive assumptions and constraints while still meeting the defined risk acceptance criteria. Runtime safety monitoring and dynamic risk assessment therefore aim to recognize such low-risk situations during operation. If they do so in a justified way, they can adapt assumptions in favor of permissiveness while retaining a conservative fallback when the conditions indicate plausible risk.

2.1.3 Assumptions, situation hypotheses, SINADRA artifacts

This subsection explains how safety constraints rely on assumptions about other agents and the environment. It introduces situation hypotheses as a way to structure and monitor these assumptions at runtime. It also outlines the SINADRA artifacts that connect assumptions and hypotheses to hazard-based risk reasoning and to runtime monitoring.

Safety constraints encode claims about how the system, other agents, and the environment behave in a given operational situation. These claims function as **assumptions** that support the definition of a safety envelope. If the assumptions hold under the current conditions, the safety constraints are intended to keep residual risk of hazardous situations within defined risk acceptance criteria. If the assumptions do not hold, the same constraints can become insufficient or unnecessarily restrictive. Constraint adaptation therefore implies assumption adaptation.

In this thesis, an assumption is a statement used as a premise in safety reasoning about properties or behavior of other agents and the environment. Examples include bounds on braking capability of other vehicles, limits on reaction times of human drivers, or expectations about adherence to traffic rules. Assumptions often contain numeric values or qualitative levels. These values are called **assumption parameters**. For example, an assumption parameter can specify a maximum deceleration of a front vehicle, a time gap that other drivers are assumed to keep, or a bound on lateral offset that a pedestrian is assumed not to exceed.

For many hazards, a single fixed parameter value is too cautious in some operational situations and too optimistic in others. This thesis therefore uses

parameter classes. A parameter class groups parameter values that are treated as justified for a particular type of operational situation under stated conditions. The monitored system can then switch between parameter classes based on the current context. For example, a low-speed urban class might allow a smaller minimum time gap than a high-speed highway class.

Situation hypotheses provide a bridge between assumptions and the current operational context. A situation hypothesis is a structured claim about the operational situation and relevant agent behavior that is used as a premise in safety reasoning. It expresses, at an abstract level, what kind of situation the system believes it is in. This matters because the same assumption parameter can be acceptable under one hypothesis and too optimistic under another. For example, an assumption about maximum braking of other vehicles may be justified under dry road and good visibility, but not under wet road and poor visibility. Runtime monitoring can use situation hypotheses to determine which assumptions and parameter classes are applicable at a given time.

Industry standards that address behavioral assumptions highlight the need to relate assumptions to hazards, risk acceptance criteria, and operating conditions. However, they do not provide a structured method for deriving monitorable situation hypotheses and for justifying runtime adaptation of assumption parameters based on observations. This motivates the criteria developed in the remainder of Chapter 2 and the method introduced in later chapters.

2.2 State of the Practice: Criteria for justified adaptive runtime monitors

After Section 2.1, the conceptual basis for hazard-based safety reasoning and runtime monitoring is in place. This section defines what it means, in this thesis, for an adaptive runtime monitor to be justified when it applies non-worst-case assumptions.

The section distills criteria from two sources. First, it distinguishes two complementary forms of assurance evidence used in practice: outcome-based validation and design-based verification. Second, it reviews standards guidance on assumptions about other agents, focusing on IEEE 2846-2022 and IEEE 3321-2024, and extracts expectations placed on assumption artifacts. Based on these inputs, the section defines four criteria for justified non-worst-case assumptions in runtime monitors. These criteria guide the assessment of data-based, motion-based, and risk-based approaches in Sections 2.3 to 2.5.

In practice, assurance evidence comes in two complementary forms.

In practice, **outcome-based validation** exercises implementations in simulation, on test tracks, and on the road and evaluates safety performance against stated risk acceptance targets for the declared ODD. However, limitations in scenario representativeness, rarity of safety-critical events, and practical bounds on validation effort restrict the confidence that can be achieved through outcome evidence alone. **Design-based verification** therefore complements validation by establishing traceability and consistency between hazard analysis, safety goals, the safety concept, and implemented safety measures, including runtime monitors. For adaptive monitors, this includes traceability from hazards to monitor criteria and parameterization, and an explicit account of how monitor behavior interacts with other measures such as ODD restrictions, behavior constraints, and fallback behavior.

A recent development in standards guidance is to treat assumptions about other road users as explicit artifacts in safety-related models for automated driving. This section draws primarily on IEEE 2846-2022 and IEEE 3321-2024, which provide terminology, expected attributes, and recommended practices for handling assumptions about reasonably foreseeable road user behavior.

IEEE 2846-2022 (Institute of Electrical and Electronics Engineers, 2022) defines assumption categories and model attributes and frames how assumption parameter values relate to risk acceptance targets and coverage across the ODD. It also links assumption handling to systematic development artifacts and to validation and verification activities. Particularly, the clauses 5.1.2, 5.1.3, 5.1.11, 5.2.1 and 6.1 are relevant for deriving the criteria. IEEE 3321-2024 (Institute of Electrical and Electronics Engineers, 2025) complements this by providing recommended practice for selecting and applying parameter values and by emphasizing two open challenges that are central to runtime adaptation: ensuring context coverage of parameter values across the ODD and inferring the relevant context variables at runtime from observations.

For the purpose of this thesis, the key takeaway from these standards is not a specific algorithm. The key takeaway is a set of expectations that assumption artifacts should satisfy. They should be linked to hazards and risk acceptance criteria. They should have argued applicability across the range of relevant operating conditions. They should be supported by systematic work products and empirical validation where appropriate. At the same time, the standards do not prescribe a structured method for deriving monitorable situation hypotheses and for justifying runtime adaptation of assumption parameters based on observations. This gap motivates the criteria defined below.

Based on current practice and standards guidance, we define four criteria for arguing justified non-worst-case assumptions in runtime safety monitors. We

use these criteria to assess whether state-of-the-art assumption adaptation approaches can improve permissiveness while remaining aligned with hazard-based safety reasoning.

C1 Traceability to hazards and risk acceptance criteria. Assumption values used in safety constraints must be traceable to the hazardous situations in scope and to stated risk acceptance criteria. This traceability provides the basis for arguing why a specific parameterization is admissible under the intended conditions.

C2 Coverage across relevant situations and conditions. The set of assumption values and their applicability conditions must provide coverage across the relevant operating situations within the ODD, including the conditions that drive risk for the hazards in scope. This coverage expectation connects assumption artifacts to the scope and outputs of hazard analysis and risk assessment.

C3 Consistency with the system safety concept. Assumptions embedded in monitor constraints must be justified in the context of the overall safety concept. This includes how runtime monitoring interacts with other safety measures such as ODD restrictions, behavior constraints, and fault-handling mechanisms, and how these measures jointly achieve acceptable residual risk.

C4 Systematic engineering and assurance. The derivation, parameterization, and change control of assumptions should follow a systematic process aligned with established development, verification, and validation practices. This includes evidence that assumptions are reviewed, justified, and maintained as part of the safety work product chain.

The remainder of this chapter reviews data-based, motion-based, and risk-based assumption adaptation approaches and assesses them against these criteria.

2.3 Assumption adaptation based on data

Data-based approaches calibrate safety monitor parameters from observations. They use naturalistic driving data from field observations or synthetic data from simulation to estimate how often critical events occur, conditioned on observable context. At deployment, the monitor identifies the current context and selects corresponding parameter values. The underlying premise is that observed behavior can inform how conservative assumptions need to be in specific contexts. The question is to what extent these approaches can support justified permissiveness improvements across an operational design domain (ODD). This section reviews representative data-

based approaches and prepares the criteria-based assessment at the end of Section 2.3.

2.3.1 From naturalistic data to RSS parameter values

Idea. Responsibility Sensitive Safety (RSS) (Shalev-Shwartz et al., 2017) specifies rule-based distance constraints with kinematic parameters such as maximum deceleration of other road users, minimum braking capability of the ego vehicle, and response time. Data-based RSS studies use naturalistic driving data to estimate distributions of these quantities and to identify context features that correlate with typical human behavior. They then calibrate RSS parameter values, potentially per context, so that estimated rates of critical surrogate events remain below a chosen threshold.

For German freeway car-following, (Naumann et al., 2021) analyzed the highD dataset (Krajewski et al., 2018) and reported that worst-case RSS parameters would flag about half of observed headways as unsafe. Conditioning on additional cues, such as pre-predecessor behavior, and relaxing deceleration bounds reduced the number of apparent violations. (Xu et al., 2021) calibrated RSS car-following on naturalistic study data, and (Königshof et al., 2022) conducted a similar calibration for highway overtaking.

Strengths. These studies provide evidence that context influences operating margins and that a small set of interpretable parameters can be calibrated from data while remaining compatible with a rule-based monitor. They also show that worst-case parameterizations can be overly restrictive relative to typical observed behavior in a dataset.

Assessment. Data-calibrated RSS suggests that context can support less conservative envelopes, but the safety argument remains limited. The calibrated parameter values are fitted to observed kinematics and surrogate measures such as time-to-collision (TTC) and time-integrated TTC (TIT). These proxies abstract from concrete hazardous situations and rely on assumptions such as constant future speed and direction, which hold only in constrained contexts. For example, (Xu et al., 2021) fixed a four-second TTC threshold and optimized TIT with a conservativeness term, but did not relate the resulting values to specific hazardous situations and risk acceptance criteria. Evidence strength is also constrained by event selection and dataset scope, which leaves the applicability limits of relaxed values unclear. Reported performance gains, such as earlier braking or lower TIT in adaptive cruise control, were not integrated with other safety measures. Finally, the calibration pipelines were tuning-oriented and tool-driven and were not mapped to safety lifecycle artifacts such as hazard analysis and risk assessment, safety goals, or safety architecture.

2.3.2 Naturalistic data ranges for IEEE 2846 assumptions

Idea. The review in (IEEE VT/ITS/AV Decision Making Working Group, 2022) supports instantiation of IEEE 2846 assumption parameters by aggregating published ranges and data distributions for kinematic properties of pedestrians, cyclists, and vehicles. The compiled values are annotated with study context, including geography, year, and experimental setup.

Strengths. The review improves transparency about what empirical evidence exists and where it is thin. It allows engineers to inspect plausible value ranges, identify outliers, and understand how reported values depend on study context. This aligns with IEEE 2846's framing of assumptions as explicit model attributes and can support early scoping of monitor parameter settings.

Assessment. The review does not provide a method for selecting or justifying monitor parameter values for a specific system, hazard set, and ODD. It primarily surfaces limitations of the available evidence, including under-representation of vulnerable road users, missing metadata on infrastructure and weather, differences between vehicle types, and geolocation bias. Because the compiled ranges are not tied to a specific safety claim, they can at most serve as plausibility bounds and as an indicator of where additional evidence is needed.

2.3.3 Simulation-supported calibration

Idea. To mitigate rare-event sparsity and probe edge cases, some approaches combine naturalistic data with synthetic data from simulators. They then select parameter values that meet surrogate event-rate targets in the expanded dataset. (Liu et al., 2021) provides an example of this strategy.

Strengths. Simulation enables stressing conditions that are infrequent or unsafe to collect on the road. It also supports counterfactual variation and larger-scale sampling, which can tighten uncertainty bounds in targeted contexts.

Assessment. Simulation-supported calibration can strengthen the empirical story of data-based parameter tuning, but it does not automatically turn it into a safety argument. It also introduces additional modeling assumptions and selection effects. In (Liu et al., 2021), 200 emergent cut-in scenarios are extracted from a larger naturalistic corpus and replayed in MATLAB/Simulink. RSS parameters are then tuned via genetic optimization to improve surrogate metrics such as time-to-collision. This supports stress testing under more aggressive maneuvers. However, scenario selection and replay remain driven by kinematic triggers and modeling convenience rather than by a hazard-structured scenario set. The study also excludes runs in which the automated

vehicle selects an unexpected overtaking strategy instead of braking, which further shapes the resulting evidence. As a result, the calibration improves fit within the modeled simulation setting, but it does not by itself establish coverage of hazardous situations across the real ODD, quantify sensitivity to simulator assumptions, or show integration of the resulting parameterization into a standards-aligned safety concept.

2.3.4 Weaknesses of data-based approaches

Data-based approaches provide empirical evidence that worst-case parameterization can be overly restrictive. However, they do not provide sufficient basis for a safety case that explains when and why assumption relaxation is justified across an ODD. Three weaknesses stand out.

(1) **Missing hazard-based traceability.** Conditioning variables are selected from observational correlations rather than derived from hazard analysis and risk assessment. This prevents a clear reasoning chain from environmental conditions to hazardous situation risk and weakens traceability from risk acceptance targets to assumption values. This directly conflicts with C1 and with IEEE 2846-2022 expectations on relating assumptions to risk and on risk-relevant explainability.

(2) **Limited coverage evidence.** Rare severe events dominate safety claims, yet available naturalistic datasets offer limited statistical strength in these cases and exhibit systematic biases, including gaps for vulnerable road users, regional skews, and vehicle-type differences. As a result, ODD-wide coverage arguments remain weak, which conflicts with C2.

(3) **Weak maintainability under change.** Parameter values calibrated for one region, period, or traffic mix can drift as infrastructure and behavior patterns change. Without hazard-based links, it becomes difficult to assess drift relevance, update assumptions in a controlled way, and preserve the justification over time. This undermines C4 and complicates C3 when assumption updates interact with other safety measures.

Implication for this thesis. Data remains important, but it should be used as scoped evidence within a hazard-based structure rather than as the primary driver of assumption selection. Naturalistic data and simulation can, for example, inform uncertainty bounds for a specific situation hypothesis and parameter class or support targeted checks within a defined ODD slice. This turns data from a tuning instrument into a source of specific evidence that supports justified relaxation.

The next sections consider two alternative approach families. Motion-based and risk-based approaches infer admissible parameter values at runtime based on different ideas, as discussed in Sections 2.4 and 2.5.

2.4 Assumption adaptation with motion-based situation hypotheses

Data-based approaches select parameter values for safety monitors from observed outcomes and argue acceptability statistically. Motion-based approaches invert this logic into a predict, assess, and adapt pipeline. They forecast near-term motion of relevant agents, evaluate a criticality or risk metric over these forecasts, and adapt monitor parameters so that the resulting behavior remains within an acceptable risk bound. This can increase permissiveness when predicted motion indicates low near-term risk.

2.4.1 Safety constraints with dynamic object hypotheses: Risk-Aware RSS and RSS+

Idea. *Risk-Aware RSS* (Oboril & Scholl, 2020) and *RSS⁺* (Oboril & Scholl, 2021) add predictive situation hypotheses to Responsibility Sensitive Safety. Risk-Aware RSS replaces standard RSS, evaluating multiple object motion hypotheses and triggering responses when a formal risk over the predicted hypotheses exceeds a threshold. In contrast, RSS⁺ keeps the original envelope but relaxes parameter values to non-worst-case and adds motion prediction and risk assessment components as a second layer if standard RSS flags situations as safe. The principles of both approaches are illustrated in Figure 2.1.

Strengths. Both approaches target the permissiveness–safety tradeoff explicitly. They report fewer unnecessary interventions and fewer collisions than fixed worst-case envelopes. RSS⁺ preserves the familiar rule structure and keeps prediction simple and efficient.

Assessment. Risk-Aware RSS and RSS+ illustrate why "prediction-first, risk-second" relaxation is difficult to justify under the criteria defined in Section 2.2. Both add a predictive guard that aggregates collision risk over a set of kinematic hypotheses and compares it to a tunable threshold R_{max} to allow more permissive behavior. Because the hypothesis set is not derived from concrete hazardous situations with associated risk acceptance criteria, R_{max} is difficult to parameterize. The resulting lack of traceability from relaxed values to hazardous situations (C1) leaves coverage dependent on which hypotheses are enumerated rather than argued across an ODD (C2). Safety also becomes dependent on the sufficiency and correctness of the predictions. A wrong or incomplete hypothesis can label as acceptable a case that worst-case RSS would restrict, yet the papers do not integrate these relaxations with other safety concept measures (C3). Finally, the pipeline is presented as an add-on evaluated in simulation with manually set parameters, without mapping the new threshold, hypotheses, and detection integrity to hazard analysis and risk assessment, safety goals, or assurance artifacts (C4). In short, the mechanism used to gain permissiveness shifts the assurance burden toward motion prediction choices that are not constrained by hazard-based reasoning.

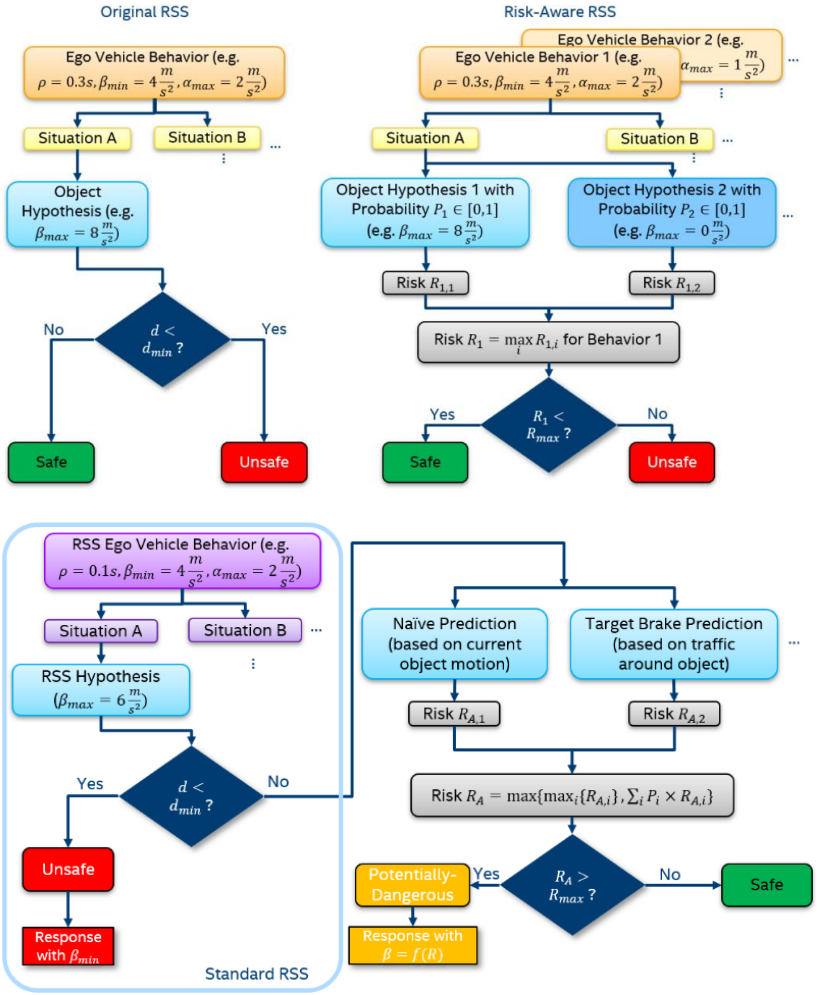


Figure 2.1 Illustration of the working principles of Risk-Aware RSS (top) and RSS⁺ (bottom). Top: Reprinted, with permission, from Fig. 8 in (Oboril & Scholl, 2020). © 2020 IEEE. Bottom: Reprinted, with permission, from Fig. 3 in (Oboril & Scholl, 2021). © 2021 IEEE.

2.4.2 Formal rule composition with environmental assumptions: goal-aware, compositional RSS

Idea. *Goal-aware RSS* (Hasuo et al., 2023) reformulates RSS as a compositional formal program logic with goals, environmental assumptions stated as preconditions, and safety conditions. It decomposes scenarios into sub-scenarios and proves, under the stated preconditions, that the composed rules enable goal-directed behavior such as safe pull-over while avoiding collisions.

Strengths. The approach introduces modularity into RSS reasoning. It also makes explicit how context, such as multilane adjacency and speed regimes, can justify behavior that is less conservative than one-size-fits-all envelopes while retaining formal reasoning.

Assessment. Goal-aware RSS advances formal composition, but two issues limit how it supports justified assumption adaptation. First, the proof assumptions, such as constant-speed behavior of other vehicles, are introduced as preconditions for tractability rather than derived from hazard-based risk reasoning with explicit risk acceptance criteria. The paper also relies on a separate collision-avoidance layer in a Simplex architecture to handle assumption violations, which leaves the relationship between relaxed preconditions, hazardous situations, and safety concept measures unclear. Second, the workflow is explicitly best-effort. It does not provide a formal criterion for sub-scenario decomposition, proper-response completeness, or precondition strength. The authors report that the derived rule set applied to 52% of scenario instances and defer broader coverage to future work. As a result, there is no coverage-by-construction claim across an operational design domain and no explicit applicability bounds for the relaxations.

2.4.3 Prediction-centered behavior planning with predictive risk maps: Honda risk-based framework

Idea. The European Honda Research Institute proposed a risk-based behavior planning stack that classifies scenes into situation hypotheses, performs multi-agent motion prediction conditioned on these hypotheses, computes predictive risk maps using the Integral Collision Risk model, and selects ego behavior by trading predicted risk against utility (Eggert, 2018; Eggert, Damerow, & Klingelschmitt, 2015; Klingelschmitt et al., 2016). Figure 2.2 illustrates the architecture.

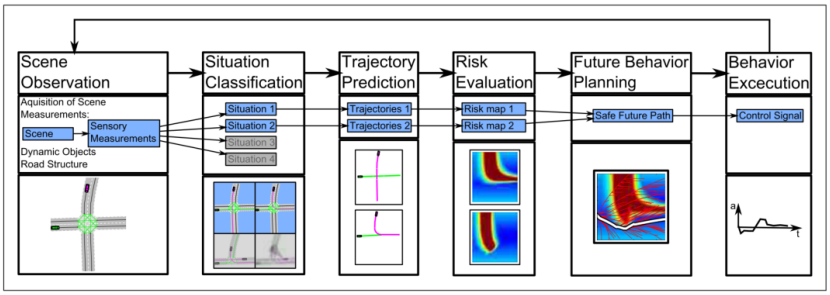


Figure 2.2 Overview of the situation-based risk evaluation and behavior planning framework. Reproduced from Fig. 1 in (Eggert, Klingelschmitt, & Damerow, 2015), licensed under CC BY 4.0.

Strengths. Situation hypotheses reduce computation by focusing on relevant interactions. Predictive risk maps provide a forward-looking basis for planning and can reduce unnecessary conservatism compared to worst-case

envelopes. Reported studies in highway and urban settings suggest that the framework can increase permissiveness while achieving favorable safety outcomes in the evaluated scenarios.

Assessment. The framework illustrates how prediction-centered planning can scale in complex scenes, but it remains hard to justify under the criteria. Situation hypotheses and prediction models are introduced to keep planning tractable rather than to guarantee coverage of hazardous situations derived from hazard analysis and risk assessment. As a result, the link from relaxed assumptions to hazardous situations and risk acceptance criteria is unclear. The approach then drives behavior by minimizing predicted risk against utility, leaving hypothesis selection, prediction fidelity, and thresholding as design choices. Without a method to connect these choices to hazards, the declared ODD, and the system safety concept, it is difficult to argue coverage and consistency. In practice, safety becomes dependent on the correctness and completeness of trajectory-level predictions under perception and behavior uncertainty. If a relevant hypothesis is missed or predicted incorrectly, a situation may be treated as low risk even though a worst-case envelope would restrict it, and the framework does not provide a standards-aligned assurance pathway to bound this residual risk across the ODD.

2.4.4 Set-based reachability runtime verification with dynamic constraints

Idea. Set-based runtime verification checks, during operation, that reachable sets of the ego vehicle and other agents remain disjoint under a library of constraints rooted in traffic law, physics, and standards, while maintaining a fail-safe fallback such as safe stop (Pek et al., 2020). Because real traffic can violate legal constraints, runtime constraint management adapts parameters upon detected violations (Koschi & Althoff, 2021). Recovery methods based on Points of No Return (PNR) and Points of Guaranteed Arrival (PGA) support re-establishing safety when plans become invalid (Pek et al., 2017). Figure 2.3 summarizes the framework.

Strengths. The approach provides formal guarantees under stated assumptions and supports structured fallback under uncertainty. Explicit parameterization provides a clear handle for trading conservatism against permissiveness.

Assessment. The guarantees hold only under stated bounds on other agents' inputs, such as velocities and accelerations, and the framework explicitly acknowledges that these assumptions can be violated. Once violated, prior guarantees no longer apply and safety depends on recovery constructs such as PNR and PGA, which identify segments that remain safe and a safety-critical passageway that must be handled carefully. This leads to two justification gaps. First, the numeric bounds are not derived from hazardous situations and risk acceptance criteria for a declared ODD. Second, coverage

becomes dependent on the completeness and timely detectability of the assumed violation set. If relevant hazards fall outside that set or are detected too late, assurance weakens exactly where safety margins can be smallest. In addition, the framework is not integrated into a standards-aligned safety work product chain that links assumption specification, monitor integrity, and update policies to hazard analysis, safety goals, and requirements and V&V. The resulting argument is therefore conditional on assumption completeness and detection quality rather than grounded in hazard-based scope.

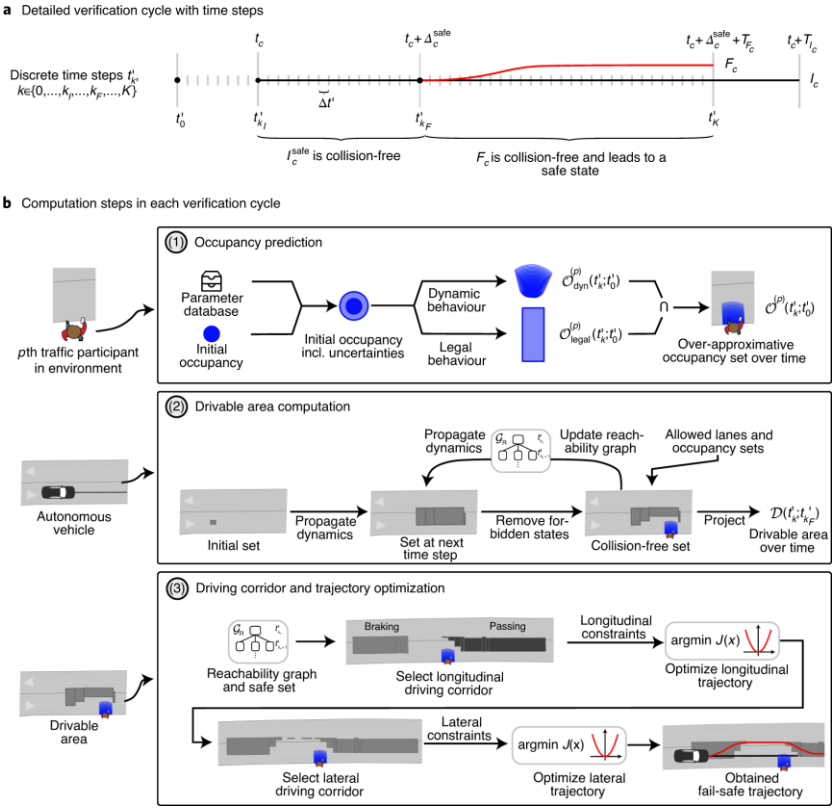


Figure 2.3 Set-based reachability runtime verification framework. Reproduced from Fig. 6 in (Pek et al., 2020), with permission from Springer Nature. © Springer Nature 2020.

2.4.5 Weaknesses of motion-based approaches

Taken together, the four motion-based strands show why “predict, assess, and adapt” can deliver permissiveness but struggles to deliver justified permissiveness under the criteria of Section 2.2.

(1) **Prediction first, risk second.** Motion-based approaches place prediction or assumption selection before risk justification. Risk-Aware RSS and RSS+ base relaxation on predicted collision risk over a small hypothesis set and a tunable

cutoff. Goal-Aware RSS composes rules under environmental assumptions introduced as preconditions. Honda's framework extends this to predictive risk maps, and reachability-based verification provides guarantees under stated bounds. Across these designs, assurance hinges on whether the chosen hypotheses, predictions, and bounds are appropriate, yet this appropriateness is not established through hazard-based reasoning.

(2) **No hazard-driven coverage.** Coverage across the ODD remains difficult to argue. Monitored situations and hypothesis sets are chosen for tractability and modeling needs rather than derived from hazard analysis with explicit linkage to unacceptable risk. Applicability limits for relaxed values therefore remain informal, which weakens C2.

(3) **Missing traceability from relaxed values to hazardous situations.** Risk thresholds, relaxed envelopes, and assumption bounds are introduced as engineering choices or physics and comfort heuristics rather than being linked to hazardous situations and stated risk acceptance criteria. This breaks the minimal traceability expected by C1.

(4) **Assurance shifts to prediction and assumption integrity instead of system safety reasoning.** A missed or incorrect hypothesis can label a case as acceptable that worst-case safeguards would restrict. Moreover, the reviewed works provide limited system-level and process-level assurance: relaxations are not integrated into a safety concept that combines ODD restrictions, behavior constraints, and fallback behavior with explicit residual-risk allocation (C3), and the associated workflows are not mapped to standards-aligned artifacts such as hazard analysis and risk assessment, safety goals, and requirements and verification/validation (V&V) (C4).

Implication for SINADRA. These findings motivate a reversed chain: derive monitorable situation hypotheses and applicability conditions from hazard analysis and risk assessment and integrate non-worst-case parameter classes as part of the safety concept, rather than introducing prediction-driven relaxations and justifying them after the fact. The next section therefore turns to risk-based approaches that follow this direction and that serve as building blocks for SINADRA.

2.5 Assumption adaptation with risk-based situation hypotheses

Section 2.4 showed that motion-varying approaches enlarge the hypothesis space to all motion-relevant factors, predict trajectories, and only then assess risk before adapting parameters. This expands model complexity, weakens traceability to hazards, and obscures coverage and justification. Risk-based approaches invert this chain. They start from Hazard Analysis and Risk Assessment (HARA) artifacts, derive situation hypotheses that encode risk-relevant influence factors, detect those hypotheses at runtime, and then adapt parameter values accordingly. In terms of functional chains:

- **Motion-based:** ODD → situation hypotheses representing **motion-relevant** influence factors → runtime detection of motion situation hypotheses → runtime risk assessment → adaptable parameter value.
- **Risk-based:** ODD → HARA artifact → situation hypotheses representing **risk-relevant** influence factors → runtime detection of risk situation hypotheses → adaptable parameter value.

The aim is increased permissiveness with a justification that is anchored in safety artifacts. Because hypotheses are derived from design-time safety work products, the parameter space can be restricted to factors that matter for risk plausibility within the declared scope.

2.5.1 Dynamic risk management

Idea. Dynamic Risk Management (DRM) provides an overarching framework for realizing intelligent runtime safety monitors from existing safety engineering knowledge (Adler et al., 2023; Schneider et al., 2024; Trapp et al., 2018). DRM shifts parts of the assurance process to runtime by structuring functions into Dynamic Risk Assessment (DRA) for external risk variability, Dynamic Capability Assessment (DCA) for ego-capability variability, and Dynamic Risk Control (DRC) for safe adaptation in a Monitor–Analyze–Plan–Execute (MAPE-MART/K) loop. The core idea is to extract safety knowledge from design-time artifacts and formalize it into runtime-inferable models so that monitors can adapt constraint parameters in a situation-aware and defensible way (Figure 2.4).

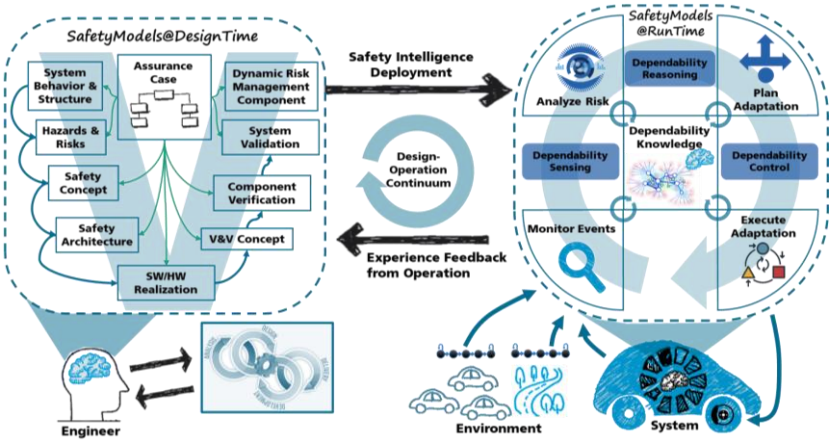


Figure 2.4 Dynamic Risk Management – conceptual overview.

Because SINADRA focuses on environmental assumptions in DRA specifically, it is important to note that capability monitors have formalized ways to monitor ego capability variation, e.g., Conditional Safety Certificates (Schneider & Trapp, 2013), predictive simulation for system timing and V2X

communication constraints (Cioroaica et al., 2022), Dynamic RSS for transition timing assurance (Salvi et al., 2022), SMOF rule synthesis for safety margins (Machin et al., 2018; Masson et al., 2018), and Skeptical Dynamic Dependability Management for learning-enabled components (Arnez et al., 2022; Ollier et al., 2023). The approaches derive runtime evidence from safety analyses, safety concepts, and architectures. However, these contributions address how to maintain safety envelopes under changing system *capabilities*. They do not supply a method for deriving risk-varying hypotheses for environmental assumptions that unlock justified relaxation of parameters in the presence of other agents' behavior.

Strengths. DRM explicitly distinguishes risk variability (DRA) from capability variability (DCA), which clarifies how environmental assumptions and ego capabilities jointly influence safety margins. Within DRA, it provides a placeholder to integrate risk-varying situation hypotheses and connect them to adaptation logic in DRC. Existing work shows how safety analyses, safety concepts, and safety architectures can seed runtime models, which offers a baseline level of assurance because these inputs are produced in established safety engineering processes.

Assessment. DRM is a solution framework rather than a concrete engineering method to derive monitorable, risk-varying hypotheses at the right granularity for relaxing environmental assumptions. It does not by itself deliver a safety justification for adapted safety constraints based on dynamic risk assessment. However, its foundation in design-time safety engineering processes and the models-at-runtime paradigm makes DRM a useful starting point for approaches that aim to improve the safety-permissiveness tradeoff with defensible reasoning.

2.5.2 Runtime representation of hazardous events

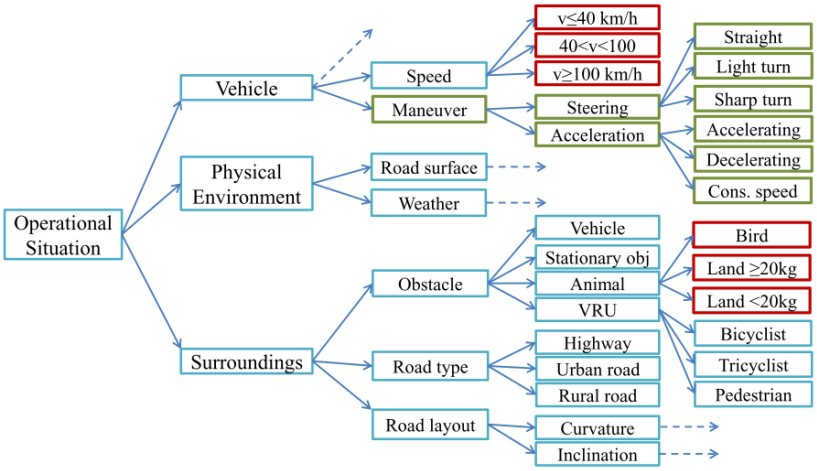
Idea. (Trapp et al., 2018) formalize how hazardous events can be represented at runtime using measurable situation descriptors and an activation probability within DRM. A hazardous event is represented as a tuple $h = \langle d, f, e, c, s, i, g \rangle$, with driving situation d, failure f, exposure e, controllability c, severity s, integrity level i, and a reference g to the safety goal. At runtime, this is mapped to $\tilde{h} = \langle \tilde{d}, \tilde{p}_e, \tilde{g} \rangle$, where the situation $\tilde{d}$ is specified by observable parameters $\vec{p}$, weights $\vec{w}$, and confidences $\vec{c}$: $\tilde{d} = \langle \vec{p}, \vec{w}, \vec{c} \rangle$ and where the probability of being in the situation is $\tilde{p}_e = \frac{1}{n} \sum_{i=1}^{n} p_i w_i c_i$. Hazards with $\tilde{p}_e < \tilde{p}_e{}^{min}$ are deactivated for the current context. This can reduce conservatism when hazard preconditions are implausible.

Strengths. The mapping makes the essential connection from HARA to runtime situations explicit. The contribution makes the connection from HARA artifacts to runtime situation representations explicit and provides

concrete parameters ($\tilde{p}_e^{min}$, indicators, weights, confidences) that can be integrated into monitoring logic.

Assessment. The focus is on representation, not on a method for deriving risk-based situation hypotheses that support relaxation of environmental assumptions. Indicator selection is left to the engineers, and the work does not demonstrate a systematic way to ensure that behavior-related risk drivers are captured.

HE	Operational Situation	Hazard	E	C	S	ASIL
1	AD on highway/rural road with animal obst.	Undetected object	2	3	2	A
2	AD with tricycle obstacle	Undetected object	2	3	3	B
3	AD with other vehicle obstacle	Loss of braking	4	3	3	D
4	AD in bright light with other vehicle obst.	Loss of braking	4	3	3	D
5	AD in high speed with stationary object obst.	Mode confusion	2	3	2	A



HE	Operational Situation	Hazard	E	C	S	ASIL
~~1~~	~~AD on highway/rural road with animal obst.~~	~~Undetected object~~	~~2~~	~~3~~	~~2~~	~~A~~
1b	AD on highway with **land animal ≥20 kg** obst.	Undetected object	**1**	3	**3**	A
~~2~~	~~AD with tricycle obstacle~~	~~Undetected object~~	~~2~~	~~3~~	~~3~~	~~B~~
~~2b~~	~~AD **on highway** with tricycle obstacle~~	~~Undetected object~~	**0**	~~3~~	~~3~~	~~QM~~
3	AD **on highway** with other vehicle obstacle	Loss of braking	4	3	3	D
5	AD **on highway** in high speed with stationary object obstacle	Mode confusion	2	3	2	A

Figure 2.5 Example hazardous situation refinement. Composite adapted from Table 2, Fig. 4, and Table 3 in (Warg et al., 2016). Reproduced with permission from Springer Nature. © Springer International Publishing Switzerland 2016.

2.5.3 Tree-based HARA refinement for variability

Idea. (Johansson & Nilsson, 2016; Warg et al., 2016) propose operational situation and hazard trees that iteratively refine hazardous situations to reduce unnecessary conservatism in derived safety goals. The refinement supports an *Environment Perception Block* that conditions required integrity on expected collision impact speed in a given situation. A sufficiency checklist requires that the generic trees are covered or justified as not applicable and that the chosen abstraction level does not force overly conservative safety goals. Refinement sharpens severity estimates using impact-speed surrogates and thereby supports lower integrity margins where justified while bounding risk elsewhere (Figure 2.5).

Strengths. The approach introduces an explicit decomposition method and a coverage-oriented checklist, and it treats permissiveness as a goal of the refinement.

Assessment. The trees structure design-time reasoning, but they do not provide a repeatable mapping from refined nodes to runtime-detectable hypotheses. Behavior-related context remains largely implicit, and refinements are primarily kinematic or topological. As a result, the approach does not produce monitor-ready artifacts that bind hypotheses to a concrete ODD slice and that support justified relaxation of constraint assumptions.

2.5.4 Safety supervisor

Idea. (Adler et al., 2016) introduced a method for deriving a runtime safety supervisor to monitor the dynamic integrity of system safety goals based on hazardous event risk analyzed during hazard analysis and risk assessment. The method determines risk-varying situation hypotheses related to accident severity, approximated via delta collision speed, and derives runtime criteria for classifying the required integrity of safety goals at runtime. The derivation from HARA risk parameters to runtime criteria builds on (Adler & Kemmann, 2014) and the *Structured Approach for Hazard Analysis and Risk Assessments* (SAHARA) framework (Kemmann, 2015). (Feth et al., 2017) formalized these concepts into a metamodel for deriving and evaluating safety supervisors. (Feth, 2020) reports application results for ADS at system and mission levels and demonstrates feasibility.

Strengths. The approach is explicitly traced to HARA and produces concrete runtime criteria, which improves explainability and auditability. It demonstrates feasibility on the system and mission levels for ADS.

Assessment. The primary focus is integrity management rather than engineering risk-varying hypotheses that justify relaxation of environmental assumptions in monitors. Situation granularity remains coarse and

predominantly kinematic. The work does not demonstrate how to incorporate environmental context that drives risk variability for other agents. Integration with other measures is described conceptually but not argued as combined sufficiency.

2.5.5 ASIL-inspired dynamic tactical safety

Idea. (Khastgir, Sivencrona, et al., 2017) compute real-time exposure, controllability, and severity to obtain a dynamic ASIL that guides tactical adaptations. The framework evaluates vehicle and environment parameters at runtime, assigns a risk category, and selects actions suited to the momentary risk level. (Khastgir, Birrell, et al., 2017) formalize risk parameter variability.

Strengths. Runtime choices remain explainable because they are guided by explicit risk ratings. The approach defines a mapping from dynamic risk estimates to adaptation actions and makes the safety-permissiveness tradeoff a design handle.

Assessment. The considered causal factors are limited to kinematic indicators such as speed, distance, and TTC. The method does not provide a procedure to derive behavior-related hypotheses that would justify relaxing assumptions about other agents. In addition, the mapping from ASIL bands to parameter value classes lacks explicit applicability conditions linked to risk acceptance criteria, and interaction with other safety concept measures is not addressed.

2.5.6 µODDs: Parameter classes per observable situation subset

Idea. (Koopman et al., 2019) propose partitioning the ODD into µODDs and associating each µODD with assured parameter classes, such as worst-case, conservative, and permissive. A monitor switches between these classes when an observable µODD holds. The proposal emphasizes suitability factors such as the expected permissiveness gain and the observability of µODD boundaries. A Bayesian formalization is suggested but not implemented (Figure 2.6).

Strengths. The µODD concept articulates exactly the abstraction needed by adaptive monitors: parameter value classes with observable applicability conditions. They make the safety-permissiveness tradeoff explicit and auditable and invite external trusted information (e.g., infrastructure signals) to strengthen observability.

Assessment. µODDs align closely with the conceptual direction pursued in SINADRA, but the proposal remains at the concept level. It is illustrated only on a simple example and does not provide a systematic method for deriving

µODDs from hazard analysis and risk assessment with explicit risk-relevant determinants.

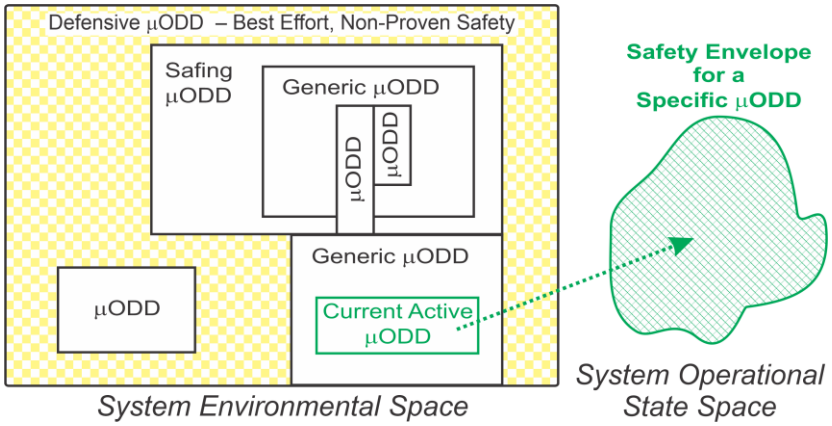


Figure 2.6 µODDs condition safety envelope definitions on specific situation subsets, where different sets of parameter values lead to the maximum permissiveness in a µODD and can be justified from a safety point of view. Reproduced from Fig. 3 in (Koopman et al., 2019) with permission of the authors.

2.5.7 Precautionary safety and the quantitative risk norm

Idea. Precautionary Safety (PCS) (Rodrigues de Campos et al., 2021) uses the Quantitative Risk Norm (QRN) (Warg et al., 2020) to adapt policy online so that predicted incident frequencies remain below acceptable thresholds. Rather than decomposing operational situations, QRN focuses on incidents and acceptable consequence frequencies conditioned on context such as location, speed, and distance. PCS then adjusts policy parameters, such as maximum speed, to keep predicted incident frequencies within the stated target.

Strengths. QRN defines a quantitative risk acceptance target, and PCS explicitly attempts to align policy choices with that target.

Assessment. PCS addresses dynamic risk assessment from a different direction than the other risk-based approaches reviewed in this section. However, the work does not specify how to realize the concept as a concrete monitoring solution. The authors point to *Dynamic Risk Management* (Trapp et al., 2018) as a candidate solution framework. The two remaining key open points are the required granularity of incident definitions and their dependence on operational context as well as the stability and representativeness of the statistics across regions and time, which directly influences derived policy limits.

2.5.8 Weaknesses of risk-based approaches

Risk-based approaches offer the most direct route toward justified permissiveness improvements because they align assumption adaptation with design-time safety artifacts and restrict runtime adaptation to explicitly risk-relevant influence factors. In contrast to data-based approaches, they do not primarily rely on observational correlations as a surrogate for safety reasoning. In contrast to motion-based approaches, they do not treat trajectory prediction and risk cutoffs as the main justification mechanism. The reviewed literature provides several important building blocks, including Dynamic Risk Management as an overarching runtime safety framework, runtime representations of hazardous events and their activation, systematic hazardous event refinement approaches, integrity-focused safety supervisors, dynamic risk rating concepts such as dynamic ASIL, and concept-level proposals such as µODDs.

Taken together, these works indicate what "good" risk-based adaptation could look like, but they do not yet close the engineering gap for runtime assumption relaxation in safety monitors. Three weaknesses stand out:

(1) **Missing end-to-end derivation of monitorable hypotheses.** The reviewed approaches start from HARA artifacts, risk ratings, or incident norms, but they do not provide a systematic method for deriving monitorable situation hypotheses at a granularity suitable for adapting environmental assumptions. In particular, behavior-related contextual factors for other agents remain under-modeled or implicit.

(2) **Weak operationalization into observable runtime evidence.** Several contributions provide design-time structuring and representation concepts, but they do not establish a repeatable mapping from refined hazards and risk drivers to observable runtime features with explicit uncertainty and integrity targets. As a result, the link between hypothesis intent and runtime detectability remains underspecified.

(3) **Under-specified parameter class design and applicability conditions.** While µODDs and related ideas articulate the need for parameter value classes with applicability conditions, the literature does not provide a systematic procedure for deriving these classes from risk acceptance criteria and for arguing when class switching is admissible. Where quantitative targets exist, such as QRN, the monitoring and implementation pathway is left open.

In summary, risk-based approaches provide strong foundations, but the literature still lacks a concrete engineering method that produces a reviewable and traceable artifact chain from hazard analysis and risk assessment to monitorable low-risk hypotheses, justified parameter value classes, and their runtime admissibility conditions.

2.6 Research Gap and Research Question

This chapter reviewed three families of approaches for adapting environmental assumptions in runtime safety monitors.

Data-based adaptation calibrates monitor parameters from observed outcomes in naturalistic data and, in some cases, simulation data, and switches parameter values at runtime based on detected context. It provides empirical evidence that worst-case parameterization can be overly restrictive. However, conditioning variables are commonly selected based on statistical associations rather than derived from hazard analysis and risk assessment (HARA). As a result, traceability from relaxed assumptions to hazardous situations and risk acceptance criteria is weak, coverage of risk-relevant ODD slices is difficult to argue, and interaction with the overall safety concept is often not addressed.

Motion-based adaptation predicts agent motions, evaluates risk over these predictions, and relaxes constraints when predicted risk is low. This can increase permissiveness in specific situations, but it shifts assurance toward motion prediction integrity and hypothesis completeness. Because hypotheses tend to cover motion-relevant influence factors rather than risk-relevant determinants, context complexity grows and coverage and justification remain difficult to establish.

Risk-based adaptation starts from HARA artifacts, derives risk-relevant situation hypotheses, detects them at runtime, and adapts parameter values accordingly. This aligns well with established safety engineering practice in principle. However, the literature still lacks a systematic method that produces runtime-monitorable hypotheses at the right granularity for assumption adaptation, ties them to pre-argued parameter value classes and complementary safety measures within a safety concept, and demonstrates permissiveness improvements under stated scope and evidence assumptions.

Table 2.1 summarizes how the three approach families address the central research problem of engineering and justifying permissiveness improvements in runtime safety monitors. The resulting gap motivates the research question stated next.

Data-based and motion-based approaches can increase permissiveness, but they do not provide a justification pathway that satisfies the criteria in Section 2.2. Risk-based approaches align better with safety artifacts by design, yet the literature still lacks an end-to-end engineering method that turns hazard analysis and risk assessment results into runtime-monitorable conditions for justified non-worst-case assumption use.

Table 2.1 State-of-the-art assessment and research gap. Safety justification criteria: C1 traceability from risk and hazardous situations to assumption values, C2 hazardous situation coverage, C3 system safety reasoning, and C4 systematic engineering and assurance process. Symbols in C1–C4: + = largely addressed, o = partially addressed, - = not addressed.

Class	Reference Approaches	Permissiveness Mechanism	Justification				Gap
			C1 Traceability	C2 Coverage	C3 System Safety	C4 Process	
Data-based	(Xu et al., 2021) (Liu et al., 2021) (Königshof et al., 2022)	Parameter distributions from naturalistic or synthetic data → assess context-conditioned risk → adapt assumptions to what is acceptable in context-conditioned distribution	-	-	o	-	**Permissive, but justification difficult** due to: 1. Missing traceability from hazards and risk acceptance criteria to assumption value 2. Insufficient sample size for safety claims 3. Outcomes confound monitor parameter choices with other safety measures and scenario selection 4. Limited mapping to standards-aligned safety engineering work products and review practices
Motion-based	(Eggert, 2018) (Pek et al., 2020) (Oboril & Scholl, 2020) (Oboril & Scholl, 2021)	Predict motion of other agents in current context → evaluate risk of predicted motion → adapt assumptions to what is acceptable in given prediction	o	-	-	o	**Permissive, but justification difficult** due to: 1. Traceability to hazards is possible in principle, but rarely established in the presented methods 2. Large prediction and context parameter space makes explainable coverage for specific risks difficult 3. Often treated as a standalone safety mechanism rather than integrated with a full safety concept 4. Engineering methods tightly integrated with nominal function development, less with safety engineering processes
Risk-based	(Johansson & Nilsson, 2016) (Adler et al., 2016) (Warg et al., 2017) (Feth et al., 2017) (Khastgir, Sivencrona, et al., 2017) (Trapp et al., 2018) (Koopman et al., 2019) (Campos et al., 2021)	HARA → predict risk influence drivers in current context → adapt assumptions to pre-argued values for particular risk changes in current context	+	o	o	-	**Justification is feasible, but permissiveness gains are limited** 1. Traceability to hazard analysis and risk assessment by design 2. Can reuse safety artifacts and refinement methods, but monitor-level coverage remains incomplete 3. Links to the system safety concept exist in principle, but are not consistently operationalized in the state of the art 4. No end-to-end method yet that derives monitorable low-risk hypotheses and justified non-worst-case parameter classes

This gap indicates that the safety-permissiveness tradeoff is not adequately addressed by the current body of knowledge. What is needed is a systematic engineering method for permissiveness-increasing safety constraint relaxation that remains justified within a safety case. Concretely, such a method must:

- refine HARA hazardous situations into an analyzable operational environment view that makes behavior-driven risk variability explicit and supports coverage evidence for the risk-relevant ODD slice,
- provide an integrated formalism that relates risk scenarios, low-risk situation hypotheses, and safety concept measures so that the admissibility of hypothesis-based relaxation can be argued, and
- transform argued hypotheses into executable runtime inference models with end-to-end traceability back to HARA artifacts.

To address this gap, the dissertation answers the **research question**:

“How can hazard analysis and risk assessment artifacts be extended to systematically identify monitorable low-risk situations that justify adaptive permissiveness-increasing constraint relaxation in runtime safety monitors for autonomous systems?”

The answer yields an operational engineering method that supports context-dependent permissiveness while preserving conservative fallback when admissibility cannot be established. It thereby links design-time safety work products to runtime monitor parameterization within a reviewable and auditable chain.

2.7 Design Science Research Process

This dissertation followed the *Design Science Research Process* (DSRP) by (Peffers et al., 2006). Figure 2.7 maps the six DSRP stages to the main activities and outputs of this thesis.

Design Science Research is suitable for this work because the goal is not only to analyze existing approaches, but to design and evaluate a concrete artifact that addresses the identified gap. In this thesis, the artifact is the SINADRA engineering method together with its intermediate models, checks, and runtime operationalization steps.

The DSRP model defines six stages (Figure 2.7): (1) problem identification and motivation, (2) objectives for a solution, (3) design and development, (4) demonstration, (5) evaluation, and (6) communication. The following chapters implement these stages by deriving the research gap and criteria in Chapter 2, defining the SINADRA concept and method in Chapters 3 and 4,

demonstrating and evaluating it in the case studies in Chapter 5, and consolidating findings, limitations, and implications in Chapter 6.

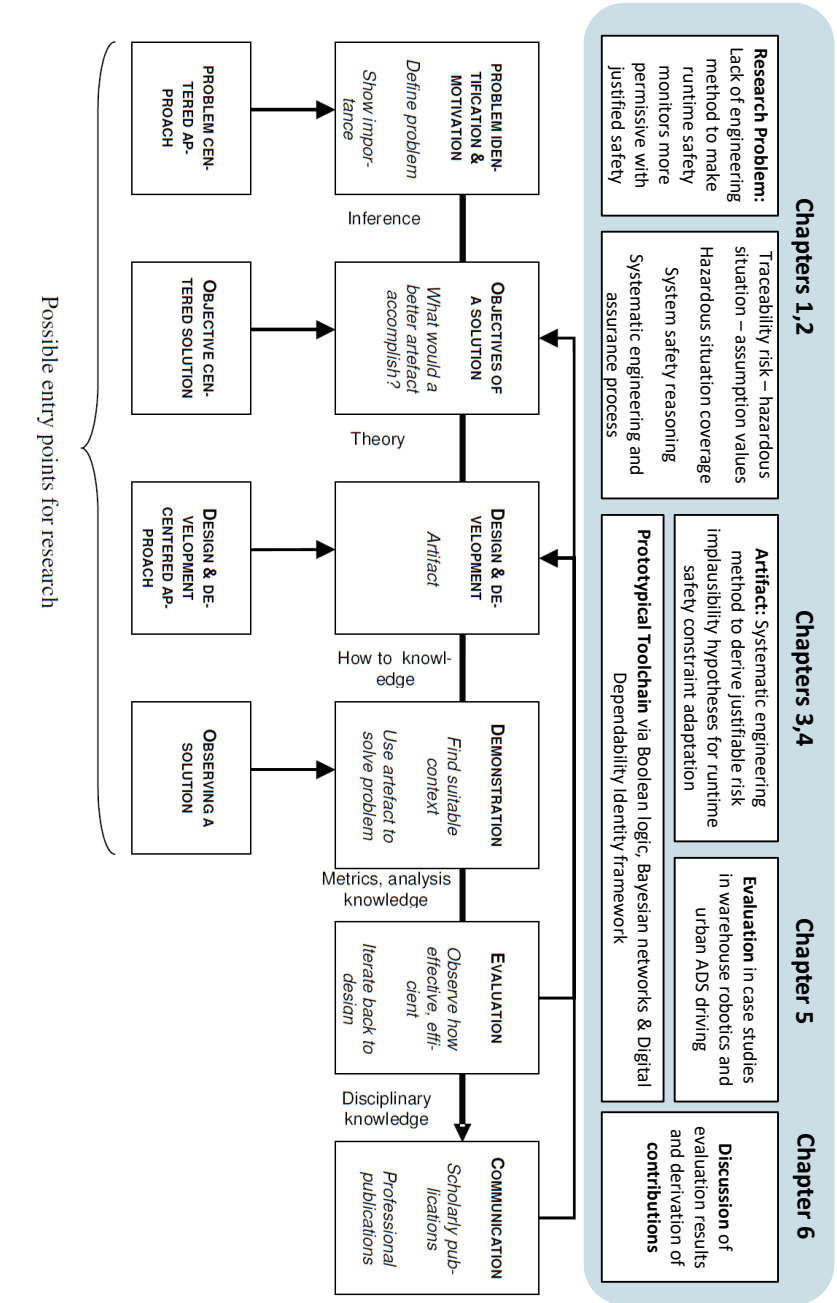


Figure 2.7 Design Science Research Process mapped to the chapters of this dissertation. Adapted from (Peffers et al., 2006); chapter mapping added by the author. Reprinted by permission of Informa UK Limited, trading as Taylor & Francis Group.

3 Solution Overview

Chapter 2 established the motivation for adaptive assumption use in runtime safety monitoring and derived the central research gap: current approaches do not provide a systematic, reviewable justification for when non-worst-case environmental assumptions are admissible. While data-based and motion-based methods can increase permissiveness, they typically lack a traceable link from hazards and risk acceptance criteria to the relaxed assumptions and their applicability conditions. Risk-based concepts align better with safety engineering practice, but the literature still lacks an end-to-end method that turns hazard analysis and risk assessment artifacts into monitorable runtime conditions and justified parameter adaptations.

This chapter introduces the **Situation-Aware Dynamic Risk Assessment (SINADRA) framework** as the solution proposed in this dissertation. SINADRA follows a risk-based direction and grounds runtime assumption adaptation in hazard analysis and risk assessment (HARA) artifacts. The initial solution concept is published in (Reich & Trapp, 2020).

The chapter serves two purposes. First, it explains the runtime concept: SINADRA permits non-worst-case assumptions only when explicitly defined risk implausibility hypotheses are satisfied in the current situation. Second, it provides an overview of the development-time method and its artifacts: Structured Environment Models (SEM), Critical Behavior Cause Trees (CBCT), and SINADRA runtime models. Together, these artifacts derive the hypotheses, connect them to hazardous situations and safety measures, and provide evidence to justify their use.

Section 3.1 introduces the runtime concept and its integration context. Section 3.2 presents the development-time method at a high level. Section 3.3 provides an example that illustrates the end-to-end idea before Chapter 4 defines each method step in detail.

3.1 SINADRA Solution Idea at Runtime

Based on the research question stated in Section 2.6, a risk-based solution for adaptive runtime monitoring should satisfy four design requirements:

(1) It should enable increased permissiveness compared to fixed worst-case assumptions.

(2) It must support a safety argument with evidence showing that non-worst-case assumptions are admissible under stated risk acceptance criteria and within a declared scope.

(3) It should be grounded in hazard analysis and risk assessment (HARA) artifacts to align with established safety engineering practice.

(4) It must provide a runtime concept that turns development-time artifacts into executable logic so that parameter adaptation is controlled and driven by observable signals.

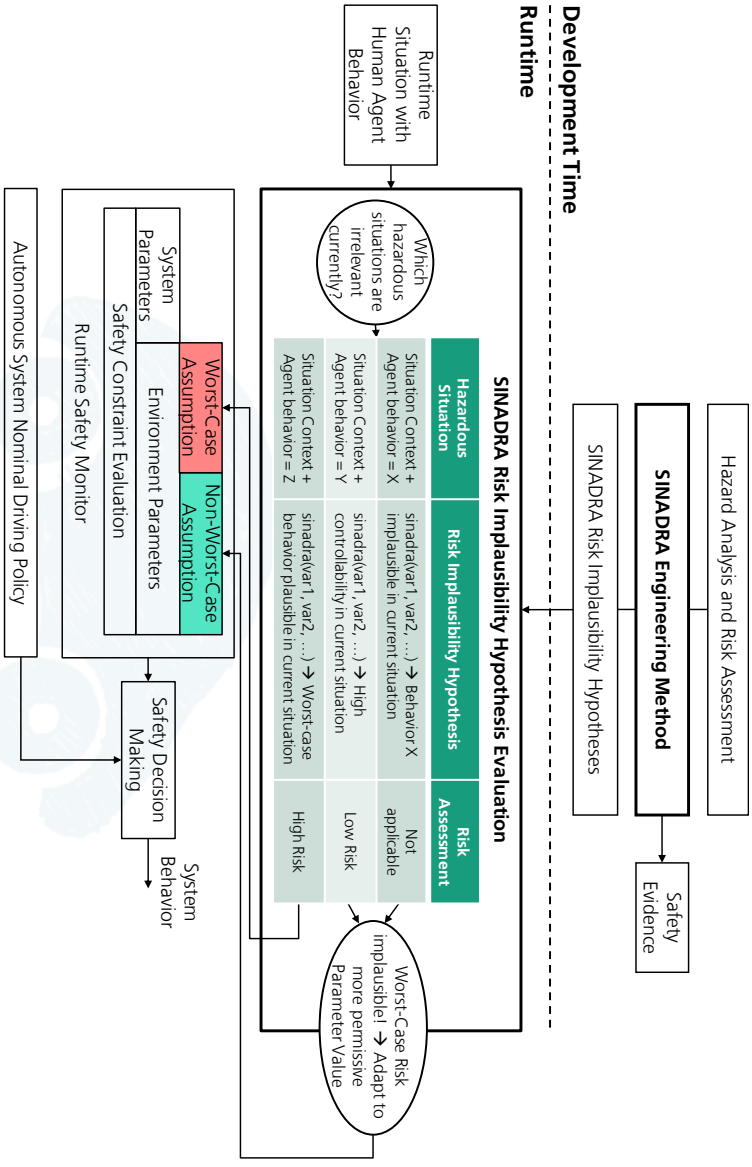


Figure 3.1 Situation-Aware Dynamic Risk Assessment (SINADRA) solution idea at runtime.

Figure 3.1 summarizes SINADRA's runtime concept. SINADRA adapts environment-related assumption parameters in a runtime safety monitor when available observations support that worst-case risk is implausible for the hazardous situations currently in scope. When this cannot be supported, SINADRA retains worst-case parameterization.

We assume an autonomous system with two components. The nominal driving policy plans behaviors and trajectories toward the destination. A runtime safety monitor evaluates a safety constraint over the planned behavior using current system and environment state and restricts or overrides the nominal policy when a violation is predicted. The safety constraint includes environment parameters that encode assumptions about nearby agents, for example bounds on lead-vehicle braking.

SINADRA uses risk implausibility hypotheses (RIH) to connect observable situation features to hazardous situations and their risk-relevant conditions. At runtime, SINADRA evaluates the RIHs relevant to the current context and assigns each hazardous situation to one of three cases (Figure 3.1).

(1) **Hazardous situation not applicable.** Available context indicates that the hazardous situation does not apply at the current moment. For example, if risk is driven by hard braking of a front vehicle, the hazardous situation is irrelevant when there is no lead vehicle or when there is no reason to brake hard.

(2) **Hazardous situation plausible but with non-worst-case risk.** The hazardous situation remains possible, but observable cues indicate that key risk drivers are reduced, for example due to higher controllability or lower severity. In this case, SINADRA selects a less conservative parameter class.

(3) **Worst-case risk still plausible.** If the observations do not support case (1) or (2), SINADRA keeps worst-case parameter values.

This runtime concept supports increased permissiveness only under stated applicability conditions and retains fallback to worst-case when risk implausibility cannot be validated from available observations.

In cases (1) and (2), SINADRA treats worst-case parameterization as unnecessary for the current situation and instructs the safety monitor to switch from worst-case to a predefined, more permissive parameter class that was justified at development time. In case (3), SINADRA retains worst-case parameterization.

This runtime concept depends on systematic development-time work. Risk implausibility hypotheses are the key artifact because they define the admissibility conditions for non-worst-case assumptions and the evidence required to justify them.

(1) Requirements for specifying risk implausibility hypotheses

For each assumption parameter and the situations in which it may be adapted, the runtime logic must have:

- a defined set of hazardous situations in scope and criteria for relevance in the current context
- an explicit link from the assumption parameter to the risk drivers of these hazardous situations
- a specification of interactions with other safety concept measures that jointly ensure acceptable residual risk

(2) Requirements for evidence that detected risk implausibility is justified

Assuming the availability of HARA artifacts and a safety concept that includes a runtime safety monitor and safety constraint, SINADRA must provide evidence that:

- the switch to non-worst-case parameter classes is explainable and reviewable
- the RIH set provides adequate coverage of the modeled risk-relevant environmental variability within the declared scope
- assumptions underpinning RIH validity, including assumptions on other safety measures, are supported by verification and validation evidence
- the implemented runtime evaluation matches the derived RIH definitions

Section 3.2 introduces the SINADRA engineering method that produces these RIHs, trace links, and evidence.

3.2 SINADRA Engineering Method

Section 3.1 introduced SINADRA's runtime concept: the safety monitor switches from worst-case to non-worst-case parameter classes only when explicitly defined risk implausibility hypotheses (RIH) are satisfied. This section explains how these hypotheses and their supporting rationale are produced at development time, starting from hazard analysis and risk assessment (HARA) work products.

Figure 3.2 summarizes the three elements of the SINADRA engineering method:

- **Structured Environment Models (SEM)** structure the risk-relevant operational environment for a hazard and guide systematic identification of context factors that influence other-agent behavior.

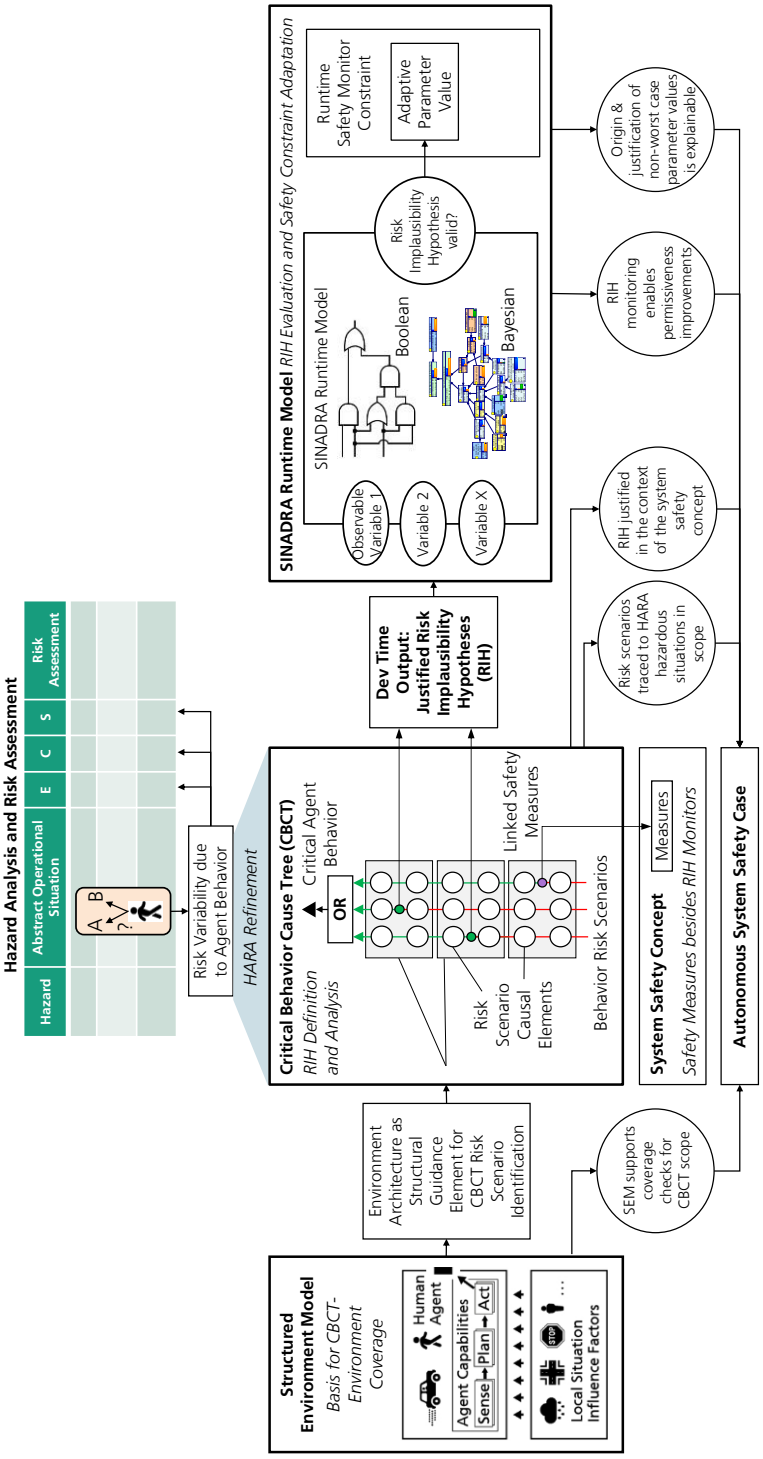


Figure 3.2 Situation-Aware Dynamic Risk Assessment (SINADRA) engineering method overview.

- **Critical Behavior Cause Trees (CBCT)** refine HARA hazardous situations into behavior risk scenarios and relate them to RIH candidates and complementary safety measures. CBCT analysis provides structured checks that support the argument for when relaxation is admissible within the modeled scope.
- **SINADRA runtime models** operationalize the resulting RIHs by mapping them to observable variables and executable runtime inference logic.

Together, these elements create an artifact chain from HARA hazardous situations to monitorable low-risk conditions and their runtime evaluation. This supports reviewable traceability from non-worst-case parameter classes back to the hazards, risk drivers, and safety measures considered at development time.

3.2.1 From HARA to CBCT: refining hazardous situations

Traditional HARA processes identify hazards and operational situations and assess risk parameters such as exposure, severity, and controllability. This level of abstraction is suitable for worst-case reasoning and for keeping hazard analysis tractable. It is, however, too coarse for SINADRA's objective: deciding at runtime whether a hazardous situation applies and whether worst-case assumptions remain necessary. Typical HARA entries do not spell out how specific environment conditions influence other-agent behavior and thereby change risk drivers. For worst-case-oriented safety design, assuming that critical behavior occurs is often sufficient. For justified non-worst-case adaptation, the engineering problem changes: the method must explain when critical behavior is implausible and when key risk drivers are reduced.

SINADRA introduces Critical Behavior Cause Trees (CBCT) as the central development-time refinement artifact (Figure 3.2). A CBCT is a Boolean cause–effect model that refines a hazardous situation into behavior risk scenarios. It makes explicit which combinations of environment conditions and agent behavior are relevant for the hazard and how they influence risk parameters. In this way, CBCTs represent risk-relevant variability driven by agent behavior within a stated modeling scope.

Based on this refinement, SINADRA introduces risk implausibility hypotheses (RIH) as monitorable conditions that can justify non-worst-case parameter values at runtime. A RIH does not claim that "risk is absent." Instead, it is defined to block a behavior risk scenario from unfolding in the risk-relevant way, or to support that worst-case risk is implausible given observable cues. For example, when a pedestrian at the curb of a crossing has noticed an oncoming vehicle and indicates intent to wait, a scenario in which the pedestrian steps into the road unexpectedly is less plausible. CBCT structure

is what enables such conditions to be defined in a focused manner and tied back to specific scenario branches.

Not all useful conditions are reliably observable. Sensing limitations and uncertainty can prevent direct monitoring of some scenario factors. SINADRA therefore relates RIHs to complementary safety concept measures, which serve as additional barriers for those scenarios that cannot be addressed by RIH monitoring alone.

To support a reviewable safety argument, SINADRA applies minimal cut set analysis on the CBCT representation. The analysis is used to check, within the CBCT model, whether the combination of RIHs and linked safety measures addresses each modeled behavior risk scenario.

The application of the CBCT solution element is shown in the warehouse robotics case study in Section 5.1.

3.2.2 Achieving environment coverage in CBCTs with SEM

A central risk in CBCT-based refinement is omission: a CBCT may fail to model a relevant behavior driver or context factor. In that case, the derived RIH set would not provide adequate justification for non-worst-case adaptation because unmodeled scenario branches could remain unaddressed. SINADRA reduces the likelihood of such omissions by using a Structured Environment Model (SEM) as structured guidance during CBCT construction (Figure 3.2).

For a given abstract hazardous situation, the SEM structures the operational environment and highlights what should be considered during refinement. It captures (1) the agent roles that can contribute risk-relevant behavior (for example other human drivers or pedestrians) and the operational situation classes in which these roles matter, and (2) the relevant capabilities and constraints that shape behavior in these situations (for example perception, decision-making, and decision execution).

The SEM is not a risk model by itself and does not define behavior risk scenarios. Its purpose is to guide systematic CBCT refinement by prompting engineers to consider a hazard-specific set of environment factors and behavior drivers. This supports more consistent CBCT scoping and review and makes omissions easier to detect.

Together, HARA, SEM, and CBCT provide the intended artifact chain for justified adaptation:

- the HARA defines the set of hazardous situations considered in scope
- the SEM structures the risk-relevant environment factors and agent roles for each hazardous situation

- the CBCT refines hazardous situations into behavior risk scenarios and relates them to RIHs and complementary safety measures, enabling structured checks and trace links that support a safety argument for admissible non-worst-case assumptions

The application of the SEM solution element is shown in the warehouse robotics case study in Section 5.1.

3.2.3 Runtime model transformation: from RIH to monitor logic

Up to this point, risk implausibility hypotheses (RIH) exist as development-time objects defined in the CBCT structure. Runtime use requires these hypotheses to be expressed in terms of what the system can actually observe and compute online. SINADRA therefore transforms the selected RIHs into an executable runtime model (right side of Figure 3.2).

The transformation has two goals. First, it operationalizes RIH evaluation by mapping CBCT RIH elements to concrete observable signals and defining evaluation logic that can be executed at runtime. Depending on the use case and the required uncertainty treatment, this logic can be realized as Boolean rules or as probabilistic inference, for example using Bayesian networks. Second, it connects RIH validity outcomes to monitor adaptation by specifying which safety monitor parameters are eligible for adaptation and which predefined parameter classes (worst-case versus non-worst-case) are admissible under which conditions.

The output of this step is a deployable runtime component that consumes observable variables, evaluates the relevant RIHs, and provides a validity output that can be used by the safety monitor to select the appropriate parameter class. Conservative handling of missing or uncertain evidence is essential: if RIH validity cannot be established from available observations with the required confidence, the runtime logic does not relax assumptions and the monitor remains in worst-case parameterization. This preserves traceability because each runtime check and each parameter class is linked back to the underlying RIH definition and, through CBCT and SEM, to the refined hazardous situation context.

The application of the runtime model transformation solution element with Boolean models is shown in the warehouse robotics case study in Section 5.1. The application with Bayesian networks is shown in the urban ADS driving case study in Section 5.2.

3.2.4 Safety justification chain and criteria coverage

SINADRA connects development-time safety artifacts to runtime admissibility conditions for non-worst-case monitor parameter classes. The method starts from hazardous situations identified in the HARA, refines them into behavior risk scenarios using CBCTs, and uses SEM as structured guidance to reduce omissions during refinement. From the CBCT structure, SINADRA derives risk implausibility hypotheses (RIH) as monitorable low-risk conditions and relates them to complementary safety measures. Runtime models then operationalize RIH evaluation by mapping them to observable variables and executable logic that can drive controlled parameter-class selection in the safety monitor.

This artifact chain supports the four safety justification criteria from Section 2.2:

C1 Traceability between risk, hazardous situations, and assumption values. SINADRA provides explicit trace links from HARA hazardous situations to behavior risk scenarios, to RIH definitions, and to the monitor parameters that may be adapted. RIHs specify the context conditions under which a hazardous situation is treated as not applicable or under which worst-case risk is treated as implausible for a particular scenario branch.

C2 Coverage of hazardous situations. SEM structures the risk-relevant environment and agent roles for each hazardous situation in scope and supports systematic review of what is considered during CBCT refinement. CBCTs then represent behavior risk scenarios as a model basis for coverage checks. Minimal cut set analysis is used to assess, within the CBCT model, whether each modeled scenario is addressed by at least one RIH or by linked safety measures.

C3 System safety reasoning. SINADRA does not treat the runtime safety monitor in isolation. CBCTs explicitly relate RIH-based adaptation to other safety concept measures, such as ODD restrictions, behavior constraints, and fallbacks, so that admissibility of non-worst-case parameter classes is argued in the context of the overall safety concept and stated assumptions.

C4 Systematic engineering and assurance. SINADRA defines roles, inputs, and checks for SEM, CBCT, and runtime model transformation, and ties them to established safety work products (HARA, safety concept, system architecture, and runtime monitor specification). This supports reviewable, standards-oriented safety argumentation rather than ad hoc parameter tuning.

3.3 Example: Pedestrian Crosswalk

To close the solution overview, this section illustrates the SINADRA artifact chain on a pedestrian crosswalk scene. The example is intentionally small and incomplete. It is meant to clarify the role of each artifact and the kind of content it carries.

HARA fragment. Consider an automated vehicle approaching a marked crosswalk in an urban street while a pedestrian is near the curb. A suitable abstract hazardous situation at HARA level is: *"Ego vehicle passes the crosswalk area while a pedestrian enters the roadway, leading to a collision or near-collision."* At this level, the HARA captures the hazardous situation and its risk relevance in the declared scope, but it typically does not specify which context factors make pedestrian entry more or less likely.

SEM fragment. For this hazardous situation, a SEM structures the operational environment into the parts that matter for behavior and risk. It captures, at a high level: the relevant agents and roles (ego vehicle, pedestrian at curb, possible occluding vehicles), the scene structure around the crosswalk (regions, lines of sight, occluders), and initiator scenarios that reflect why the pedestrian is there (crossing intent versus waiting versus passing by). It also makes explicit which pedestrian capabilities matter for safe interaction (noticing, interpreting, deciding, acting), because capability degradation is a typical driver of behavior risk scenarios.

CBCT fragment. A CBCT then refines the abstract hazardous situation into behavior risk scenarios, expressed as causal paths that explain how risk can arise in this scene. For illustration, consider: pedestrian enters while unaware (distraction or occlusion), and pedestrian enters after misjudging gap or yielding expectations. These behavior risk scenarios provide the "targets" for defining low-risk conditions, because SINADRA can only justify relaxation if it can argue why these risk-driving paths are blocked or downgraded under the current context.

Safety concept measures. Not all risk paths are suitable for monitorable implausibility checks. Therefore, the CBCT also references other safety concept measures that contribute to risk control, such as a fixed speed restriction near marked crosswalks, conservative fallback when pedestrian perception health is degraded, or automatic emergency braking. In SINADRA, these measures are treated as part of the same combined argument for keeping residual risk acceptable.

RIH and runtime model. In the CBCT, SINADRA defines risk implausibility hypotheses that can be checked at runtime and tied to a monitor parameter change. In this scene, a direct "irrelevance" hypothesis is that no pedestrian is present in the interaction region. A direct "reduced risk" hypothesis is that

a pedestrian is present, but observable cues indicate that near-term road entry is unlikely and that the interaction remains controllable in the current moment. Concrete cues include that the pedestrian remains behind the curb, is oriented toward the roadway (or toward the ego vehicle) without stepping forward, and shows no approach motion toward the crosswalk entry. If the vehicle is already at low speed and has sufficient distance to the crosswalk, severity and controllability are also improved. Runtime models map these hypotheses to ego-observable features such as pedestrian presence in the crosswalk interaction region, pedestrian distance-to-curb, pedestrian velocity and motion direction relative to the roadway, pedestrian body orientation (coarse bins such as facing road versus facing away), ego speed, and time-to-crosswalk (or distance-to-crosswalk), and then select among pre-argued parameter classes. When these features are unavailable or unreliable, the monitor retains the worst-case setting.

In summary, the crosswalk scene shows how SINADRA links one hazardous situation to a small set of behavior risk scenarios, derives monitorable risk implausibility hypotheses, and maps them to ego-observable features for controlled switching among pre-argued parameter classes with worst-case fallback when validity cannot be established. In Chapter 4, the detailed method is introduced. Section 4.1 uses an example related to front-vehicle braking in a car-following scenario to illustrate structured environment modeling. Sections 4.2 and 4.3 return to the example related to the crosswalk-related pedestrian scenario.

The two example scenarios are chosen because they emphasize different, method-relevant properties. The front-vehicle braking example is dominated by scene structure and interaction roles (ego, lead vehicle, infrastructure context) and therefore fits the SEM purpose of systematically enumerating environment factors and agent capabilities that drive risk variability. The crosswalk-related pedestrian example is dominated by behavioral uncertainty and observability constraints (whether and why a pedestrian may enter the roadway, and which cues can be monitored with sufficient integrity), which makes it better suited to demonstrate CBCT-based refinement into behavior risk scenarios and the definition of risk implausibility hypotheses.

4 Method

Chapter 3 introduced SINADRA as a risk-based framework with three elements: Structured Environment Models (SEM), Critical Behavior Cause Trees (CBCT), and runtime models for parameter adaptation. This chapter specifies these elements in an engineering method. Section 4.1 defines SEM and shows how HARA hazardous situations are refined into initiator sources, initiator scenarios, and agent capabilities. Section 4.2 introduces CBCTs and explains how they relate behavior risk scenarios, risk implausibility hypotheses, and other safety measures in a single Boolean model used for structured checks. Section 4.3 describes how assured hypotheses are transformed into executable runtime logic that adapts monitor parameters while maintaining traceability to HARA, SEM, and CBCT. Section 4.4 summarizes the prototype toolchain that partially supports these steps.

4.1 Structured Environment Modeling (SEM)

SINADRA increases monitor permissiveness only under stated low-risk conditions that can be justified from safety artifacts. This requires more structure than typical HARA operational situation classes provide. For SINADRA's purpose, the key missing detail is why critical behavior of other agents becomes plausible in a given context, and which external conditions and capability demands shape that plausibility. SEM provides this structure by refining a hazardous situation into three linked elements: (1) initiator sources in the environment that can trigger the behavior, (2) initiator scenarios that capture the context in which a source becomes active, and (3) agent capabilities required for safe handling of the scenario. The resulting SEM instance is the input for Section 4.2, where CBCTs use it to derive behavior risk scenarios and to support later derivation of monitorable risk implausibility hypotheses.

Section 4.1.1 defines the SEM concepts and their relation to HARA and the monitor assumption in scope. Section 4.1.2 specifies the engineering method from an ODD and an environment ontology and summarizes the trace links and review checks used before CBCT modeling.

4.1.1 From HARA to risk-driven environment models

This subsection introduces Structured Environment Modeling (SEM) as a refinement step that bridges from HARA hazardous situations to CBCT analysis. SEM links each hazardous situation to three elements that make risk variability explicit: initiator sources, initiator scenarios, and agent capabilities.

In road traffic, risk often depends on how nearby human agents behave in context. If a runtime safety monitor is allowed to use non-worst-case assumptions in some situations, then the safety argument must explain when those assumptions are valid and when they are not. This requires a structured description of the external conditions that can trigger critical behavior and the capabilities an agent needs to handle those conditions safely.

Example. In highway car-following, a safety monitor may rely on a non-worst-case bound for front-vehicle braking to avoid unnecessary restrictions in normal traffic. The associated hazardous situation is that the front vehicle brakes more strongly than the assumed bound, which can create a rear-end collision risk. The same hazardous situation can occur for different reasons, such as yielding to an emergency vehicle, reacting to a cut-in, or responding to a downhill section. These contexts differ in their triggers, cues, and timing, even though the kinematic outcome looks similar.

HARA entries usually stay at a coarse level because that is sufficient for high-level risk rating. For SINADRA, the coarse level is not sufficient because the method must decide, at runtime, whether a non-worst-case assumption is justified in the present situation. SEM addresses this by decomposing the environment into (1) agent-external preconditions that activate demanding situations and (2) agent-internal capabilities needed for safe handling. This creates a reviewable structure that remains linked to the original HARA item and can be used as the input frame for CBCT modeling.

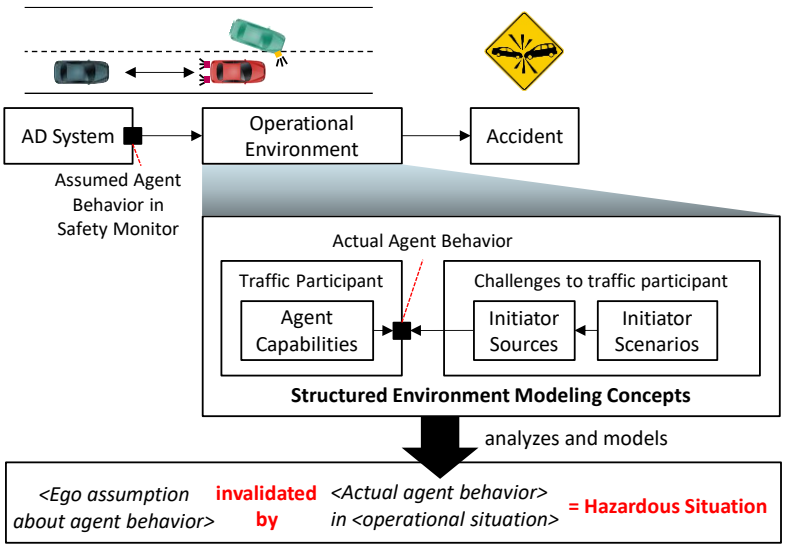


Figure 4.1 The Structured Environment Modeling (SEM) concept enables a detailed analysis of the operational environment relevant to risks originating from the behavior of other traffic participants. The model uses initiator sources, initiator scenarios, and agent capabilities as conceptual elements to express potential causes of critical behavior relevant to assumptions the ego system makes regarding that behavior.

The Structured Environment Model (SEM) supports a detailed analysis of the operational environment that matters for risks driven by other traffic participants' behavior. It is used when a safety monitor relies on assumptions about that behavior and when those assumptions may be relaxed in specific situations. Figure 4.1 summarizes the SEM concept.

The SEM refines the operational-situation part of a hazardous situation of the form "Actual agent behavior in an operational situation class critically exceeds an ego assumption." It introduces three modeling elements:

- **Initiator sources** are external sources in the environment that can trigger the agent's behavior in the given operational situation class. Examples include a driver command, an environmental influence, or a vehicle fault.
- **Initiator scenarios** are concrete situations in which a given initiator source becomes active. Examples include a lane ending due to a construction site or a reaction to a slow leading truck in dense traffic. Initiator scenarios add the external context needed to reason about risk.
- **Agent capabilities** describe what the agent must be able to do to handle the initiator scenario safely. Examples include focusing on a region, perceiving cues, understanding the challenge and response options, deciding, and acting.

Together, these elements form a SEM instance that is specific to one hazardous situation. The SEM focuses the analysis on explicit preconditions and capability demands that can be reviewed and checked. It also provides trace links that support later coverage arguments and the CBCT construction in Section 4.2.

4.1.2 Engineering method

This subsection defines a three-step SEM construction procedure for a given hazardous situation within a declared ODD. The goal is to produce a reviewable SEM instance that makes the activation context of risk-relevant agent behavior explicit and preserves trace links back to the originating HARA item.

The SEM method uses three inputs: the hazardous situation ("Actual agent behavior in an operational situation class critically exceeds an ego assumption"), the declared ODD, and an environment ontology selected for that ODD. The method produces three outputs: a set of initiator sources, a set of initiator scenarios per source (each tagged with ontology elements and a short rationale), and a scenario-specific capability model, all linked to the original hazardous situation.

Step 1 Behavior Initiator Source Analysis

Step 1 identifies possible sources that can trigger the risk-relevant agent behavior so that the later scenario analysis is less likely to overlook activation mechanisms or to focus only on human drivers. In principle, engineers could start directly with Step 2. During the SINADRA trials, however, we observed that informal scenario enumeration tends to over-focus on driver-initiated behavior because these cases are common in practice. Step 1 therefore asks engineers to enumerate initiator sources first. This creates separate starting points for Step 2 and reduces the risk that entire source categories are missed.

Example. Vehicle deceleration can be triggered by several initiator sources:

- the agent/driver, for example an intentional reaction to a slow front vehicle.
- the *vehicle system* itself, for example due to a flat tire.
- an *environmental condition*, for example strong headwind.
- a *static object*, for example due to hitting a fallen tree in the lane.
- the *road structure*, for example due to a steep downhill or uphill section.

The same influence-source thinking can also be applied to behaviors beyond kinematics by asking which sources can create unsafe energy transfer that matters for the hazardous situation. ISO 12100 (International Organization for Standardization, 2010) provides background on hazard sources and energy-based hazard thinking.

Step 2 Initiator Scenario Analysis

Step 2 refines each initiator source into concrete initiator scenarios by combining it with items from an environment ontology. The result is a traceable set of external contexts in which the hazardous situation can arise.

Initiator sources alone do not specify when and why they become active. This activation context is needed to understand how critical agent behavior can arise in the hazardous situation and to assess its plausibility. Step 2 therefore derives initiator scenarios for each initiator source. Each initiator scenario describes a distinct environmental context in which the source may be activated and the assumed behavior may be exceeded.

For example, agent-initiated deceleration may occur in scenarios such as: (1) braking due to a traffic signal change, (2) braking to avoid vehicle damage due to a detected pothole, or (3) reacting to the end of a traffic jam or stop-and-go traffic. Environment-initiated deceleration may occur in scenarios such as: (1) strong wind that generates additional deceleration forces, (2) a sudden change in road friction, or (3) driving into deep water leading to loss of tire-road adhesion.

Table 4.1 Example results of the initiator scenario analysis for the hazardous situation “Front vehicle braking in highway car-following exceeds the ego assumption.”

Hazardous Situation: Front vehicle braking in highway car-following exceeds the ego assumption
Guide Question: How can <ontology item> activate <initiator source> to cause <hazardous situation>?

Initiator Source initiates FV braking force	**Ontology Item**	**Interpretation / Initiator Scenarios**
Driver	L1: Speed Limit	IS1: Driver adapts to lower speed limit IS2: Driver brakes abruptly to avoid getting a speeding ticket
	L2: Parking Spaces	N/A, no parking spaces immediately present on highways
	L3: Construction Zone	IS3: Driver intentionally brakes to switch lanes due to end of lane before construction zone
	L4: Emergency Vehicle	IS4: Driver gives way to approaching emergency vehicle from behind.
	L5: Sun Glare	IS5: Driver reacts to lack of situational awareness due to sun glare
Environment Condition	… L5: Hydroplaning …	IS6: Hydroplaning causes braking force
Object	… L4: Emergency Vehicle …	IS7: Physical contact with static emergency vehicle causes braking force
Road Structure	L1: Pothole …	IS8: Pothole causes braking force
Vehicle System	… L4: Tire …	IS9: Tire blowout causes braking force

A more extensive list of example initiator scenarios is documented in Appendix A.1.

Ontology-guided initiator scenario identification. In realistic ODDs, many distinct contexts can activate the same initiator source. Without structured guidance, it is difficult to justify that the scenario space has been searched systematically. Step 2 therefore uses an environment ontology as a checklist-like structure. For each initiator source, the analyst traverses the ontology items and, for every source–item pair, answers: "How could this item contribute to activating the source in the given hazardous situation?" If the contribution is plausible, an initiator scenario is documented together with a short rationale and an ontology tag. If not, the pair is documented as not applicable (N/A) with a brief justification. This yields both a scenario list and an explicit record of rejected combinations, supporting review and later updates. SINADRA does not require a specific ontology, but assumes that the chosen ontology is representative of the target operational domain and the declared ODD.

As a concrete illustration, the Automotive Urban Traffic Ontology (A.U.T.O.) (Westhofen, 2025) builds on the six-layer model (Scholtes et al., 2021) and structures traffic situations using items from six layers: road network, roadside structures, temporary changes, dynamic objects, environmental conditions, and data and communication. Figure 4.2 shows the six-layer model and selected ontology items that can serve as entry points for the analysis.

Table 4.1 shows example results of the initiator scenario analysis for the hazardous situation "Front vehicle braking in highway car-following exceeds the ego assumption." For selected initiator sources from Step 1 and selected ontology items, the table records the derived initiator scenarios together with a short rationale and tag, and also shows example N/A entries.

The following observations highlight the value of using an ontology in this step:

- The same ontology item (for example, a speed limit sign) leads to different initiator scenarios for the same source (see IS1 vs. IS2 in Table 4.1), with different expected deceleration behavior in the hazardous situation.
- The same ontology item can yield different initiator scenarios across different initiator sources (for example, the emergency vehicle item in IS4 vs. IS7), with different expected frequencies and consequences within the ODD.
- Explicit N/A entries (for example, the driver–parking-space combination in Table 4.1) document why certain source–item combinations are considered implausible and support review and later change impact analysis.

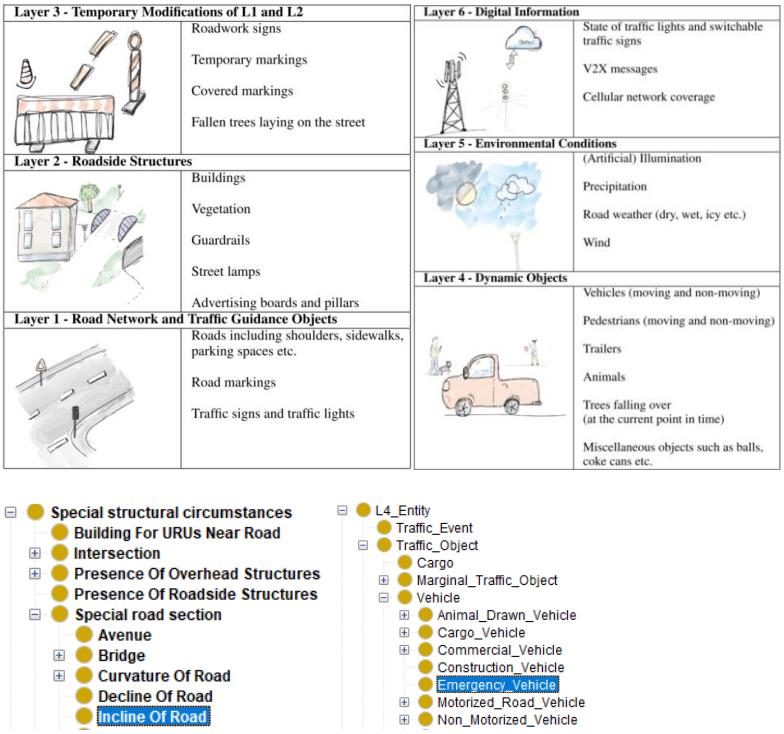


Figure 4.2 Top: Six-layer model for structuring the operational environment, reproduced from Table 1 in (Scholtes et al., 2021) under CC BY 4.0. Bottom: Selected A.U.T.O. class hierarchies from Layers 1 and 4, displayed in Protégé; author-created screenshot based on A.U.T.O. v2.0.0, © 2022 Lukas Westhofen, MIT License.

Step 3 Agent Capability Analysis

Step 3 completes the SEM by adding the agent-internal view: what the agent must be able to do to handle each initiator scenario without exhibiting the critical behavior relevant to the hazardous situation.

Step 2 makes explicit *when* an initiator source can be activated by describing initiator scenarios. Step 3 complements this by describing *how* the agent is expected to cope with these scenarios. The output is a scenario-specific capability model that lists required capability items grouped by capability type and, where useful, a coarse ordering that supports later CBCT construction.

To structure this elicitation, SINADRA uses the behavior model in Figure 4.3, which distinguishes the capability types Focus, Perceive, Understand Challenge, Understand Reaction Options, Decide, and Act. The model is intentionally simple: it is used as an engineering checklist for capability elicitation and later causal modeling, not as a detailed cognitive theory.

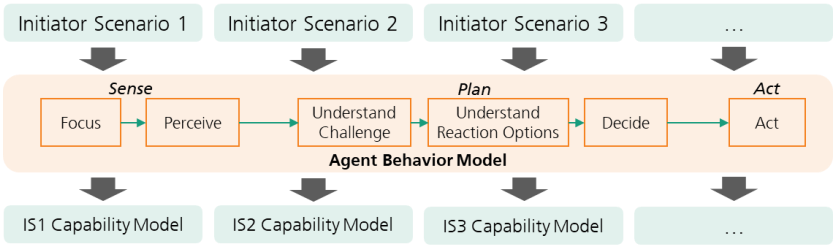


Figure 4.3 Agent behavior model for structuring the agent capability analysis step.

Guide questions. Engineers take an initiator scenario as context and answer:

- **Focus:** What must the agent attend to (e.g., visual/auditory cues, regions of interest)?
- **Perceive:** What must the agent detect or estimate?
- **Understand Challenge:** What must the agent infer about the situation's demand?
- **Understand Reaction Options:** What options must the agent consider as feasible and allowed?
- **Decide:** What decision must the agent make (including timing and strength)?
- **Act:** What must the agent execute reliably?

Answering these questions yields a scenario-specific capability model: a list of required capabilities per type and, where useful, an indicative ordering (e.g., Perceive before Decide before Act). This structure is later used as a scaffold when CBCTs model how capability deviations can contribute to critical behavior.

Example. The guide questions are illustrated for the initiator scenario IS4 "Driver gives way to an approaching emergency vehicle from the rear" in the context of the hazardous situation "Front vehicle braking in highway car-following exceeds the ego assumption."

- **Focus:** The driver needs to allocate visual and auditory attention to the rear area of the vehicle to notice siren and lights.
- **Perceive:** The driver needs to perceive the presence, distance, and approach direction of the emergency vehicle, as well as the surrounding traffic and available gaps.
- **Understand Challenge:** The driver needs to recognize the obligation to yield and to assess how urgent a reaction is, given the speed and proximity of the emergency vehicle and the current traffic state.
- **Understand Reaction Options:** The driver needs to understand which maneuvers are allowed and feasible in the situation, for example, controlled braking and changing lanes or moving to the shoulder.

- **Decide:** The driver needs to decide when and how strongly to brake and whether a lane change is needed, balancing the need to clear a path for the emergency vehicle with the risk of causing a rear-end collision.
- **Act:** The driver needs to carry out the chosen maneuver, for example, braking and changing lanes in a controlled way and signaling intentions to other road users.

Figure 4.4 models the capabilities for this initiator scenario in a capability model.

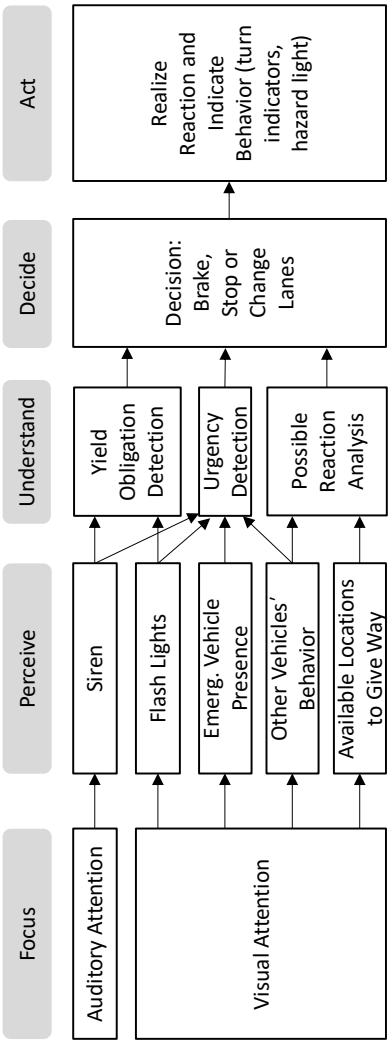


Figure 4.4 Agent capability model for initiator scenario "Driver gives way to an approaching emergency vehicle from the rear".

Qualitative SEM quality checks for review support

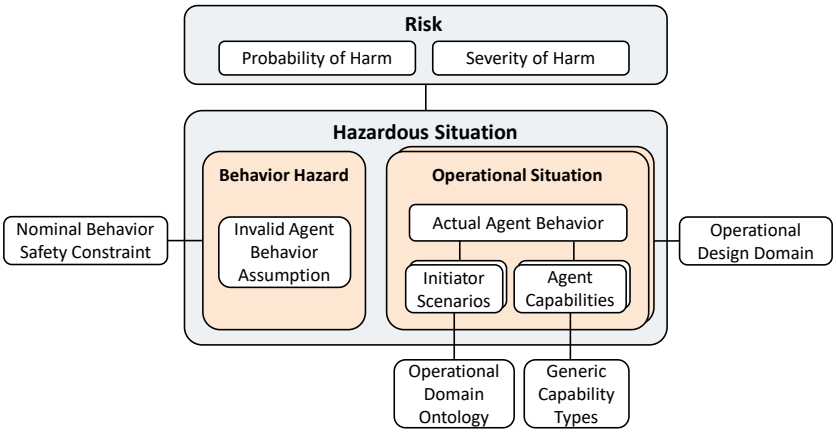


Figure 4.5 Conceptual traceability between HARA context and the structured environment model elements.

The SEM method refines a HARA hazardous situation into one SEM instance with three element types: initiator sources, initiator scenarios, and scenario-specific agent capabilities. Each SEM element is traced back to the originating HARA item (Figure 4.5). This trace structure supports later CBCT construction and later derivation of monitorable low-risk hypotheses by making scoping decisions explicit and reviewable. A concept and metamodel showing how safety-driven environment representation can be technically established as a basis for verification or quality checks is introduced in (Reich et al., 2023).

The SEM quality depends less on a specific notation and more on whether the model is a sound, reviewable basis for later CBCT modeling and for deriving runtime hypotheses. In SINADRA, a SEM is fit for purpose if it:

(1) captures the environment elements that can influence the hazardous-situation relevance or risk in scope,
(2) makes actor roles and task context explicit,
(3) defines initiator scenarios that bound the analysis scope, and
(4) links each scenario to the agent capabilities assumed for safe behavior in the given context.

Because SEM is a structured, domain-specific model, verification is treated as qualitative review support. The aim is to surface missing concepts, unclear scoping, and hidden assumptions early. The checks below provide practical guidance without implying full completeness or fully verified SEMs.

Ontology representativeness is a first check. The environment ontology (as a controlled vocabulary) should represent the entities, relations, and attributes needed to state the key context factors relevant to the hazardous situation.

Reviewers can sample related HARAs in the same application domain and confirm that the relevant context factors can be expressed using ontology items. If the analysis requires ad-hoc terms, either refine the ontology or tighten the modeled scope. Initial ideas have been proposed to tailor base ontologies to achieve coverage in the context of concrete safety analysis processes (Hillen & Reich, 2023; Hillen et al., 2025) based on a formal integration between environment and safety engineering models.

Note that SINADRA is ontology-agnostic: Analysts may use any operational environment ontology that suits a HARA in the declared ODD. Examples are ISO 34503 (International Organization for Standardization, 2023), the Automotive Urban Traffic Ontology A.U.T.O. (Westhofen, 2025), the VDA 702 situation catalog 2.0 (German Association of the Automotive Industry, 2023), or the Operational World Model Ontology (Czarnecki, 2018a, 2018b). We assume that a representative ontology is available.

Initiator Scenario–Ontology Coverage is a second check. For each initiator scenario, list the ontology items used to specify its context (actors, regions, objects, relations, constraints). Then check (a) whether any scenario statement relies on a concept not represented in the ontology, and (b) whether key ontology concepts remain unused across all scenarios. The point is not to maximize item counts, but to make scoping gaps visible and to keep scenario descriptions consistent in wording and meaning.

Scenario–Capability Coverage is a third check. For each initiator scenario, record the capabilities assumed for safe interaction (sense/notice, understand, decide, act, at the chosen granularity). Then confirm two links: every capability is motivated by at least one scenario element, and every scenario element that can plausibly drive unsafe behavior is linked to at least one capability whose deviation would matter. The output can be a table or checklist used in reviews and later reused as input for CBCT modeling.

These checks are lessons learned from the execution of the warehouse case study (Section 5.1), where SEM quality is argued via ontology- and task-driven scoping, explicit initiator scenarios, and scenario-specific capabilities reused in CBCT modeling.

4.2 Critical Behavior Cause Trees (CBCT)

Section 4.1 introduced the Structured Environment Model (SEM) as a structured description of the operational environment and other traffic participants' behavior for a given hazardous situation, together with qualitative checks that support review of scope and trace links. This section introduces Critical Behavior Cause Trees (CBCTs) as the second SINADRA method element. CBCTs reuse the initiator sources, initiator scenarios, and capability structure from the SEM to model causal chains that can lead to

critical behavior. They also integrate risk implausibility hypotheses (RIH) and other safety concept measures into a single Boolean cause–effect model that supports sufficiency checks within the modeled scope.

This section defines the CBCT concepts and Boolean semantics (Section 4.2.1), and presents a four-step procedure to construct CBCTs from a given SEM and safety concept (Section 4.2.2).

4.2.1 Concept and formal model definition

This subsection introduces the CBCT concept and formal model. It defines the core element types used in CBCTs and explains the Boolean semantics that support a reviewable sufficiency argument for risk implausibility hypotheses within the modeled scope.

SINADRA aims to relax safety-constraint parameters at runtime in low-risk situations and increase permissiveness without introducing unargued risk. To do so, the method must make explicit when critical agent behavior can arise in the modeled initiator scenarios and which hypotheses or safety concept measures prevent the associated risk paths. CBCTs provide this on top of the SEM by introducing two element groups (Figure 4.6).

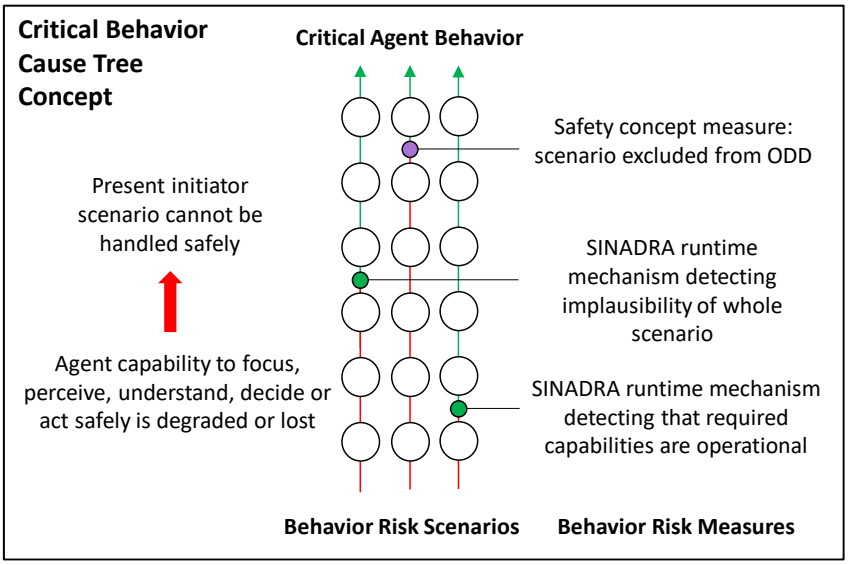


Figure 4.6 Critical Behavior Cause Trees represent behavior risk scenarios together with behavior risk measures, so that reviewers can check whether the modeled risk paths are blocked by at least one measure within the declared scope.

The SEM bounds the analysis by making the relevant initiator scenarios and agent capabilities explicit for a given hazardous situation. The CBCT then refines this scope into **behavior risk scenarios**, which describe causal paths

from situational influences to capability deviations and finally to the critical behavior that invalidates an ego assumption.

Figure 4.6 shows this idea at a high level. CBCTs represent behavior risk scenarios together with behavior risk measures, so that reviewers can check whether the modeled risk paths are blocked by at least one measure within the declared scope.

Behavior risk measures capture how the causal paths to critical behavior are blocked or constrained. In SINADRA, two measure types are relevant:

(1) **Risk implausibility hypotheses (RIHs)**, which are monitorable conditions that indicate that a behavior risk scenario is not applicable in the current situation or that worst-case risk is not plausible within the argued scope. Section 4.2 defines and justifies these hypotheses at design time. Section 4.3 covers their runtime realization.

(2) **Other safety concept measures**, such as ODD restrictions, design constraints on behavior, additional monitors, and fallback functions.

CBCTs combine both types in one Boolean model. This is necessary because RIHs alone do not cover all relevant causal paths, and any permissiveness increase must be argued in the context of the full safety concept.

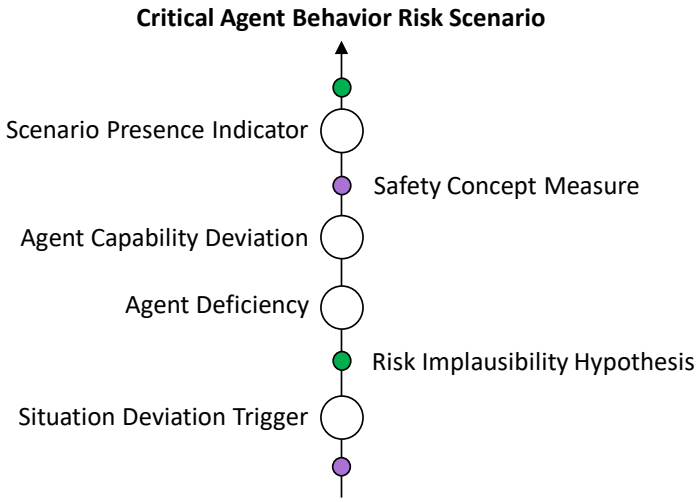


Figure 4.7 Typed elements representing the causal structure of behavior risk scenarios (left) and behavior risk measures in CBCTs.

Before defining the CBCT syntax, we introduce the typed elements used to model behavior risk scenarios and behavior risk measures (Figure 4.7). Using explicit element types supports a structured construction of CBCTs from a

SEM, separates SINADRA hypotheses from other safety concept measures, and enables later review checks on scope and completeness.

CBCTs use six element types:

- **Scenario Presence Indicators (PI)** state the conditions under which an initiator scenario from the SEM is present. If a PI is false, the corresponding scenario branch is inactive in the CBCT.
- **Agent Capability Deviations (CD)** represent deviations of scenario-specific agent capabilities that contribute to the critical behavior. They form the central internal nodes of behavior risk scenarios.
- **Agent Deficiencies (AD)** represent internal shortcomings that can cause capability deviations, such as distraction, missing knowledge, or reduced performance in a specific capability.
- **Situation Deviation Triggers (SDT)** represent external influences that trigger or amplify an agent deficiency in a scenario, such as occlusions, weather, or unusual traffic constellations.
- **Risk Implausibility Hypotheses (RIH)** represent monitorable conditions that, when satisfied, block selected causal paths or rule out worst-case conditions within the modeled scope, thereby allowing the associated assumption relaxation to be considered.
- **Safety Concept Measures (M)** represent other measures defined in the system safety concept, such as ODD restrictions, design constraints, additional monitors, or fallback strategies, that block or mitigate behavior risk scenarios.

Figure 4.8 illustrates the extensions to the overall SINADRA concept provided by CBCTs. Behavior risk scenarios concretize the operational situations in which the risks associated with the hazardous situation are plausible. Behavior risk measures consisting of risk implausibility hypotheses aligned with safety concept measures indicate the implausibility of risk scenarios and enable the permissiveness-increasing relaxation of safety constraints.

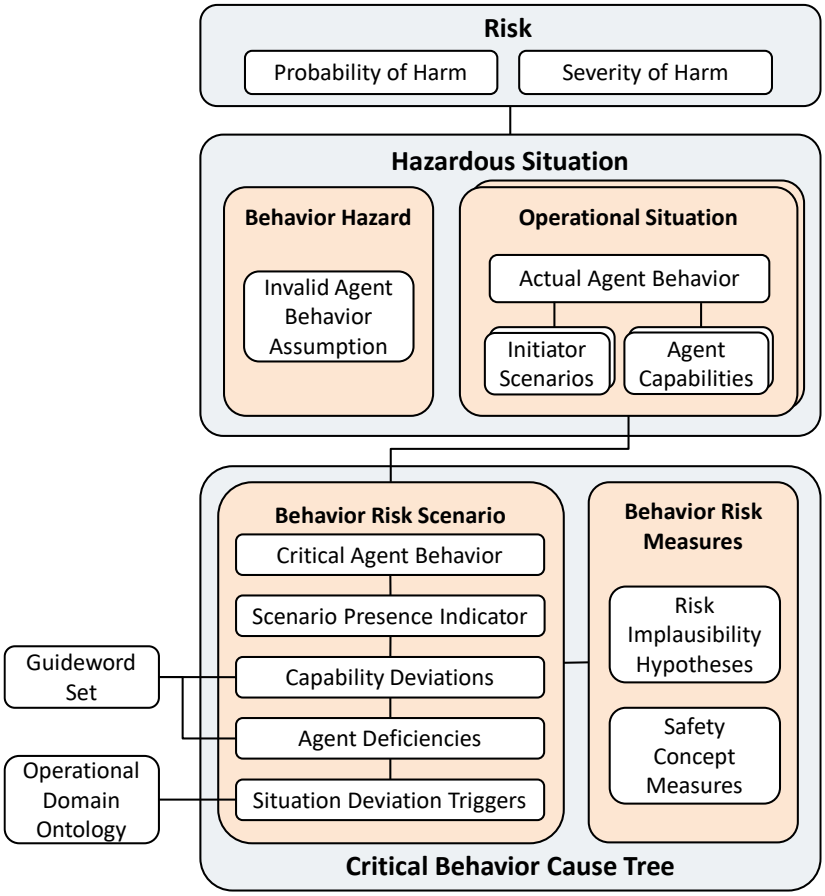


Figure 4.8 Conceptual traceability between behavior risk scenarios and behavior risk measures as the essential contents of critical behavior cause trees and the Structured Environment Model concretizing the elements of operational situations in the HARA.

Formalization foundation and mathematical specification

CBCTs are grounded in Boolean logic and use a deductive, top-down analysis style in the tradition of Fault Tree Analysis and related methods. This is suitable for SINADRA because the goal is to show that all causal combinations leading to critical agent behavior are either blocked by safety measures or rendered implausible by risk implausibility hypotheses. Starting from the critical behavior in a given initiator scenario and tracing backward to capabilities and situational influences yields a closed set of behavior risk scenarios that can be checked algorithmically.

CBCTs are architected to mirror the SEM: initiator scenarios and scenario-specific capabilities induce modular subtrees and typed elements. This supports systematic model population and SEM–CBCT alignment checks. The

modular structuring is inspired by Component-Integrated Fault Trees (Domis, 2012), adapted from technical fault propagation to behavior risk reasoning.

Related approaches model environmental contributions differently (for example Environment Fault Trees (Kramer et al., 2020), Component Fault and Deficiency Trees (Zeller, 2022), or Fuzzy Cause Trees (Herd et al., 2025)), typically focusing on how environmental conditions affect ego-system capability. CBCTs instead focus on behavior of other agents under situational influences, because these behavior deviations are what invalidate the environmental assumptions embedded in runtime safety constraints.

We now proceed to the formal CBCT specification. A *Critical Behavior Cause Tree* is the tuple

$$CBCT = (Sub, ODM, IDM, PI, AD, SDT, RIH, M, G, E)$$

with the following finite sets:

- $\boldsymbol{Sub}$ is a set of sub-trees. Each $subCBCT \in Sub$ is a triple $subCBCT = (IN, OUT, cbct_j)$ where $IN \subseteq IDM$, $OUT \subseteq ODM$, and $cbct_j$ is a reference to another CBCT that is used as a module.
- $\boldsymbol{ODM}$ is a set of output deviation modes. It contains the top-level critical agent behavior odm_{top} and, if needed, further intermediate deviation modes for capabilities.
- $\boldsymbol{IDM}$ is a set of input deviation modes that serve as interface inputs to CBCTs and subCBCTs.
- $\boldsymbol{PI}$ is a set of scenario presence indicators. Each element $pi \in PI$ encodes the presence of one initiator scenario from the SEM.
- $\boldsymbol{AD}$ is a set of agent deficiencies. Each $ad \in AD$ describes an internal shortcoming that contributes to a capability deviation.
- $\boldsymbol{SDT}$ is a set of situation deviation triggers. Each $sdt \in SDT$ describes an external influence that contributes to a capability deviation.
- $\boldsymbol{RIH}$ is a set of risk implausibility hypotheses. Each $rih \in RIH$ represents a condition that, if true at runtime, is intended to block one or more behavior risk scenarios.
- $\boldsymbol{M}$ is a set of safety concept measures. Each $m \in M$ represents a measure from the safety concept that blocks or mitigates behavior risk scenarios.
- $\boldsymbol{G}$ is a set of Boolean gates. Each gate $g \in G$ has one output $g.out$, one or more inputs $g.in_{\mathrm{i}}$, and an associated Boolean transfer function over its inputs.
- $\boldsymbol{E}$ is a set of directed edges that connect elements. Edges run from basic events (PI, AD, SDT, RIH, M), gate outputs, input deviation modes, or subCBCT outputs to gate inputs, ODMs, or subCBCT inputs. This wiring induces the Boolean structure of the CBCT.

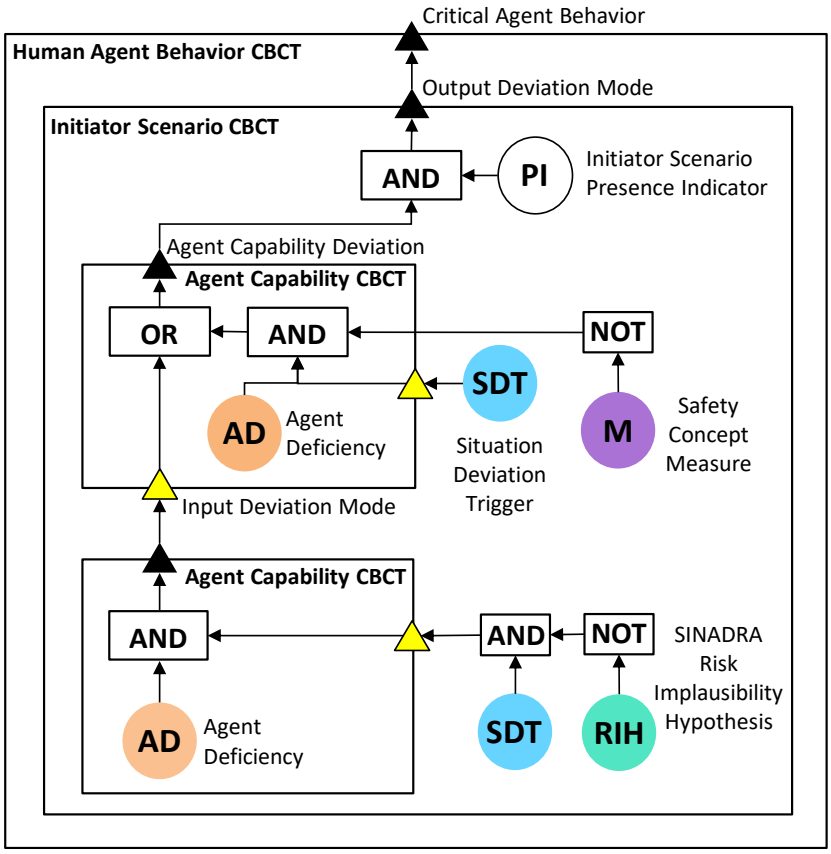


Figure 4.9 Illustration of modeling elements of Critical Behavior Cause Trees (CBCT).

Figure 4.9 illustrates these modeling elements in a typical generic usage setting, where CBCTs are instantiated to analyze the causal structure of critical agent behavior. With the CBCT elements and Boolean structure defined, the next subsection will explain how to construct CBCTs from a given SEM and safety concept.

4.2.2 Engineering method

Building on the CBCT concept in Section 4.2.1, this subsection presents a four-step procedure for constructing CBCTs from a given SEM and system safety concept. The procedure instantiates the CBCT element types as a concrete model: it captures behavior risk scenarios, integrates SINADRA risk implausibility hypotheses (RIHs) and other safety concept measures, and produces the structure required for the sufficiency checks in Section 4.2.2.

The CBCT engineering method proceeds in four steps:

(1) Create CBCT analysis structure based on the SEM
(2) Identify and model behavior risk scenarios
(3) Add risk implausibility hypotheses and other safety concept measures
(4) Justify sufficiency of RIHs and measures to control the modeled risk scenarios in the context of the safety concept

Running example. Section 3.3 used an ego–pedestrian crosswalk scene. Here, we use a closely related case at a signalized intersection to show that the CBCT procedure applies across hazards that share the same interaction pattern but differ in the agent under analysis. The running example spans Steps (1)–(4) and considers an urban approach to a green, signalized intersection with a crosswalk. The system-level Output Deviation Mode (ODM) is "Front vehicle braking exceeds the ego assumption." The ODD slice covers urban roads with moderate speeds and approaches to signalized intersections within stopping distance, in day or night and under dry or wet conditions (snow and ice excluded).

While Section 3.3 used an ego–pedestrian crosswalk scenario, this section uses a related situation with a front vehicle braking at a signalized intersection with a crosswalk. This highlights that the CBCT method applies across different but structurally similar hazards. To illustrate the method, we use a running example across Steps 1-4: an urban approach to a green, signalized intersection. The system-level Output Deviation Mode (ODM) is "Front vehicle braking exceeds the ego assumption." The ODD slice covers urban roads with moderate speeds, approaches to signalized intersections within stopping distance, and day and night in dry or wet conditions.

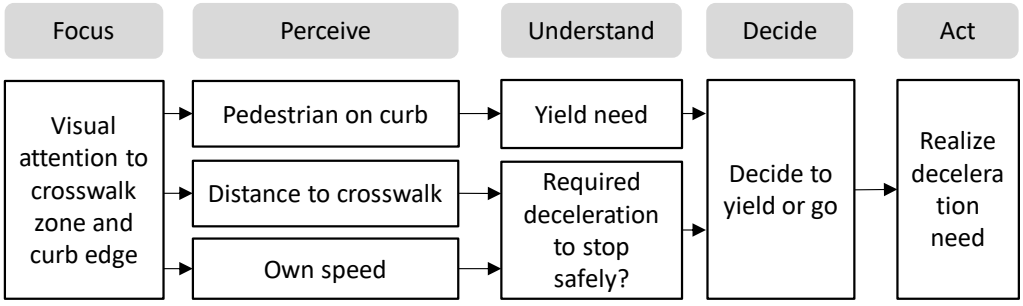


Figure 4.10 Example capability model for initiator scenario "Driver reacts to pedestrian entering the crosswalk".

We focus on the initiator scenario "Front vehicle driver reacts to a pedestrian entering the crosswalk." This mirrors the crosswalk setting from Section 3.3, but treats the front vehicle's human driver as the agent whose behavior is analyzed. Figure 4.10 shows the capability model used as the structural backbone of the CBCT in Step (1). In the subsequent steps, capability

deviations (CD) and their causes, agent deficiencies (AD) and situation deviation triggers (SDT), are attached to these capabilities.

Step 1 Create CBCT analysis model structure based on SEM

Goal. Create a CBCT structure that is aligned with the SEM so that later modeling decisions remain scenario-scoped and traceable (Figure 4.11).

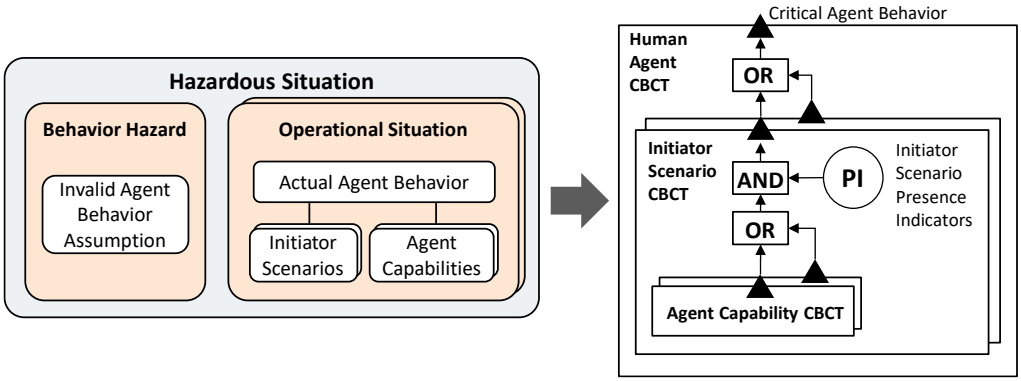


Figure 4.11 CBCT modular top-level structure derived from SEM.

Construction rules.

(1) **Fix the top-level ODM.** Define the critical agent behavior as the CBCT's top event (*Output Deviation Mode*). This is the scope anchor for the tree.

(2) **Decompose by initiator scenarios.** For each SEM initiator scenario, create one subCBCT under an OR gate. Assign a scenario-scoped ODM as the local root of that subCBCT. Add the scenario's *Presence Indicators* (PI) as enabling conditions for that branch.

(3) **Decompose by capabilities.** Within each initiator scenario subCBCT, create one capability *subCBCT* per SEM capability. The output of each capability subCBCT is reserved for the capability deviation that will be instantiated in Step 2.

Output of Step 1. A CBCT "skeleton" consisting of scenario branches and capability subtrees, with PIs attached, but without deviations, deficiencies, triggers, hypotheses, or measures yet.

Running example. The top ODM is "Front vehicle braking exceeds the ego assumption." Figure 4.12 instantiates initiator scenario IS1 "Driver reacts to a pedestrian entering the crosswalk" as a dedicated branch (ODM-IS1). Presence Indicators such as "Marked crosswalk ahead" and "Pedestrian present or about to enter" activate this branch. Inside IS1, the capability chain (Focus, Perceive, Understand, Decide, Act) is instantiated as capability

subCBCT placeholders. Further initiator scenarios can be added in the same pattern. The following steps focus on IS1.

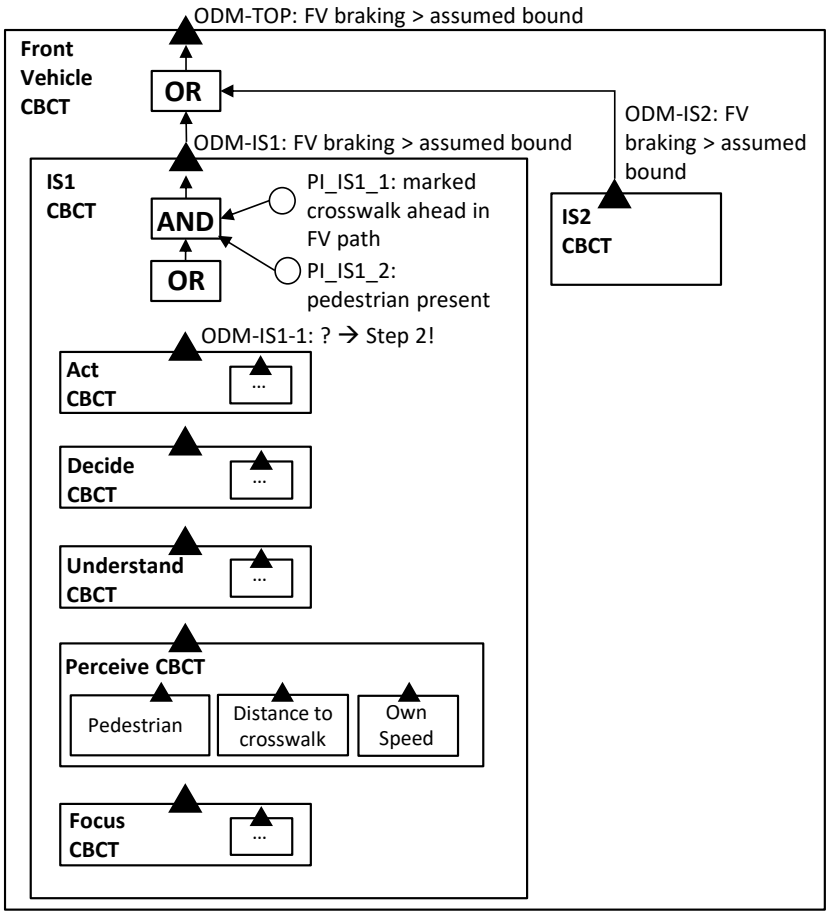


Figure 4.12 CBCT structure derived from the SEM in the running example.

Step 2 Model behavior risk scenarios: Capability deviations, agent deficiencies, and situation deviation triggers.

Goal. Populate each initiator-scenario branch of the CBCT with the causal elements that can produce the scenario-scoped critical behavior, so that the CBCT encodes explicit *behavior risk scenarios* that can later be blocked by hypotheses and measures (Figure 4.13).

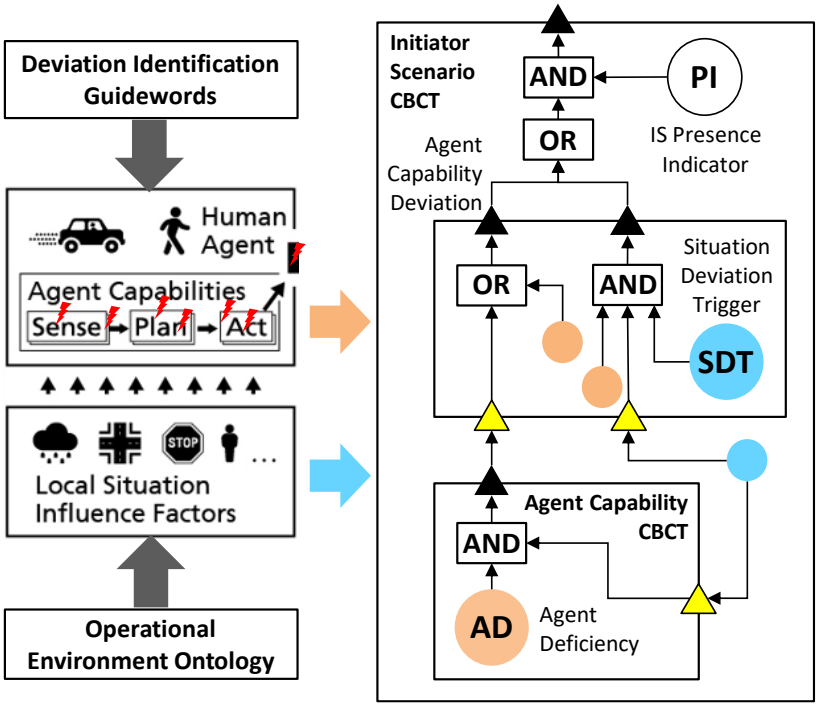


Figure 4.13 Structure of an initiator scenario CBCT. The causal structure of capability deviations consists of agent deficiencies combined with situation deviation triggers that cause deviations to occur or propagate. To identify deviations and deficiencies, HAZOP-inspired guide words are used as guidance. To identify situation deviation triggers, an operational environment ontology is used as guidance.

Construction rules. For each initiator scenario subCBCT created in Step 1:

(1) **Create capability deviations (CD).** For each capability subCBCT (Focus, Perceive, Understand, Decide, Act), define one or more Capability Deviations (CD) that would contribute to the scenario-scoped ODM if they occur (for example: “detected too late,” “misjudged,” “delayed decision,” “incorrect actuation”). To keep this repeatable and reduce interpretation effort, SINADRA uses a capability-specific guideword set (Table 4.2) rather than generic HAZOP or generic failure-type lists. For each capability, the engineer walks through the relevant guidewords and records the candidate deviations they suggest for the scenario.

Table 4.2 Concretized HAZOP guideword set to support the identification of capability deviations and agent deficiencies. Full set with examples in Appendix A.2.

Capability Type	Guidewords
Focus	Missed Focus, Distracted, Overfocused, Impaired Focus, Fluctuating Focus, Delayed Focus
Perceive	No, False, Delayed, Partial Perception
Understand Challenge	Misunderstood, Under/Overestimated, Unassessed Challenge
Understand Reaction	Inaccurate assessment, Ignorance of limits, Physical Impairment, Cognitive Overlead, Misjudged feedback
Decide	Indecisive, Incorrect Decision, Impulsive, Delayed Decision
Act	Incorrect Action, Delayed Action, Overreaction, Partial Action

(2) **Attach agent deficiencies (AD) using guidewords.** For each CD, identify one or more Agent Deficiencies (AD) that can cause that deviation in the initiator scenario context.

(3) **Add situation deviation triggers (SDT) via ontology traversal.** For each AD, identify external conditions that can trigger or worsen it in the current initiator scenario. These are modeled as Situation Deviation Triggers (SDT). SDTs are identified by reusing the same operational environment ontology family used in SEM (Step 2 of SEM), so that "what can vary in the environment" is handled with a controlled vocabulary and can be reviewed. The micro-procedure for SDT identification has three steps:

(3.1) Select a capability deviation and its associated agent deficiency in the current subCBCT

(3.2) Traverse ontology items layer by layer and ask, for each item, how it could contribute to this deviation and deficiency in the given initiator scenario

(3.3) For each plausible contribution, create an SDT node and connect it to the deficiency with an AND or OR gate, depending on how the factors combine.

(4) **Wire the Boolean causal structure.** For each capability subCBCT, connect CD to its causes (AD combined with SDTs) using AND/OR gates that reflect the intended logic ("AD must hold and at least one SDT holds," or "either AD1 or AD2 is sufficient," etc.). At the initiator-scenario level, connect the scenario-scoped ODM to the relevant CDs (often via OR), and include the Presence Indicators (PI) as enabling conditions.

Output of Step 2. A populated initiator-scenario CBCT in which the scenario-scoped ODM is derived from explicit Boolean combinations of PI, CD, AD, and SDT nodes. This structure is the basis for enumerating behavior risk scenarios and placing hypotheses and measures in Step 3.

Running example (IS1: "Driver reacts to a pedestrian entering the crosswalk"). The capability subCBCTs are populated with CDs, ADs, and SDTs. Two illustrative behavior risk scenarios are identified:

Scenario 1 "Hard braking due to late pedestrian detection"

The driver detects the pedestrian too late, so braking starts close to the crosswalk (short-margin stop). The capability "Perceive pedestrian on curb" deviates.

- **CD_P1:** Pedestrian detected too late
- **AD_P1:** Limited crosswalk attention
- **SDT_P1a:** Near-side occluder (parked bus or van/A-pillar)
- **SDT_P1b:** Oncoming vehicle in opposite direction (focus on that vehicle)
- **Wiring:** CD_P1 = AD_P1 ∧ (SDT_P1a ∨ SDT_P1b)
- **How SINADRA finds these:** Use Perceive—Delayed/No to elicit CD_P1; traverse the environment ontology to identify SDT_P1a/b (ontology items: 1. occluders → SDT_P1a, 2. adjacent lane → SDT_P1b).

Scenario 2 "Hard braking due to misjudged yield need"

The driver sees the pedestrian but misjudges how urgent yielding is, delaying the stop decision. The capability "Understand yield need" deviates.

- **CD_U1:** Yield need misjudged as go
- **AD_U1:** Misappraisal of yield urgency (rule/kinematics gap)
- **SDT_U1a:** Pedestrian hand gesture suggests giving priority, but still steps on street
- **SDT_U1b:** Curb-entry cue (weight shift/jogging speed)
- **Wiring:** CD_U1 = AD_U1 ∧ (SDT_U1a ∨ SDT_U1b)
- **How SINADRA finds these:** Apply an Understand guideword such as Underestimated/Misunderstood to identify CD_U1, then query the ontology (Signals/Phasing → hand gesture; Pedestrian Behavior → entry cues) for SDT_U1a/b.

Behavior risk scenario enumeration via minimal cut sets. After Step 2, qualitative minimal cut set analysis can enumerate the distinct minimal combinations that suffice to produce the scenario-scoped ODM and therefore define the baseline behavior risk scenarios that must be blocked in Step 3.

Schematic encoding for IS1:

ODM-IS1 = PI_IS1_1 ∧ PI_IS1_2 ∧ (CD_P1 ∨ CD_U1) with

CD_P1 = AD_P1 ∧ (SDT_P1a ∨ SDT_P1b) and

CD_U1 = AD_U1 ∧ (SDT_U1a ∨ SDT_U1b)

The resulting pre-measure minimal cut sets for IS1 are:

ODM-TOP =
{PI_IS1_1, PI_IS1_2, AD_P1, SDT_P1a}
{PI_IS1_1, PI_IS1_2, AD_P1, SDT_P1b}
{PI_IS1_1, PI_IS1_2, AD_U1, SDT_U1a}
{PI_IS1_1, PI_IS1_2, AD_U1, SDT_U1b}

These sets correspond to “IS1 present” plus either the late-detection path (with one of its SDTs) or the misjudged-yield path (with one of its SDTs). For other initiator scenarios under the same top-level ODM, Step 2 yields analogous minimal combinations that can be used as the entry points for adding risk blockers in Step 3.

Step 3 Model behavior risk measures: risk implausibility hypotheses and safety concept measures

Goal. Extend the Step-2 behavior risk scenarios with explicit *risk blockers* so that the CBCT makes it checkable which combinations leading to the scenario-scoped ODM are ruled out by SINADRA risk implausibility hypotheses (RIH) and/or controlled by other safety concept measures (M) (Figure 4.14).

Construction rules. Step 3 proceeds per initiator scenario subCBCT:

(1) **Use minimal cut sets as placement targets.** The pre-measure minimal cut sets computed in Step 2 define the concrete “risk paths” to be blocked. For each cut set, engineers ask which monitorable condition(s) would invalidate at least one element of the cut set in the given scenario context. Typical patterns are:

(1.1) negate a situation deviation trigger (“occlusion not present,” “visibility good,” “pedestrian motion not toward entry”)
(1.2) assert availability of a capability (“pedestrian detected early enough,” “front area attention not limited”)
(1.3) gate the scenario by a stronger applicability condition (“scenario not active / interaction region not relevant”)

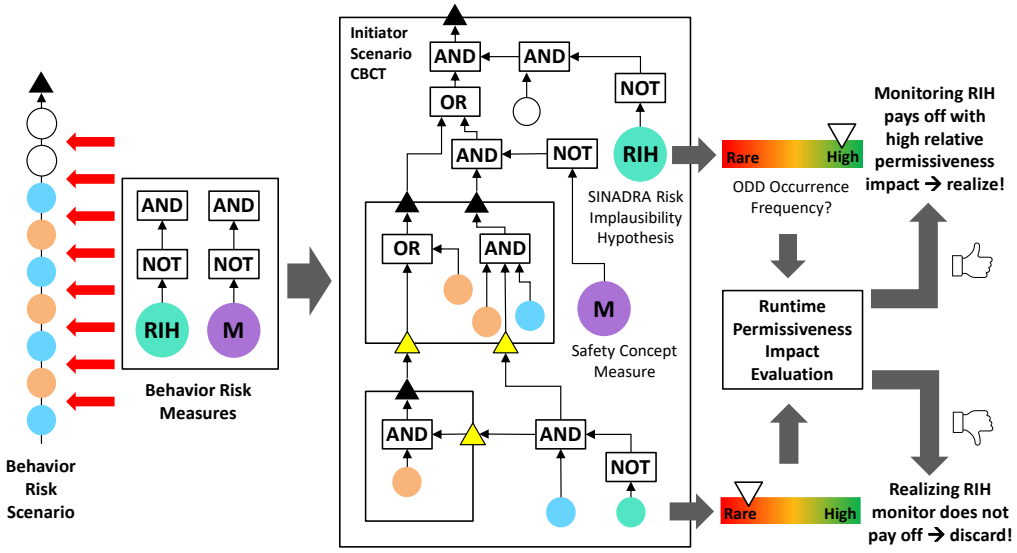


Figure 4.14 Adding risk implausibility hypotheses (RIH) and other safety measures to the CBCT.

(2) **Create risk implausibility hypotheses (RIH).** Each RIH is defined as a *monitorable* condition that, when satisfied at runtime, makes one or more behavior risk scenarios implausible or reduces their risk to a level compatible with a relaxed assumption. RIHs are introduced as typed CBCT nodes and placed where they block the intended causal chain(s).

(3) **Represent blocking as negated conditions in the Boolean model.** Because the CBCT encodes sufficient conditions for the critical behavior, blockers are modeled as negated basic events on the guarded path. Concretely, a guarded node X is typically rewritten as X ∧ ¬RIH_k (or, equivalently, the path is enabled only if the hypothesis is *not* valid). This makes the blocking effect explicit for later cutset recomputation in Step 4.

(4) **Add safety concept measures (M) as additional blockers.** SINADRA assumes a system-level safety concept exists. Measures such as ODD restrictions, behavior constraints, additional monitors, and fallbacks are introduced as typed M nodes. Each measure is placed at the point in the causal structure where it breaks or mitigates a path to the ODM, and it is wired so that its effect appears in the Boolean structure used for analysis.

(5) **Make the division of labor explicit.** Step 3 does not assume that RIHs alone can block all paths: unobservable conditions, limited sensing integrity, or residual corner cases are expected. The CBCT therefore explicitly records which paths are blocked by RIH and which require M measures, so that Step 4 can judge sufficiency of the combined configuration.

Heuristics for selecting candidate RIHs (engineering view). Candidate RIHs are preferred when they (1) block cut sets that occur often enough to matter, (2) are expected to be valid often enough in the ODD to yield real permissiveness gains, and (3) are checkable with the available

observation integrity and timing. The CBCT structure and cut sets provide the concrete basis for weighing these trade-offs. This thesis does not attempt to optimize them exhaustively, which is an interesting future direction though.

Running example (IS1). One candidate hypothesis is RIH-1 (Crosswalk-Clear):

- **RIH-1 definition (intent).** At the decision point before the crosswalk, no pedestrian is on the crosswalk and no pedestrian at the curb shows motion and orientation consistent with entering.
- **Placement.** In the IS1 subCBCT, RIH-1 is placed to guard the SDT nodes that capture occlusion/ambiguity factors around pedestrian presence and entry cues.
- **Boolean representation.** The guarded causal segments are enabled only if the hypothesis is false, modeled by inserting ¬RIH-1 in conjunction with the guarded nodes (so a risk path can only proceed if "Crosswalk-Clear" does *not* hold).

As examples of safety concept measures, consider:

- **M1 (Worst Case on Short Time).** Enforce a conservative deceleration assumption whenever time-to-crosswalk is below a threshold.
- **M2 (Pedestrian Perception Health).** Enforce a conservative assumption whenever perception health for pedestrian features is degraded.

In the CBCT, M1 and M2 are modeled as separate elements that block the path to the output deviation mode whenever their conditions hold, independent of RIH-1. Step 4 then recomputes the minimal cut sets on the augmented tree and checks whether every remaining path to the ODM contains at least one blocker (some ¬RIH_k and/or some M_j).

Step 4 Verify sufficiency of hypotheses to block risk scenarios in the context of the safety concept

Goal. Use the CBCT's Boolean structure to check whether the selected SINADRA risk implausibility hypotheses (RIH) and referenced safety concept measures (M) are sufficient to control the behavior risk scenarios that lead to the critical agent behavior. Step 4 is the method's decision point: it determines whether the intended parameter relaxation is justified *for the analyzed scope*.

Procedure.

(1) **Recompute minimal cut sets on the populated tree.** Starting from the Step-2 CBCT (scenarios only), the tree is re-analyzed after Step 3 has inserted the blockers (negated RIH nodes and M nodes). The output is a new set of minimal

cut sets for each initiator scenario subCBCT (and, if needed, for the overall top ODM).

(2) **Apply the sufficiency rule ("no unblocked cut set").** A relaxation of the assumption linked to the analyzed ODM is justified only if every minimal cut set contains at least one blocker, i.e., at least one ¬RIH_k and/or at least one M_j.

(2.1) If every cut set contains a blocker, then the CBCT makes explicit that the critical behavior can only occur along paths that are controlled by either (a) hypothesis invalidity (so SINADRA would not relax) and/or (b) active safety concept measures.

(2.2) If any cut set contains no blocker, it represents a residual behavior risk scenario that is not controlled by the current configuration. In that case, the intended relaxation is not justified for the scope, and the CBCT must be revised.

(3) **Iterate the configuration when residual cut sets exist.** Residual unblocked cut sets indicate *where* to act:

(3.1) add an additional RIH that blocks (part of) the residual combination,

(3.2) move an existing RIH "upstream" to cover more branches,

(3.3) add or strengthen a safety concept measure, or

(3.4) reduce the intended relaxation scope (tighten applicability conditions / ODD slice).

(3.5) After changes, minimal cut sets are recomputed until the sufficiency rule holds or the relaxation is abandoned.

Running example (IS1). Step 2 yielded four pre-measure minimal cut sets for IS1 ("Driver reacts to a pedestrian entering the crosswalk"). After Step 3, the CBCT contains RIH-1 (Crosswalk-Clear) guarding pedestrian-related SDT nodes and two safety concept measures M1 (Worst Case on Short Time) and M2 (Pedestrian Perception Health). Recomputing minimal cut sets shows that each resulting cut set contains at least one of the blockers ¬RIH-1, M1, or M2. For this initiator scenario scope, this means: the critical behavior can only propagate if RIH-1 is false and the relevant safety concept measures are not in their blocking state. Therefore, the combination of RIH-1, M1, and M2 is sufficient to support the intended relaxation *for IS1*.

Interpretation and acceptance conditions. The sufficiency judgment from Step 4 is accepted under three explicit conditions:

- **Model scope and quality:** the CBCT instance adequately represents the relevant behavior risk scenarios and the placed blockers for the declared scope (supported by the CBCT quality checks described below).
- **Runtime realization of RIHs:** the RIHs are implemented as runtime functions with the integrity needed to infer validity and trigger parameter selection as intended (Section 4.3).

- **Realization of referenced measures:** the safety concept measures referenced as M nodes are implemented and validated within the system-level safety concept.

Under these conditions, the populated CBCT provides a reviewable, HARA-traceable artifact that supports the claim: the selected hypotheses and measures control the relevant behavior risk scenarios sufficiently to justify the proposed relaxation of the corresponding monitor assumption within the analyzed scope.

Qualitative CBCT quality checks for practical review support

CBCT quality depends on whether the tree is a sound, reviewable causal argument for the hazardous situation in scope and whether it supports the later selection of risk implausibility hypotheses (RIHs) and measures. In SINADRA, a CBCT is fit for purpose if it:

(1) is correctly scoped to declared initiator scenarios and its SEM context,
(2) expresses plausible causal paths from situation deviation triggers, agent deficiencies and agent capability deviations to the top output deviation mode,
(3) makes key assumptions explicit as RIH nodes and safety measures, and
(4) yields a structure that can be inspected for sufficiency within the modeled scope (for example via minimal cut set inspection).

Because CBCTs are domain- and scope-specific, verification is treated as qualitative review support. The aim is to make gaps, inconsistencies, and unsupported assumptions visible early, without implying that SINADRA automatically computes full completeness or quantitative quality scores.

SEM–CBCT alignment is a first check. Reviewers confirm that each CBCT is rooted in a SEM initiator scenario and that its main context elements (actors, regions, objects, constraints) are reused consistently. Capability SubCBCTs should correspond to the scenario-specific capability model from SEM, rather than introducing new capability concepts ad hoc. This keeps the CBCT anchored in the declared scope and reduces accidental broadening or unanalyzed capabilities.

Deviation-type coverage via guidewords is a second check. For each capability used in the CBCT, reviewers check that deviation guidewords were applied to confirm that the CBCT considered the main ways the capability can deviate in the given scenario. The goal is not to enumerate everything, but to ensure that the tree does not miss important deviations that dominate risk.

Situation trigger grounding and plausibility is a third check. Situation deviation triggers should be grounded in SEM context and, where available,

in the environment ontology used for SEM scoping. Reviewers check that triggers are stated in observable, domain-meaningful terms, and that their direction of influence on the capability deviation is plausible. Where triggers depend on assumptions (for example about perception limits), these assumptions should be stated explicitly.

Safety concept coverage and consistency is a fourth check. Safety concept measures in the CBCT must be consistent with the system safety concept as a separate safety engineering work product. Reviewers therefore verify that each CBCT measure has an equivalent specification in the safety concept. Any mismatch is resolved by aligning the CBCT or updating the safety concept. If the CBCT introduces new measures to block behavior risk scenarios, these need to be documented in the safety concept and carried into implementation and validation so that they can be realized and assessed within the established safety process. As a technical basis to check safety concept coverage and consistency semi-automatically, existing work such as *Safety Concept Trees* (Zimmer, 2008) or the *Open Dependability Exchange Metamodel* (Zeller et al., 2023) can be used.

These checks are lessons learned from the execution of the warehouse case study (Section 5.1), where CBCT quality is argued via SEM reuse, guideword-supported capability deviation coverage and ontology-informed situation deviation triggers used as scoped justification.

4.3 Runtime Model Transformation

The CBCTs in Section 4.2 justify where worst-case risk is implausible and which assumption relaxations are admissible under the system safety concept. The purpose of this section is to translate that justification into a runtime-realizable mechanism: a model that takes ego-observable features as input, infers the validity state of each relevant RIH, and uses that state to select the corresponding safety monitor parameter class. The transformation is defined such that the runtime behavior can be reviewed against the original design-time artifacts (HARA → SEM → CBCT → runtime model) and such that missing or unreliable observations lead to conservative fallback rather than relaxation.

Section 4.3.1 specifies the runtime function in the safety architecture. Section 4.3.2 defines the feature mapping constraints and resulting feature sets per RIH. Section 4.3.3 outlines inference formalisms that can represent the hypothesis logic with the needed explainability and uncertainty handling. Section 4.3.4 specifies parameter class selection and switching rules based on hypothesis outputs.

4.3.1 SINADRA runtime concept

This subsection positions the SINADRA runtime function in the overall control architecture and explains how it supports situation-dependent adaptation of environment parameters in the safety monitor. As illustrated in Figure 4.15, SINADRA evaluates risk implausibility hypotheses (RIHs) online based on ego-observable features. The resulting hypothesis state is then used to select the applicable parameter class for the safety constraint, while the monitor continues to run in parallel to the nominal driving policy.

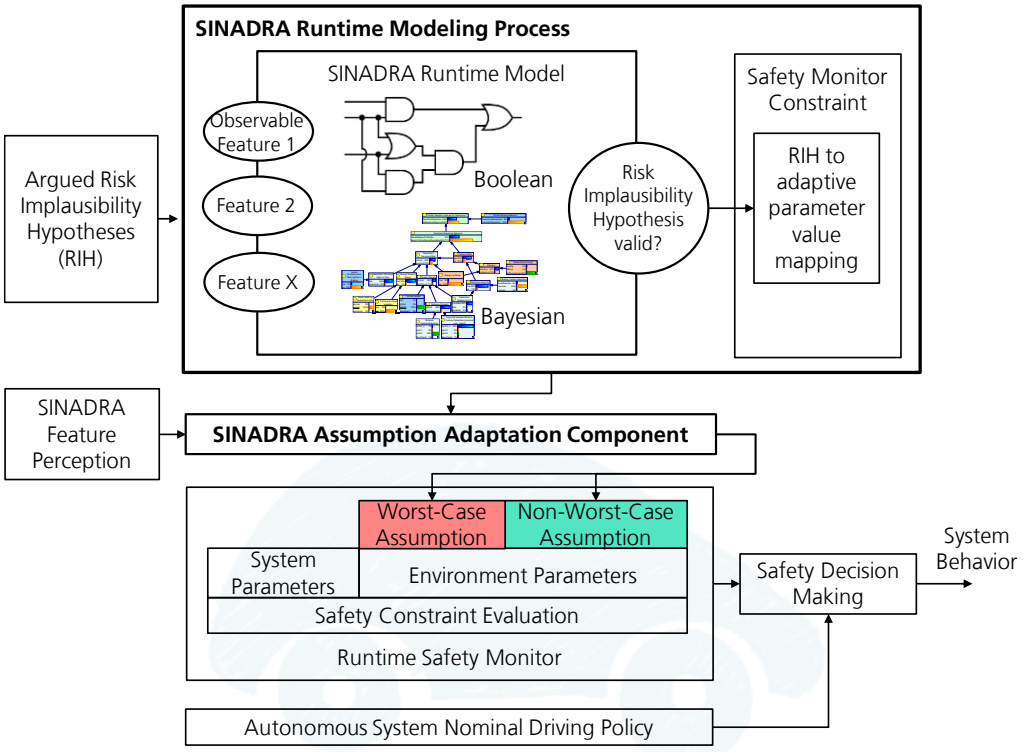


Figure 4.15 SINADRA runtime concept: Argued risk implausibility hypotheses are concretized to ego-observable features and transformed into a runtime inference model, which infers the hypothesis validity based on perceived features and switches a runtime safety monitor's environment parameter to a pre-defined permissiveness-increasing value.

The transformation from design-time CBCT hypotheses to executable monitor adaptation is defined by three steps:

(1) **Feature mapping.** For each assured RIH in the CBCT, identify ego-observable features whose values allow the hypothesis to be evaluated online. Features must be observable in the declared ODD and measurable with integrity consistent with the safety concept, either via onboard perception or trusted external inputs.

(2) **Inference model.** Implement a runtime inference model (e.g., Boolean logic or Bayesian network) that links feature states to the validity of the RIH. The model should preserve the CBCT intent, remain reviewable, and be efficient enough for embedded execution.

(3) **Parameter selection.** Map hypothesis validity to discrete parameter classes of the safety monitor (e.g., worst-case / conservative / permissive) with pre-argued margins. Separating inference from parameter choice makes residual uncertainty and safety concept constraints explicit at the parameter-class level.

Sections 4.3.2–4.3.4 detail these steps and explain how the resulting logic remains traceable to the CBCT evidence.

4.3.2 From risk implausibility hypotheses to observable features

This subsection explains how argued risk implausibility hypotheses (RIHs) from CBCTs are mapped to ego-observable features for runtime evaluation, and which admissibility constraints this imposes on feature selection.

Extraction and feature mapping. For each RIH that participates in a minimal cut set of the CBCT, engineers derive a set of ego-observable features that represent the hypothesis in its concrete behavior risk scenario. The goal is a feature set that is sufficient to decide at runtime whether the hypothesis is valid, within the scope and assumptions stated in the CBCT and safety concept.

For example, the hypothesis *"Critical front-vehicle braking need is implausible because the crosswalk is clear at intersection entry"* can be mapped to features such as:

- distance of the relevant vehicle to the crosswalk (or time-to-crosswalk),
- presence of pedestrians on the crosswalk,
- presence of pedestrians at the curb in the crosswalk entry area,
- pedestrian motion direction and speed relative to the crosswalk entry,
- coarse pedestrian body orientation relative to the roadway.

Together, these features capture whether pedestrian interaction at the crosswalk is currently relevant and whether near-term entry is plausible in a way that would induce harsh braking. Selecting a concrete feature set is an engineering decision and may draw on domain knowledge. This thesis focuses on how features are used and justified within SINADRA, not on designing optimal predictors.

Feature admissibility and detectability. A feature is admissible for SINADRA only if it is observable in the declared ODD and measurable in the ego system with integrity consistent with the safety concept. "Observable" means the

information exists in the environment and is, in principle, accessible via onboard sensing or trusted external inputs. “Measurable” means the ego system can estimate it with the required accuracy, reliability, and timing.

Three practical cases occur:

- **Not observable:** the information is not externally available (e.g., a pedestrian’s internal destination). Such variables cannot be used as hypothesis evidence with direct perception. In some settings, observability can be improved by a controlled interaction that elicits an observable response (e.g., a brief signal to test pedestrian awareness), provided such interaction is permitted in the ODD and is itself addressed in the safety concept. In SINADRA terms, this changes the hypothesis from “infer latent intent” to “check response to an observable cue,” which can be easier to specify and review, but still requires explicit assumptions and integrity justification. This aligns with the conceptual distinction in the Cynefin framework (Snowden, 2005) between complicated situations, where all relevant information is visible and a Sense–Analyze–Respond strategy is sufficient, and complex situations, where parts of the cause–effect chain are hidden and a Probe–Sense–Analyze–Respond strategy is recommended.
- **Observable but not measurable in the current setup:** the information exists but cannot be obtained with the required integrity (e.g., fully occluded pedestrians without supporting V2X/V2I).
- **Observable and measurable, but method-dependent:** the information can be estimated, but the estimation method may require safety justification (e.g., gaze direction inferred by ML).

For each RIH, feature candidates are therefore screened using two questions:

(1) Can the feature be observed in the ODD with the available sensing, communication, or permissible interaction mechanisms?
(2) Can it be measured with integrity consistent with the safety concept?

Only features that satisfy both conditions may justify parameter relaxation. If an RIH would rely on features that fail either condition, then the hypothesis cannot be used to support relaxation and the monitor must retain worst-case assumptions for the corresponding risk scenarios.

4.3.3 From features to runtime inference

This subsection explains how ego-observable features are combined into a runtime inference model that evaluates risk implausibility hypotheses (RIHs), and how the resulting hypothesis state is mapped to discrete parameter classes for the safety monitor. SINADRA requires a formalism that (1) reflects

the CBCT-derived cause–effect rationale, (2) remains reviewable for safety assurance, and (3) is efficient enough for embedded execution. For the scope of this thesis, two concrete options are used: Boolean logic and Bayesian networks.

Many modeling formalisms could be applied to infer hypothesis validity from features (e.g., temporal logics, Markov models, Dempster–Shafer, or learned classifiers). In SINADRA, the selection is guided by five criteria: expressiveness for causal dependencies, explainability for review and V&V, runtime efficiency, explicit representation of uncertainty, and portability across ODD instances. Boolean logic and Bayesian networks provide a pragmatic trade-off against these criteria and are therefore discussed as realization options below.

Runtime Inference with Boolean models

Boolean logic was used as an initial runtime formalism for three reasons. First, CBCTs already provide a Boolean structure for behavior risk scenarios and blockers, so RIH evaluation rules can reuse that structure with minimal translation effort. Second, Boolean expressions are easy to inspect because intermediate terms can be named in domain language (supporting review and test derivation). Third, evaluation is computationally inexpensive for the small variable sets targeted by SINADRA at runtime.

In the simplest case, an RIH can be expressed as a Boolean guard over discretized feature states. For the crosswalk hypothesis, a rule that mirrors the feature mapping of Section 4.3.2 is:

$$\begin{aligned} RIH_{cw_clear} = \; & \neg CrosswalkInPath \vee (CrosswalkInPath \\ & \wedge VisibilityGood \wedge \neg PedestrianOnCrosswalk \\ & \wedge \neg (PedestrianAtCurbPresent \\ & \wedge PedestrianMotionFast \wedge TTCW_short)) \end{aligned}$$

Here, CrosswalkInPath, VisibilityGood, PedestrianOnCrosswalk, PedestrianAtCurbPresent, PedestrianMotionFast, and TTCW_short are the discretized feature predicates introduced in Section 4.3.2. The rule states that the hypothesis is valid if no crosswalk is in the ego path, or, if a crosswalk is relevant, only when visibility is good, no pedestrian occupies the crosswalk, and there is no short time-to-crosswalk with a fast-approaching pedestrian at the curb.

Boolean models have two limitations in the SINADRA context. First, they represent uncertainty only indirectly (e.g., via conservative discretization or separate “unknown” states), which makes it difficult to express graded evidence from perception. Second, they cannot represent ODD-specific quantitative differences when the causal structure is the same but conditional

likelihoods differ (e.g., different pedestrian entry rates across cities given the same curb-waiting cues). These limits motivate a probabilistic inference option.

Runtime Inference with Bayesian Networks

Bayesian networks (BNs) represent variables as nodes in a directed acyclic graph, with conditional probability tables (CPTs) defining dependencies. In SINADRA, evidence nodes correspond to ego-observable features, intermediate nodes capture latent situation aspects used in the CBCT rationale, and target nodes encode hypothesis validity. Probabilistic inference then updates the hypothesis belief when evidence is observed.

BNs are useful for SINADRA for three reasons. First, they represent uncertainty explicitly, allowing noisy perception evidence and uncertain causal links to be handled in a controlled, auditable way. Second, they remain reviewable: nodes and edges have explicit semantics that can be checked against SEM/CBCT rationale, and sensitivity analyses can be used to test plausibility and robustness of the inference-to-parameter switching behavior. Third, small and sparse BNs can be executed efficiently with established inference engines suitable for embedded use (e.g., using the SMILE (BayesFusion, 2025) or aGrUM (aGrUM team, 2025) libraries).

As with any runtime model, scope limits must be explicit: a BN is only justified for the ODD slice and scenario family for which its structure and CPTs are argued. If CPTs are learned, e.g. by the approaches applied in the supervised master thesis in (Wellstein, 2021), representativeness assumptions and data governance must be stated. If engineered, the sources and reasoning for parameter choices must be documented. Out-of-scope situations and insufficiently reliable feature evidence must revert to worst-case parameter classes as described in Section 4.3.4.

Example: Bayesian network for the crosswalk hypothesis

To make the runtime transformation step tangible, we reuse the running crosswalk-related example. The hazardous situation is “Front vehicle deceleration in vehicle-following at an urban crosswalk exceeds the non-worst-case assumption.” The hypothesis of interest is “Critical front-vehicle braking due to a pedestrian at the crosswalk is implausible at crosswalk entry.” The purpose of the BN in Figure 4.16 is to operationalize this hypothesis as a function of ego-observable evidence while keeping the intermediate reasoning steps explicit.

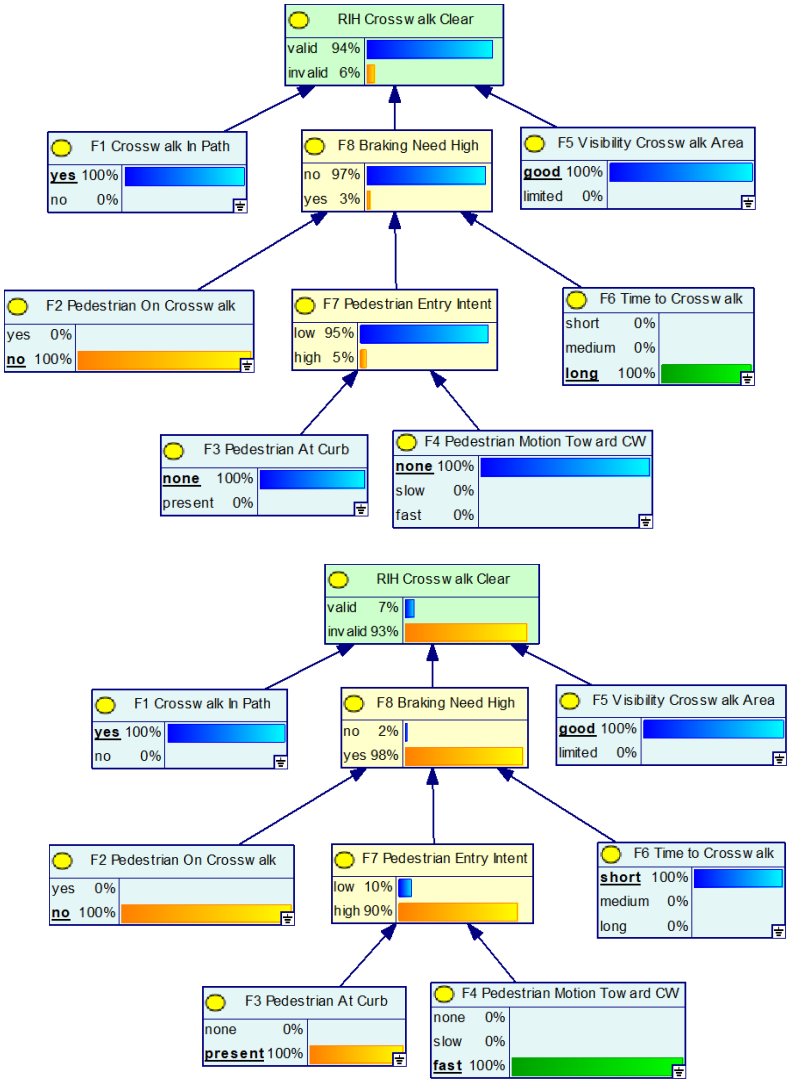


Figure 4.16 SINADRA runtime model as a Bayesian network. Top: evidence of a clear crosswalk, leading to a high probability that RIHcw-clear is valid. Bottom: evidence of a critical pedestrian situation, leading to a low validity probability and a high probability that the hypothesis is invalid. Figure created using GeNIe modeler, a product of BayesFusion, LLC.

Evidence layer (ego-observable features). The BN uses a subset of the features introduced in Section 4.3.2:

- F1: crosswalk in ego path (yes / no)
- F2: pedestrian on crosswalk (yes / no)
- F3: pedestrian at curb (none / present)
- F4: pedestrian motion toward crosswalk (none / slow / fast)

- F5: visibility of crosswalk area (good / limited)
- F6: time-to-crosswalk TTCW (short / medium / long)

These variables correspond to observable situation aspects that appear as conditions in the CBCT initiator scenario (presence and motion of pedestrians, proximity to the interaction point, and visibility constraints).

Intermediate reasoning layer (latent variables). Two latent nodes aggregate feature evidence into CBCT-aligned intermediate statements:

- F7 (Pedestrian entry intent): captures whether the curb-side cues indicate near-term road entry. Parents are F3, F4.
- F8 (Pedestrian-related braking demand high): captures whether the observed configuration is consistent with a pedestrian-driven need for strong braking near the crosswalk. Parents are F7, F2, F6.

These nodes serve two roles: they preserve a readable causal chain that reviewers can inspect, and they provide explicit insertion points for later refinement (e.g., adding orientation cues to F7).

Hypothesis layer (RIH validity). The top node is:

- RIH Crosswalk clear (RIH_{cw}) (valid / invalid) with parents F8**,** F5 and F1.

Here, F1 gates the scenario family: if no crosswalk lies in the ego path, pedestrian-driven crosswalk braking is irrelevant and the hypothesis is typically valid. When a crosswalk is in the path, the hypothesis becomes sensitive to (a) visibility as a precondition for reliable assessment and (b) braking demand as the summarized risk driver. The intended interpretation is: valid when visibility is good and pedestrian-related braking demand is low; invalid when visibility is limited or braking demand is high.

Relationship to CBCT and feature selection. The BN structure is chosen to reflect the CBCT rationale in executable form: evidence nodes represent observable elements of the initiator scenario, latent nodes summarize the intermediate conditions that drive the critical behavior path, and the hypothesis node corresponds to the CBCT RIH that blocks pedestrian-driven risk paths. SINADRA constrains feature sets to be ego-observable with sufficient integrity and timing and to remain consistent with the CBCT reasoning, but it does not prescribe an independent feature discovery method. Feature selection and justification, including the adequacy of discretization and sensing integrity assumptions for the target ODD slice, remains part of the safety engineering work.

Illustrative runtime behavior. With evidence F1–F6 observed, the BN outputs a posterior belief over RIH_{cw}. In a “clear crosswalk” configuration (no

pedestrian on crosswalk, no curb pedestrian, TTCW long, visibility good), the example parameterization yields high belief in valid (~0.94). In a configuration with near-term entry cues (curb pedestrian present, fast approach motion, TTCW short, visibility good), belief shifts toward invalid (~0.07 for validity). Section 4.3.4 then maps these outputs to parameter classes: permissive settings are selected only when validity is established; otherwise, the safety monitor retains conservative or worst-case assumptions.

4.3.4 From runtime model to parameter value adaptation

The previous subsections explained how risk implausibility hypotheses from CBCTs are mapped to ego-observable features and realized as runtime inference models. This subsection completes the chain by defining how a hypothesis state is translated into concrete safety-monitor parameter values. SINADRA does this by selecting among pre-argued parameter classes, enforcing immediate fallback to worst-case settings when admissibility cannot be established, and keeping this adaptation consistent with the system safety concept.

Adaptation policy. The adaptation policy binds hypothesis acceptance to parameter selection in a conservative way. A parameter class more permissive than the worst case is admissible only when the relevant hypothesis is accepted and required signal health checks are satisfied. Hypothesis acceptance can be defined either by a Boolean condition evaluating to true or, when using probabilistic inference, by a posterior threshold such as $P(\text{RIH} = \text{valid} \mid \text{evidence}) \geq \theta$. Loss of acceptance or degradation of required signals triggers immediate reversion to the worst-case class. In the Boolean case, missing or low-integrity evidence prevents the condition from evaluating to true, which again implies worst-case retention.

Parameter classes and project-specific calibration. SINADRA assumes a small discrete set of classes per adapted monitor parameter (e.g., *worst-case*, *conservative*, *permissive*). The runtime model decides which class is admissible in the current situation. However, it does not fix the numeric values of the classes or the acceptance threshold θ. These design choices are project-specific and must be justified against the acceptable residual risk stated in the safety concept. Suitable means include simulation-based studies, test campaigns, and analysis of operational data.

Acceptable residual risk thresholds for automated driving are not fully settled at the societal level. SINADRA therefore treats these thresholds as given inputs rather than attempting to define them. For a broader discussion of “How safe is safe enough?”, see (Koopman, 2022).

Separation of admissibility and margin choice. The separation between hypothesis validity and parameter value selection is intentional. SEM and

CBCT provide a traceable argument for *when* a relaxation is admissible within the analyzed scope. How far to move within that admissible range remains a safety engineering decision that can depend on ODD, vehicle platform, and evidence base without changing the SINADRA method. The key requirement is that each parameter class is argued in the safety case with respect to residual risk and that the adaptation logic reverts to worst-case settings whenever hypothesis acceptance or signal integrity is not established.

Fit into safety assurance. SINADRA's scientific contribution ends with (1) the evidence chain from HARA to SEM, CBCT, and runtime hypotheses and (2) the adaptation policy that switches among pre-argued parameter classes based on hypothesis acceptance and signal health. Deploying these components in a safety-critical system requires applying established functional safety and SOTIF life cycles to the concrete implementation and its assumptions.

In practice, a project adopting SINADRA must:

- derive safety and SOTIF requirements for the added features, inference models, and adaptation logic
- implement and verify perception and inference such that integrity and timing satisfy these requirements
- justify parameter class values and acceptance thresholds against residual risk targets in the safety concept
- preserve traceability from HARA through SEM and CBCT down to the software and configuration items that realize the runtime function

The project-specific assurance activities and detailed evidence are beyond the scope of this thesis. They form part of industrial uptake of SINADRA within existing functional safety and SOTIF processes, using the SINADRA artifacts and trace links as inputs to the safety case.

4.4 Prototypical tool support

This thesis is accompanied by tool support that shows that the main SINADRA steps can be realized in software. The tool support reported here originates from implementation work carried out in the European Horizon 2020 research project SESAME[1] ("Secure and Safe Multi-Robot Systems"). The purpose of this subsection is to document the existence and capabilities of that prototype and to position it in the final version of the method presented in this thesis.

The SESAME prototype demonstrates that the SINADRA engineering method can be technically executed. The prototype enables engineers to model CBCT

[1] SESAME Project Website (grant agreement No. 101017258): https://www.sesame-project.org/

and Bayesian network models in modeling frontends, create structured representations in the tool-independent *open dependability exchange* (ODE) format and generate Boolean and Bayesian runtime software components from this format, which can be integrated in target runtime stacks like the Robot Operating System (ROS).

The prototype serves as feasibility evidence that the method steps introduced in this chapter can be partially operationalized and that the "design-time artifact → runtime monitor component" link can be implemented.

At the same time, this thesis' final CBCT concept and element names evolved after the SESAME reports were written in 2022. Therefore, the SESAME prototype is not claimed to implement the final CBCT method one-to-one. Instead, it is used as a documented prior prototype that implements the same *class of functions*: structured model handling, automated generation of runtime monitor components, and runtime deployment support.

Figure 4.17 shows the SINADRA toolchain prototype architecture. It supports three phases of the SINADRA engineering process and enables engineers:

(1) to model CBCT models in the *safeTbox*[2] modeling frontend, formally trace CBCT safety concept measure elements to Goal Structuring Notation (GSN) safety concept models and CBCT output deviation mode elements to hazards in HARA models

(2) to create a structured tool-independent design-time representation of SINADRA models, Boolean models and Bayesian network models conforming to the respective metamodels in the Open Dependability Exchange (ODE) format, which is the format for so-called *Executable Digital Dependability Identities* (EDDI). EDDIs are structured knowledge graphs for dependability information and are a central technology in the SESAME project.

(3) to transform design-time representations of Boolean or Bayesian runtime models in the ODE format into Python or *Robot Operating System* (ROS) software components, which can be integrated with applications in target software stacks of robotic systems

[2] safeTbox safety engineering toolsuite developed at Fraunhofer IESE: https://safetbox.de/

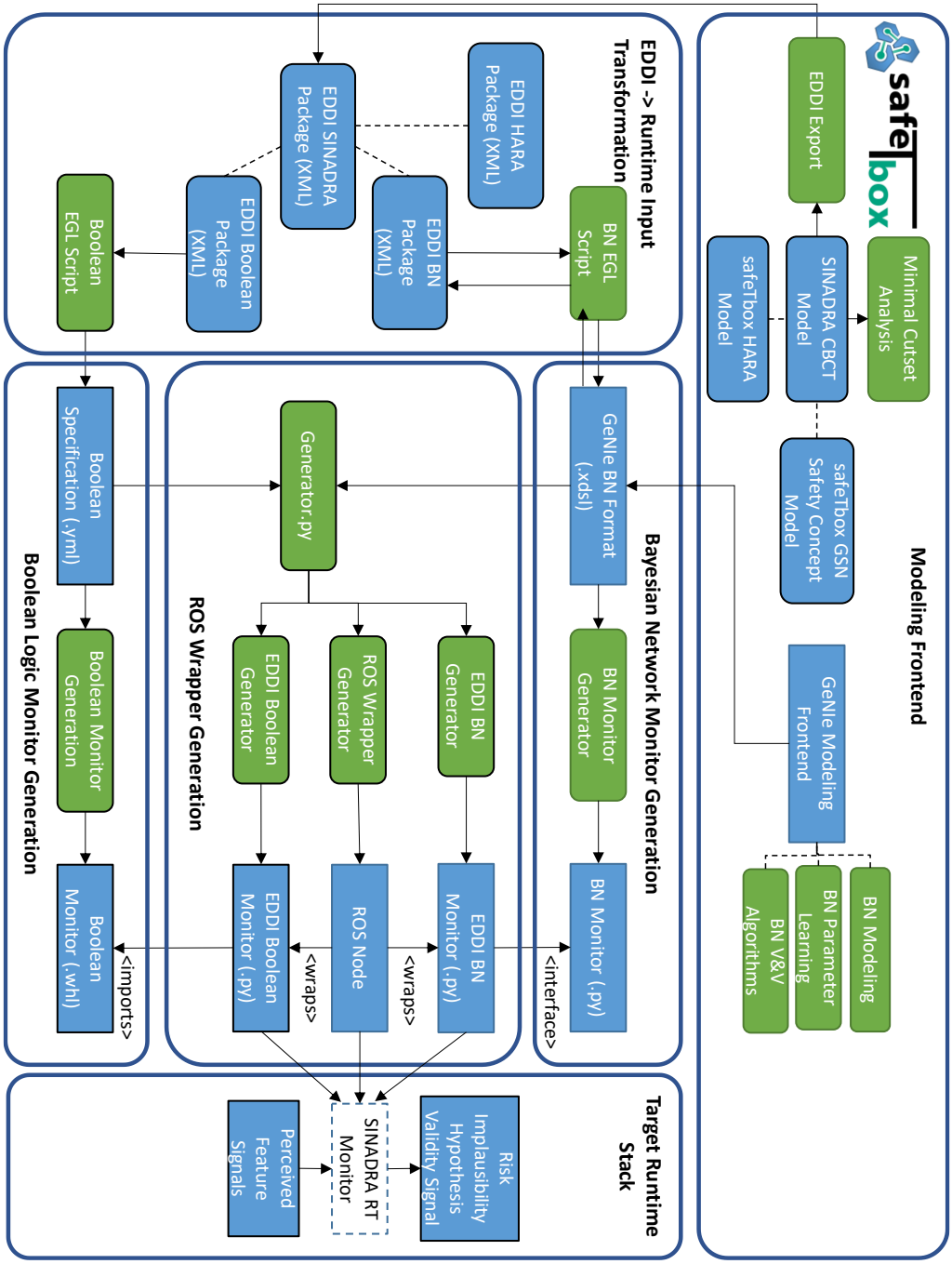


Figure 4.17 SINADRA Toolchain Prototype Architecture. Blue nodes are artifacts (models, files, software code), green nodes are scripts, algorithms or tool features. Abbreviations: EDDI = Executable Digital Dependability Identity, EGL = Epsilon Generation Language, ROS = Robot Operating System, GSN = Goal Structuring Notation, V&V = Verification & Validation.

4.4.1 CBCT modeling in safeTbox

For modeling CBCT models in safeTbox, existing features surrounding component fault tree analysis have been adapted to support the semantics of CBCTs. Structurally, CBCT models are similar to component fault trees given that both formalisms are hierarchical Boolean tree structures. Therefore, the basic modeling elements like sub models, logical gates and basic events can be reused. The safeTbox feature to assign formal types to basic events was used to model and differentiate the specific CBCT element types *Presence Indicator*, *Agent Deficiency*, *Situation Deviation Trigger*, *Risk Implausibility Hypothesis* and *Safety Concept Measure*. Figure 4.18 shows an example CBCT model in safeTbox and the component fault tree toolbox.

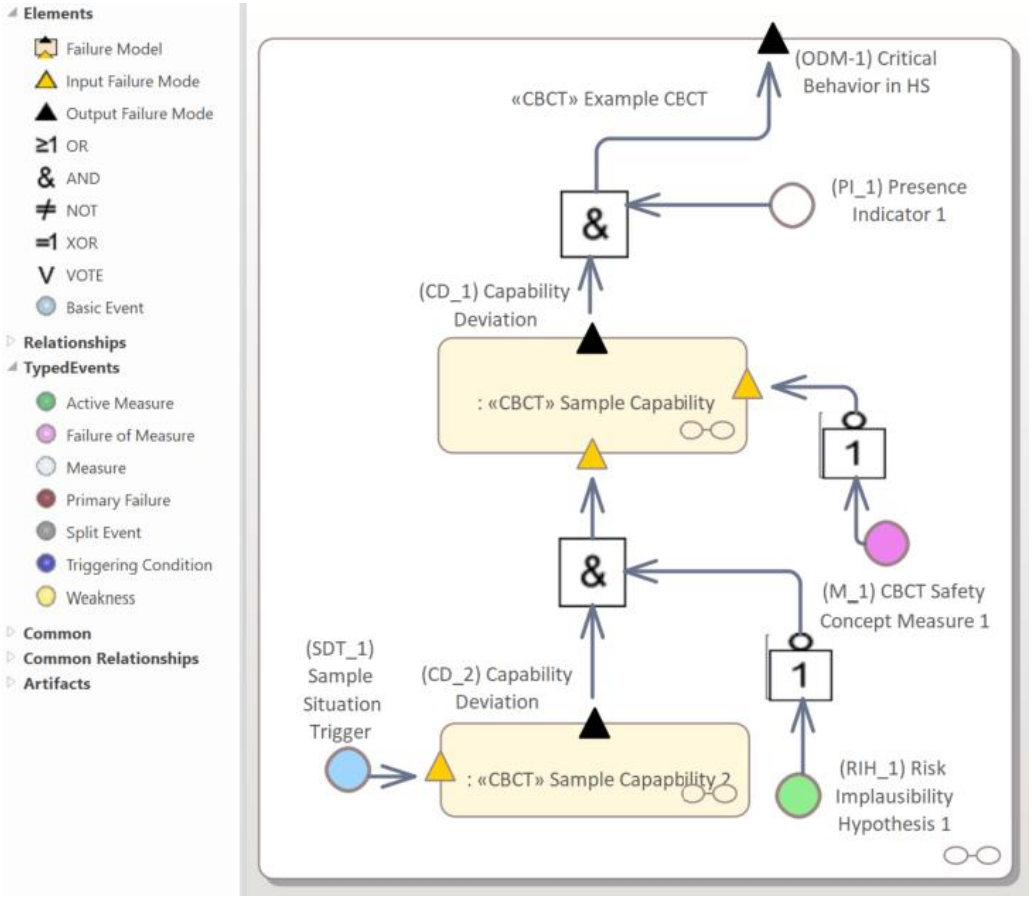


Figure 4.18 CBCT modeling toolbox and example model in safeTbox. The basic elements of the component fault tree modeling features in safeTbox are adapted to CBCT semantics by using typed basic events.

The formal traceability between CBCT elements on the one hand and hazards in the HARA as well as safety measures defined in GSN safety concepts on the other hand is natively supported by the so-called dynamic linking feature of safeTbox. Establishing this traceability helps for documentation purposes and can be used to verify consistency between the models. Figure 4.19 shows an example safety concept modeled with the Goal Structuring Notation (GSN) profile in safeTbox. Within the description of the typed safety measure element, the formal dynamic link to the example CBCT safety concept measure element *CBCT Safety Concept Measure 1* (*M_1* in Figure 4.18) is established. Similarly, Figure B.1 in Appendix B shows the formal traceability between CBCT output deviation modes and hazard elements in the safeTbox HARA model.

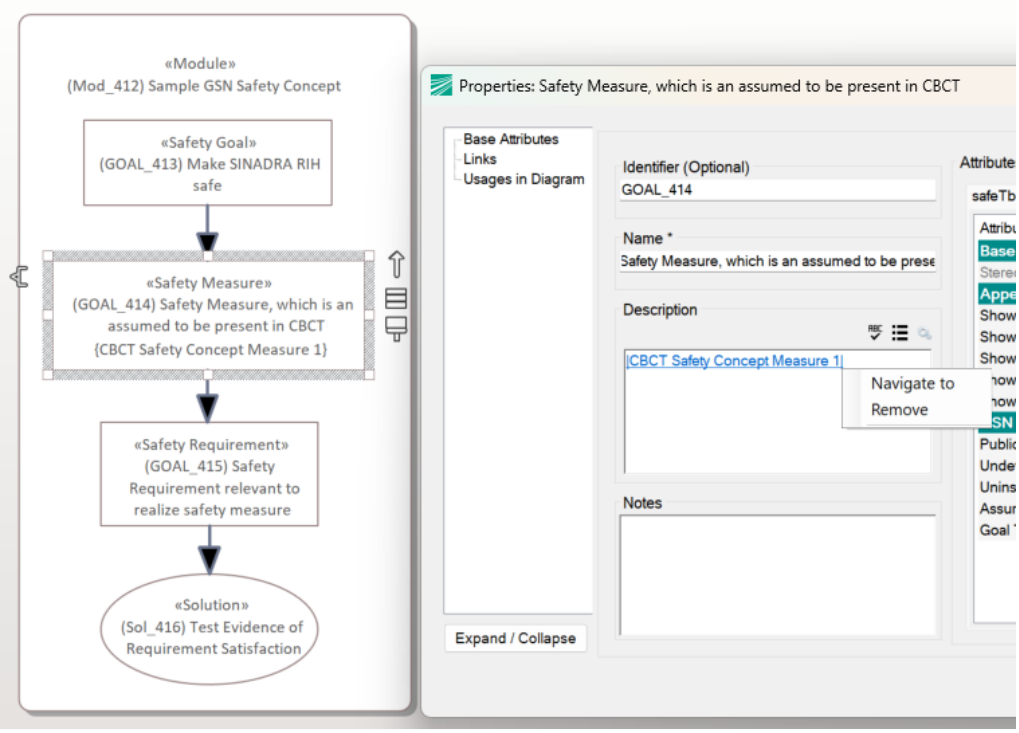


Figure 4.19 Traceability between safety concept measures and GSN safety concepts in safeTbox.

The implemented qualitative analysis algorithms in safeTbox are used to automatically retrieve the minimal cut sets for CBCTs. In Figure B.2 in Appendix B, the safeTbox minimal cut set analysis result dialog is shown with the possibility to filter and process CBCT minimal cut sets by using the formal CBCT element types. In the future, CBCT basic elements might even get assigned quantitative parameters. Then, the quantitative analysis algorithms implemented in safeTbox can be used to retrieve event importance or top event likelihood. This might be useful for analyses to determine the

quantitative effects of situation deviation triggers on critical behavior likelihood, if frequency parameters are assigned to the triggers.

After the CBCT model is finalized in safeTbox, the next step is to export it into the tool-independent representation specified by the *Open Dependability Exchange* (ODE) format (Zeller et al., 2023). The ODE format was originally developed as part of the *Digital Dependability Identity* (DDI) concept in the Horizon 2020 DEIS project[3] (2017-2019) and evolved towards runtime dependability monitoring (*Executable DDIs* or *EDDI*) in the SESAME project (2021-2023). EDDIs are XML-based instance representations of integrated dependability models compliant with the metamodels defined by the ODE format. The SESAME deliverable D4.2 (SESAME Project Consortium, 2022a) describes the latest version of the ODE metamodel specification. Tooling to create, manipulate and query DDI models with the Eclipse Epsilon Scripting Engine is described in deliverable D4.6 (SESAME Project Consortium, 2023). It also describes the modeling and analysis features of safeTbox as well as the Bayesian network tool GeNIe[4] from BayesFusion. Deliverable D5.4 (SESAME Project Consortium, 2022b) describes how the DDI *tool adapter* connects tools like safeTbox or GeNIe to the DDI world for import/export/manipulation purposes.

The ODE::SINADRA metamodel is described in the appendix in Figure B.3. Although this is the target format for the CBCT export, the naming and structure of elements is different than the final SEM and CBCT elements in this thesis. The reason is that the SINADRA metamodel evolved since the end of the SESAME project. Nevertheless, the goal of this section is to show that the transformation from CBCT models to the DDI world is *in principle* possible. While SEM and CBCT concepts were still mixed in the ODE::SINADRA model, they are now separated in this thesis.

To still show the principle, Figure 4.20 shows the example output of the automated EDDI export of the example CBCT from Figure 4.18 in safeTbox into an ODE::FailureLogic model. This metamodel captures hierarchical Boolean structures like component fault trees or CBCTs. With this step, the tool-independent model world is entered and the contents can be processed via the full capabilities of the Epsilon scripting engine. Since the CBCT element semantics are preserved in the export, an Epsilon script can transform the CBCT elements into the target ODE::SINADRA metamodel after it is technically updated to the model structure presented in this thesis.

[3] https://cordis.europa.eu/project/id/732242

[4] https://www.bayesfusion.com/genie/

```
<?xml version="1.0" encoding="ASCII"?>
<integration:DDIPackage xmi:version="2.0" xmlns:xmi="http://www.omg.org/XMI" xmlns:xsi="http://www.w3.org/2001/XMLSchema-in
  <odeProductPackages xsi:type="failureLogic_:FailureLogicPackage" Id="1" name="FailureLogicPackage" description="">
    <failureModels xsi:type="failureLogic_:FaultTree" Id="3" name="&#xab;CBCT&#xbb; Sample Capapbility 2" description="">
      <failures Id="4" name="(CD_2) Capability Deviation" description="" originType="Output" failureClass="NONE"/>
        <failureProbDistribution Id="7" name="" description="" type="NONE"/>
      </failures>
      <failures Id="9" name="IDM" description="" failureClass="NONE">
        <failureProbDistribution Id="10" name="" description="" type="NONE"/>
      </failures>
      <failures Id="14" name="Agent Deficiency 2" description="" originType="Agent Deficiency" failureClass="">
        <failureProbDistribution Id="15" name="" description="" type="NONE"/>
      </failures>
      <causes xsi:type="failureLogic_:Gate" Id="5" name="(CD_2) Capability Deviation" description="" causeType="OutputEvent
      <causes xsi:type="failureLogic_:Gate" Id="11" name="IDM" description="" failure="//@odeProductPackages.0/@failureMode
      <causes xsi:type="failureLogic_:Gate" Id="12" name="Or_62" description="" causeType="Gate" causes="//@odeProductPacka
      <causes xsi:type="failureLogic_:Gate" Id="13" name="And_96" description="" causeType="Gate" gateType="AND" causes="//
      <causes Id="16" name="Agent Deficiency 2" description="" causeType="BasicEvent" failure="//@odeProductPackages.0/@fai
    </failureModels>
    <failureModels xsi:type="failureLogic_:FaultTree" Id="17" name="&#xab;CBCT&#xbb; Sample Capability" description="">
      <failures Id="18" name="(CD_1) Capability Deviation" description="" originType="Output" failureClass="NONE"/>
      <failures Id="20" name="(IDM_1) IDM" description="" failureClass="NONE">
        <failureProbDistribution Id="21" name="" description="" type="NONE"/>
      </failures>
      <failures Id="23" name="Input Deviation Mode" description="" failureClass="NONE">
        <failureProbDistribution Id="24" name="" description="" type="NONE"/>
      </failures>
      <failures Id="28" name="Agent Deficiency" description="" originType="AD" failureClass="">
        <failureProbDistribution Id="29" name="" description="" type="NONE"/>
      </failures>
      <causes xsi:type="failureLogic_:Gate" Id="19" name="(CD_1) Capability Deviation" description="" causeType="OutputEven
      <causes xsi:type="failureLogic_:Gate" Id="22" name="(IDM_1) IDM" description="" failure="//@odeProductPackages.0/@fai
      <causes xsi:type="failureLogic_:Gate" Id="25" name="Input Deviation Mode" description="" failure="//@odeProductPackag
      <causes xsi:type="failureLogic_:Gate" Id="26" name="Or_60" description="" causeType="Gate" causes="//@odeProductPacka
      <causes xsi:type="failureLogic_:Gate" Id="27" name="And_73" description="" causeType="Gate" gateType="AND" causes="//
      <causes Id="30" name="Agent Deficiency" description="" causeType="BasicEvent" failure="//@odeProductPackages.0/@failu
    </failureModels>
    <failureModels xsi:type="failureLogic_:FaultTree" Id="2" name="&#xab;CBCT&#xbb; Example CBCT" description="" subModels=
      <failures Id="31" name="(ODM-1) Critical Behavior in HS" description="" originType="Output" failureClass="NONE"/>
      <failures Id="38" name="Risk Implausibility Hypothesis 1" description="" originType="RIH" failureClass="">
        <failureProbDistribution Id="39" name="" description="" type="NONE"/>
      </failures>
      <failures Id="41" name="Sample Situation Trigger" description="" originType="SDT" failureClass="">
        <failureProbDistribution Id="42" name="" description="" type="NONE"/>
      </failures>
      <failures Id="44" name="CBCT Safety Concept Measure 1" description="" originType="M" failureClass="">
        <failureProbDistribution Id="45" name="" description="" type="NONE"/>
      </failures>
      <failures Id="47" name="Presence Indicator 1" description="" originType="PI" failureClass="">
        <failureProbDistribution Id="48" name="" description="" type="NONE"/>
      </failures>
      <causes xsi:type="failureLogic_:Gate" Id="32" name="(ODM-1) Critical Behavior in HS" description="(Unsafe Distance)"
```

Figure 4.20 Example output excerpt of the automated EDDI Export of the example CBCT from safeTbox into an ODE::FailureLogic Package.

4.4.2 Modeling Bayesian networks in GeNIe and runtime component generation

Based on the assumed CBCT model now being part of the EDDI model in the ODE::SINADRA format, the next step is to create a SINADRA runtime model out of it. This realizes the runtime model transformation step introduced in Section 4.3. Although the tool prototype supports the generation of both runtime Boolean models and Bayesian networks (BN), the following shows exemplarily the BN toolchain.

Based on the CBCT information in the EDDI, risk implausibility hypotheses can be modeled along with their situation features as Bayesian networks in the GeNIe modeler tool (see Figure 4.16 for an example). The tool contains features for modeling BNs, but also for automated learning of the BN's quantitative parameters based on data and various validation and verification analyses. After the model is created in GeNIe, the SINADRA tooling provides scripts for two purposes:

First, the export of the GeNIe BN (the tool uses the .xdsl format to model BNs) back into the EDDI to establish traceability to the CBCT contents in the ODE::SINADRA package. The target ODE::BayesianNetwork format for this export and the interface to ODE::SINADRA are shown in Figure 4.21.

Second, the generation of runtime software code according to the code generation pipeline shown in the prototype architecture in Figure 4.17. This can be either Python code if compatible with the target software stack or wrapped into a ROS node to be integrable with ROS-based applications.

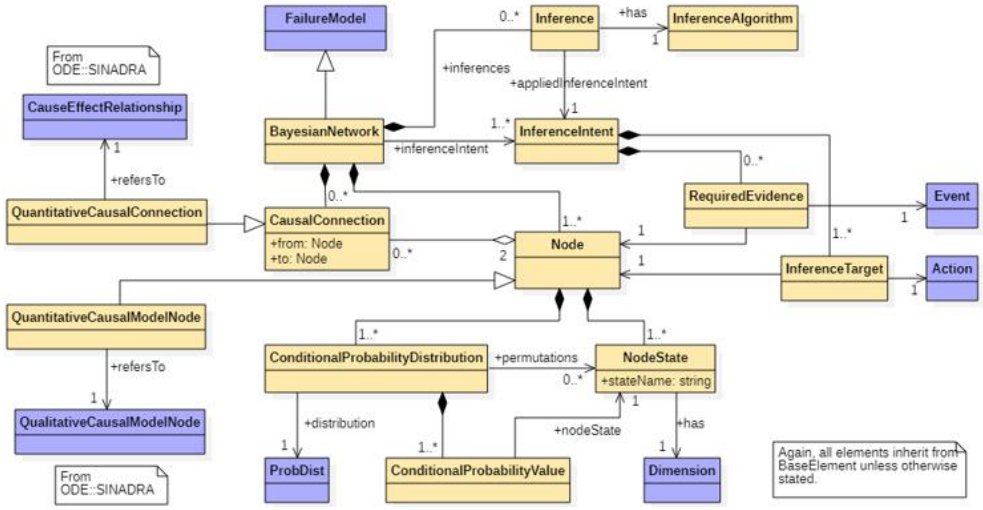


Figure 4.21 ODE::BayesianNetwork metamodel in the Open Dependability Exchange (ODE) format

The intermediate transformation results of the BN Runtime Code Generation pipeline for both steps are shown in Figure 4.22. For step 1, the runtime BN model created in GeNIe as an .xdsl file is transformed via an Epsilon Generation Language (EGL) script into the EDDI BN model to establish traceability to EDDI CBCT contents. For step 2, the BN interface Python code stubs are automatically generated to specify the mapping from input signals in a concrete application to node states in the software representation of the BN. A more extensive code example is shown in Figure C.5 in Appendix C.2. After this specification, the BN software component is inference-ready in Python applications or a ROS wrapper can be generated for it with another script. For the algorithmic BN inference, implementations were integrated using either the pgmpy[5] library or the commercial SMILE[6] library. SESAME D7.2 (SESAME Project Consortium, 2022d) contains more details about this process.

[5] https://pgmpy.org/

[6] https://www.bayesfusion.com/smile/

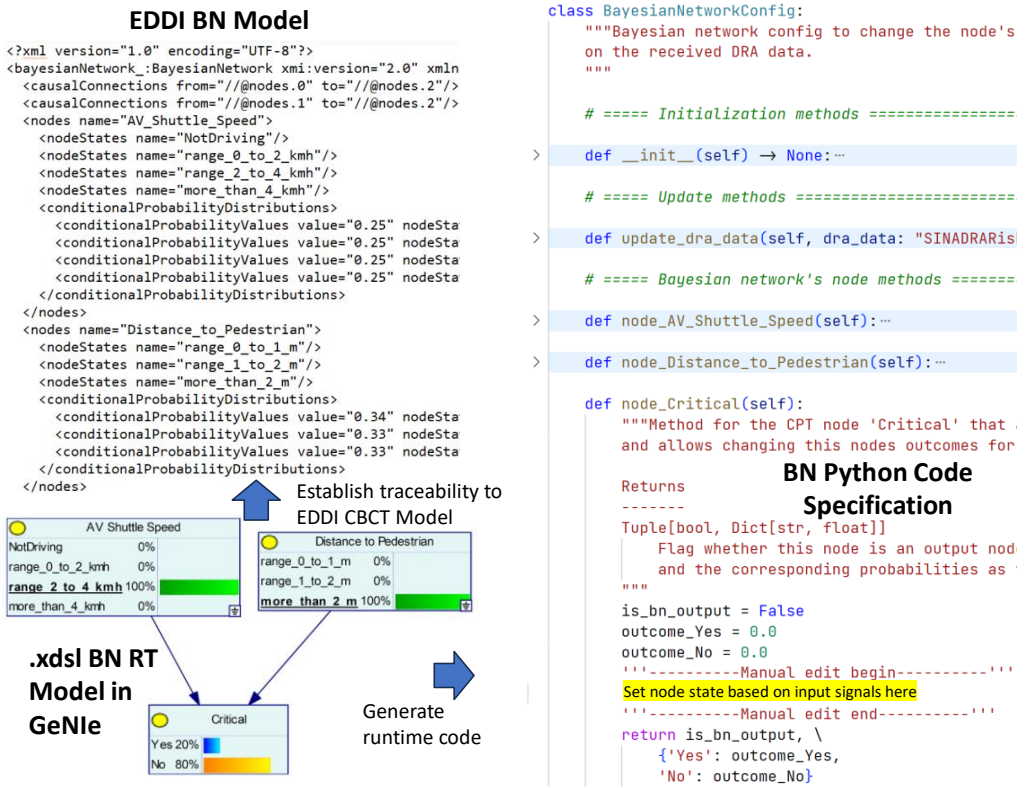


Figure 4.22 Intermediate Transformation Results of BN Runtime Code Generation Pipeline.

4.4.3 Usage of prototype tool in this thesis

Warehouse Case Study (Section 5.1). The CBCT modeling capabilities in safeTbox are applied to model CBCTs for critical behaviors of human workers and produce verification evidence with the minimal cut set analysis algorithms. To generate the Boolean runtime models for the simulative feasibility demonstration and integrate them in the ROS-based symbiotic safety system, the Boolean generation pipeline described in SESAME deliverable D7.2 (SESAME Project Consortium, 2022d) is used.

ADS Case Study (Section 5.2). In this case study, the Bayesian networks capturing the SINADRA runtime models are modeled in the GeNIe tool. Afterwards, the Bayesian generation pipeline described in Section 4.4.2 is used to generate Python code, into which the CARLA input and output signals are manually integrated into the generated code stubs as seen in Figure 4.22, right. For inference in this case study, the pgmpy library was used. The code

generation scripts from .xdsl are publicly available in the public GitHub repository[7] that contains the material about the ADS case study.

4.5 Chapter Summary

Chapter 3 introduced SINADRA as a solution concept with three linked elements. Chapter 4 instantiated these elements as an engineering method that connects HARA artifacts to environment and critical behavior modeling, integrated risk reasoning, and executable runtime adaptation.

First, **Structured Environment Modeling (SEM)** refines coarse HARA situation classes into initiator sources, initiator scenarios, and agent capabilities. This refinement is needed because HARA situation classes are usually too coarse to express which external conditions activate critical other-agent behavior and which capability assumptions are relied upon when relaxing monitor parameters. The chapter defined SEM modeling primitives, an ontology-guided elicitation procedure, and qualitative checks for ontology representativeness and scenario–capability coverage. SEM thereby provides a traceable and reviewable environment model that bounds the later CBCT analysis.

Second, **Critical Behavior Cause Trees (CBCT)** build on SEM to model behavior risk scenarios and risk measures in a single Boolean structure. This integration is needed because permissiveness-increasing relaxation must be justified against the causal paths to critical behavior, including both SINADRA risk implausibility hypotheses and other safety concept measures. The chapter introduced typed CBCT elements and their formal semantics and showed how minimal cut set analysis supports sufficiency checks: relaxation is admissible within the modeled scope only if each cut set is intercepted by at least one blocker (a hypothesis or a safety measure).

Third, the chapter specified **runtime model transformation** from assured hypotheses to executable inference and parameter selection. Ego-observable features are mapped to hypothesis validity via Boolean logic or Bayesian networks, and an adaptation policy selects among pre-argued parameter classes based on hypothesis acceptance and signal health. Conservative fallback is mandatory: if hypothesis acceptance or required signal integrity cannot be established, the monitor remains (or returns) at worst-case settings. Parameter class values and acceptance thresholds are project-specific and must be justified against residual-risk targets stated in the safety concept. SINADRA provides the traceable structure that links these choices back to HARA, SEM, and CBCT.

[7] https://github.com/JaRei/sinadra

Finally, Section 4.4 presented a prototype toolchain that demonstrates partial operationalization of the method steps, including structured model handling and generation of Boolean and Bayesian runtime components along the DDI workflow.

Overall, Chapter 4 establishes the method backbone required for evaluating SINADRA: it defines how to derive monitorable hypotheses from HARA with explicit behavior and context modeling, how to check their sufficiency in context, and how to realize controlled runtime assumption adaptation with conservative fallback. Chapter 5 evaluates the approach in two case studies.

5 Application and Evaluation

This chapter provides an overview of the case studies, in which the different elements of the SINADRA framework introduced in Chapter 4 were applied, and evaluates the degree to which the solution answers the research question posed in Section 2.6.

The case studies were run during the SINADRA design science research process in different industry and research projects and therefore each case study focuses on different aspects of the framework. The experiences and results of each case study led to an evolution of both concept and method towards the final framework described in this thesis.

To make the evaluation of the research question

"*How can HARA artifacts be extended to systematically identify monitorable low-risk situations that justify adaptive, permissiveness-increasing constraint relaxation in runtime safety monitors for autonomous systems?"*

more concrete, the following evaluation perspectives are used:

EP1 – Methodological feasibility. Do the core concepts and overall SINADRA engineering methodology enable engineers to derive risk implausibility hypotheses from hazardous situations in a systematic and verifiable manner across applications and operational design domains? This evaluation perspective particularly examines the capabilities of the solution elements related to Structured Environment Modeling and Critical Behavior Cause Tree Modeling, introduced in Sections 4.1 and 4.2, respectively. The solution is considered valid regarding EP1, if:

- **EP1a:** risk implausibility hypotheses are derived in two different operational design domains with different types of external agents influencing the risk of the autonomous system and
- **EP1b:** risk implausibility hypotheses are derived from hazardous situations by using the introduced SEM and CBCT modeling methods
- **EP1c:** the created models enable the provision of verification evidence that adapting safety constraints during runtime based on the hypotheses is justifiable in the context of a system safety engineering process

EP2 – Runtime feasibility. Can the derived risk implausibility hypotheses be transformed into executable runtime models, which are integrable with industrial safety monitoring architectures and increase the permissiveness at

runtime compared to static worst-case assumptions? This evaluation perspective particularly examines the capabilities of the runtime model transformation solution element introduced in Section 4.3. The solution is considered valid regarding EP2 if:

- **EP2a:** risk implausibility hypotheses are systematically transformed into software components, which enable runtime inference of hypothesis validity in real time based on state-of-the-art perception
- **EP2b:** the resulting SINADRA runtime component can be integrated into state-of-the-art safety monitoring architectures
- **EP2c:** the integrated runtime chain produces a runtime-varying output (risk implausibility hypothesis validity signal or collision risk metric output) based on inferred hypothesis validity. This output is shown to be suitable to support increased permissiveness compared to worst-case assumptions. Case Study 1 demonstrates explicit monitor-driven behavior change, while Case Study 2 demonstrates runtime risk-output differentiation relative to a worst-case intent baseline.

The SINADRA approach was applied in numerous studies with varying scales and scopes in different research and industry projects. From this set of applications, the two case studies described in Sections 5.1 and 5.2 serve as the primary evaluation, while the scope of the other studies will be briefly summarized below.

Case Study 1 – Co-located human-robot operation in warehouses (Section 5.1): The first case study was conducted in an industrial research cooperation with Hitachi Ltd. (Reich et al., 2022). The goal was to leverage the SINADRA approach to reduce waiting and stopping times of autonomous mobile robots operating in smart logistics warehouses in shared spaces with human workers. The controlled nature of the warehouse operational design domain compared to public road traffic and the traits of human workers in industrial environments compared to human agents in street traffic enable an end-to-end SINADRA application to study the concept and method applicability. Hitachi's symbiotic safety system (Ishigooka et al., 2022) as a baseline safety monitoring approach used worst-case assumptions prior to the SINADRA integration. In the case study, the full SINADRA process chain was executed, but the evaluation focus rests on the methodological feasibility perspective, specifically regarding the SEM and CBCT analysis and modeling aspects. The resulting risk implausibility hypotheses are operationalized as Boolean runtime models and technically integrated and demonstrated with the ROS-based symbiotic safety monitoring system.

Case Study 2 – Urban ADS Driving (Section 5.2): The second case study was conducted in an industrial research cooperation with Intel Labs (Reich et al., 2021). The goal was to study how situation-specific knowledge about risk at

runtime can support the safety–permissiveness tradeoff in urban driving scenarios involving nearby human-driven vehicles. The case study transfers SINADRA concepts to the public road domain and focuses on runtime feasibility at component level. In contrast to the simplified Boolean operationalization used in Case Study 1, this study operationalizes behavior-related risk assumptions as Bayesian networks (BN) that infer discrete behavior intents (e.g., braking intent or cut-in intent) from situation cues at runtime. The BN output is integrated with the integral collision risk computation of (Eggert, 2018) to produce an intent-weighted risk output. The feasibility of this runtime chain is demonstrated in the CARLA simulator for front vehicle braking and side vehicle cut-in situations, showing how the risk output differs from a worst-case intent baseline assumption. The supervised master's thesis in (Wellstein, 2021) is referenced as complementary work that explored alternative BN structure and parameter learning options for a related lane-change use case.

Table 5.1 lists the contents and evaluation scopes of the two primary case studies. The next sections present the results of the case studies in more detail.

In the following, other applications and studies related to SINADRA are listed. In the European SESAME project, SINADRA was applied in drone-based search-and-rescue missions to dynamically determine the situation-aware risk of missed persons (Nikolaou et al., 2025). In the same project, qualitative CBCT-style models were created to model dynamic human UV-C overexposure risks related to human actors in an application, where robots automatically disinfected spaces in hospitals (SESAME Project Consortium, 2022c). Instead of dynamically adapting the parameter values of rule-based safety monitors as done in the warehouse case study, SINADRA was applied to dynamically adapt the constraints of a stochastic model predictive control (MPC) architecture in automated driving use cases reported by (Ulmen et al., 2023). Within an automated campus shuttle use case, the capability of SINADRA to avoid the freezing robot problem in crowded passageways with many pedestrians was studied by (Jan et al., 2022).

Table 5.1 Scoping of evaluation focus for the primary case studies and produced evidence for evaluation criteria.

Evaluation Perspective		**Industrial Case Study 1: Warehouse Robotics** Scope: End-to-end application and methodological feasibility	**Industrial Case Study 2: Urban ADS Driving** Scope: Runtime feasibility at component level (BN intent inference and risk integration)
Key Publications		(Reich et al., 2022)	(Reich et al., 2021)
		EP1 Methodological Feasibility	
EP1a	Cross-domain applicability	- **Human worker** behavior as risk variability source in autonomous robot-human co-located operation in **smart logistics warehouses**	- **Human vehicle driver** behavior as risk variability source in **urban ADS driving** scenarios
EP1b	Design-time artifact chain feasibility	- SEM and CBCT for human worker behavior for hazardous situations related to unsafe distance between robot and human in passing and overtaking scenarios. - Exemplary integration of **safety concept measures**	- Situation class scoping and critical behavior cause–effect intent models (early CBCT variant) for front vehicle braking and side vehicle cut-in
EP1c	Design-time justification support	- Verification of derived risk implausibility hypotheses through CBCT minimal cut set analysis. - Verification of CBCT model quality by design via using SEM for CBCT model construction	- No data-based verification of BN structure/parameters
		EP2 Runtime Feasibility	
EP2a	Runtime model feasibility	- Runtime operationalization as **Boolean models** - Semi-automated generation of ROS component from model specification	- Runtime operationalization as **Bayesian networks** for critical behavior intent inference, implemented in Python
EP2b	Runtime integration feasibility	- Technical integration into **ROS-based** Hitachi symbiotic safety monitoring architecture	- Prototypical integration with the integral collision risk computation pipeline (Eggert, 2018) in CARLA simulation stack
EP2c	Observable runtime effects	- Runtime Feasibility demonstrated in simple simulation environment for **passing / overtaking scenarios** - **Analytic permissiveness improvement** compared to worst-case average passing time of robot: ~2% (passing scenario), ~21% (overtaking scenario)	- Runtime intent inference and intent-weighted risk output demonstrated in **CARLA simulator** for **front vehicle braking** and **side vehicle cut-in** - Shows identification of lower-risk situation instances compared to a worst-case intent baseline **at the risk-output level**. This supports (but does not by itself prove) constraint relaxation

5.1 Case Study 1: Co-located human-robot operation in warehouses

The first case study was conducted with Hitachi Ltd. in the context of smart logistics warehouses where autonomous mobile robots and human workers operate in the same areas. In particular, the study focuses on shared narrow corridors in the sorting area, where close interactions between robots and workers occur frequently. The accidents in scope are collisions between a robot and a human worker. Within this scope, the hazardous situation addressed is an unsafe distance between a robot and a worker during corridor interactions. To concentrate the analysis on recurring interaction patterns, the study considers two abstract operational situations in shared narrow corridors. The first is passing, where worker and robot move in opposite directions within the same corridor. The second is overtaking, where a robot approaches a slower worker from behind in the same corridor direction. The central challenge is that corridor safety monitoring often relies on conservative assumptions about worker motion. These assumptions are reflected in the size of the worker risk area, which represents the space in which the worker is assumed likely to move within a given time horizon. If worst-case assumptions are applied uniformly, the risk area becomes overly large in many situations and reduces permissiveness by forcing the robot to stop or wait unnecessarily. The aim is thus to derive and justify situation-dependent assumptions about worker motion for these corridor interactions and to show that selected assumptions can be operationalized in a ROS-based demonstrator, where runtime validity signals drive dynamic safety constraints on the worker risk area.

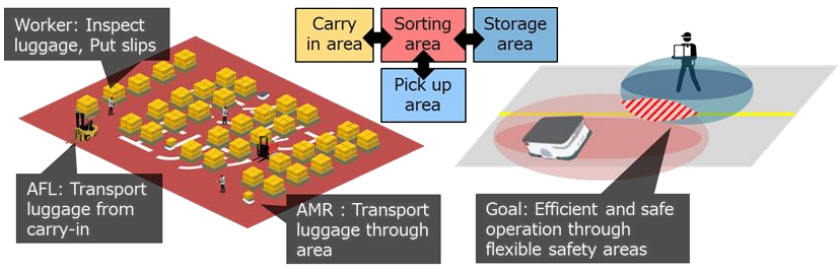


Figure 5.1 Context of the warehouse use case. Reproduced from Fig. 1 in (Reich et al., 2022) with permission from Springer Nature. © 2022, under exclusive licence to Springer Nature Switzerland AG.

Special characteristics. The warehouse environment provides boundary conditions that differ from public road traffic. The layout and movement spaces are planned and constrained by corridor networks, operational zones, and defined task flows. In addition, the setting can be instrumented in ways that support structured modeling and runtime observation. For example, worker-worn devices such as smart helmets or augmented reality glasses can provide context such as the active work task and indicators related to attention, while infrastructure and robot sensors can provide information

about corridor context and environmental conditions. Worker tasks and the associated situation-specific visibility and awareness conditions determine which assumptions about worker motion are plausible in a corridor situation, and therefore whether a smaller worker risk area can be justified. In contrast, the robot's own motion capabilities are controlled by its software by design, partly by the SINADRA runtime mechanism itself, and are thus not the focus of the modeling performed in this section.

Methodological steps carried out in the case study. Within the overall SINADRA evaluation, this case study is used primarily to assess the practicality of the design-time SINADRA engineering method in an industrial, non-automotive domain, and to provide supporting evidence that selected results can be carried into an executable runtime realization. In the case study, the work started from the hazardous situation of unsafe distance between a robot and a worker in corridor interaction, with collisions as the accident to be avoided. Based on this scope, a structured environment model was created for the relevant warehouse context, including the actor roles, corridor-relevant worker tasks, and the two corridor interaction initiator scenarios of passing and approaching from behind. Building on these inputs, critical behavior cause trees were constructed with a focus on human worker behavior in the selected initiator scenarios. The resulting behavior risk scenarios were then used to define risk implausibility hypotheses expressed through observable situation features that reflect worker task context and awareness conditions. For the corridor scenarios, selected hypotheses were refined into Boolean runtime models whose output is a risk implausibility hypothesis validity signal. This signal was used in a demonstrator to switch between predefined modes of dynamic safety constraints that govern the worker risk area. The case study therefore yields both design-time artifacts and an executable runtime realization for a limited set of corridor interactions.

Study procedure. The study was carried out in collaboration with the industry experts through a sequence of focused scoping and modeling activities. First, the sorting area and corridor interactions were selected as the target context due to their relevance for close human-robot operation. Hitachi provided domain inputs such as an ontology for warehouse concepts and descriptions of task flows that were used to ground the modeling content. Based on these inputs, joint workshops were used to elicit task-dependent worker capabilities and to identify candidate situation features that influence worker awareness and behavior in corridor interactions. The SEM and the CBCTs were then produced and reviewed in joint sessions relative to current practice for hazardous situation concretization in Hitachi. Finally, a ROS and Gazebo demonstrator was implemented to illustrate feasibility. The demonstrator contrasts a baseline configuration based on conservative assumptions with a configuration that applies dynamic safety constraints driven by the runtime validity signals for passing and overtaking.

Table 5.2 Structured Environment Model (SEM) excerpt for the hazardous situation “unsafe distance between a human worker and an AMR during co-located operation in narrow corridors.”

SEM element	Passing in narrow corridor	Approaching from behind with overtaking intent
Abstract Hazardous Situation	Unsafe Distance between AMR and human worker during co-located operation in narrow corridors	
Task Drivers giving rise to initiator scenarios concretizing hazardous situation	**HW**: repeated “move to point” steps between work points around inspection/lift actions. **AMR**: “move to point” toward the same points for receive/unload actions. **Result**: opposing flows on the same corridor segment.	**HW**: speed reduction or brief stops tied to inspect box, lift box to AMR, and carrying after pick-up. **AMR**: continues “move to point” toward receive/unload points. **Result**: same-direction approach with speed difference.
Environment constraints	Narrow corridor geometry, limited lateral clearance, short mutual time window to coordinate passing	Narrow corridor geometry, limited lateral clearance, limited space to step aside during overtake, rear approach reduces early detection
Initiator scenario	HW and AMR enter the same corridor segment from opposite directions	AMR approaches HW from behind in the same corridor direction with overtaking intent
Relevant agent capabilites (human worker)	**Focus**: attend to corridor traffic while moving between points. **Perceive**: detect AMR early, localize oncoming direction and relative position. **Understand**: anticipate near-term AMR path through the segment. **Act**: execute timely walking adjustment to keep separation.	**Focus**: attend to relevant cues despite handling/carrying/inspection. **Perceive**: detect AMR from behind, localize closing distance. **Understand**: anticipate overtaking timing and where the pass may occur. **Act**: execute response quickly enough despite task constraints and limited space.
Influences affecting capabilities	Task state, cognitive load, occlusion, presence of other workers, carrying a load, AMR speed/acceleration, HW speed, relative proximity, planned path	Carrying a load (field of view limits), occlusion, cognitive load, presence of other workers; AMR speed/acceleration, HW speed, relative proximity, corridor geometry (no lateral buffer)

5.1.1 Structured environment modeling

The application of structured environment modeling in the warehouse case study resulted in a scoped and concretized situation model for the hazardous situation “unsafe distance between AMR and human during co-located operation in narrow corridors”. The SEM makes explicit which parts of the

warehouse environment and which human–task conditions matter for this hazard in the targeted setting. It captures (1) the involved actor roles, (2) the task context that drives co-location, (3) two corridor interaction initiator scenarios used as entry points for later CBCT analysis, and (4) a human worker capability model structured by the capability types Focus, Perceive, Understand and Act. Table 5.2 lists an excerpt of the SEM produced in the case study. The following paragraphs briefly explain how the model was created step by step.

Use case scope and information basis

The study focuses on the warehouse sorting area and its narrow corridor network, because this is where co-located operation and close encounters between workers and robots occur frequently. To understand the specific initiator scenarios and to enable the analysis of warehouse situations, Hitachi provided two inputs: a warehouse ontology that captures the relevant domain concepts (Table 5.3) and the cross-role task decomposition of warehouse tasks for different actors (Figure 5.2). The full ontology is confidential; therefore Table 5.3 lists an excerpt to understand the nature and abstraction level of ontology items.

Table 5.3 Warehouse Ontology Excerpt.

Category	Example ontology items in category
Digital Information	Warehouse Management System instructions (tasks), Blockage control instructions, ….
Environmental condition	Road surface, Illumination, Environment Noise, …
Dynamic object	Pedestrians (system administrator, warehouse workers, others), Autonomous / manned forklifts, Autonomous mobile robots, Automated guided vehicles, Load or cart left by someone in aisle, …
Side structures	Item Racks, …
Warehouse network and guidance objects	Crossed aisles, Split aisles, physically separated aisles with two directions, Warning Lights for forklifts entering intersections, Door locations, …

To make task-driven encounter conditions between AMR and human worker explicit, the SEM integrates a task decomposition across roles as shown in Figure 5.2. The decomposition highlights that corridor encounters are not random: they are induced by repeated cycles of (1) moving to defined points and (2) handling actions at points (e.g., pick up / unload boxes, inspect boxes, lift boxes to an AMR or receive boxes at the AMR).

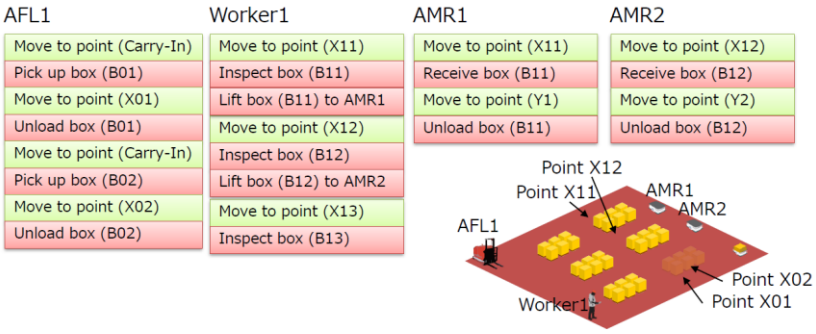


Figure 5.2 Exemplary cross-role task decomposition for warehouse operation. AMR = Autonomous Mobile Robot, AFL = Autonomous Forklift, Worker = Human worker.

The task decomposition in combination with ontology items therefore supports the analysis of:

(1) **Where co-location is expected:** work points (handover and handling points) and the corridor segments that connect them.
(2) **When human worker capabilities might be unavailable:** during inspection and lifting actions and while carrying boxes, which can change view direction, field of view, and reaction time.

Initiator scenarios

Based on the actor task decomposition and warehouse ontology, two corridor interaction initiator scenarios were derived:

(1) **Passing in a narrow corridor.** In this scenario, HW and AMR traverse the same corridor segment in opposite directions.
(2) **Approaching from behind with overtaking intent.** In this scenario, an AMR approaches a slower HW from behind in the same corridor direction. The scenario arises when HW speed is temporarily reduced due to task context like inspection or carrying a box, while the AMR continues toward its next receive/unload point. The task decomposition is important here, because it makes slowdown reasons concrete: inspection and carrying boxes are recurring actions tied to specific points and box handling steps, and therefore recurring causes for speed changes and brief pauses.

Capability models

For each initiator scenario, the SEM includes a human worker capability model stated in terms of Focus, Perceive, Understand and Act. In each category, Table 5.2 lists initiator-scenario specific capabilities in the “Relevant agent capabilities” row and concrete situation elements potentially affecting the capabilities in the “Influences affecting capabilities” row.

For the **passing** scenario, the SEM assumes that the worker (a) keeps attention on corridor traffic while moving between work points (Focus), (b) detects the AMR early and correctly localizes that it is oncoming (Perceive), and (c) anticipates the near-term AMR path through the shared corridor segment (Understand). Based on this, the worker chooses and executes a passing behavior that preserves separation despite the narrow corridor and short coordination window (Act).

For the **overtaking** scenario, the SEM places emphasis on the fact that perception is harder because the approach is from behind and early visual cues are limited. The worker is therefore assumed to (a) keep attention to relevant cues even while handling/carrying/inspection activities (Focus), (b) detect the AMR from behind and localize the closing distance (Perceive), and (c) anticipate overtaking timing and where the pass may occur given corridor geometry (Understand). The expected response is executing a behavior that supports safe separation by holding a stable walking line without stepping in the AMR path (Act).

Two points are important for the evaluation: First, the *capabilities are scenario-specific*: the same abstract hazardous situation "unsafe distance in narrow corridors" is concretized into *different* required worker capabilities depending on whether the encounter is head-on or from the rear. Second, the SEM makes explicit that these *capabilities depend on different environment conditions*, rather than being generic statements about "attentive workers." This shows that the SEM process is a necessity to scope the later CBCT analysis of the causes and effects of capability deviations like unavailability for an abstract hazardous situation.

SEM Industry expert feedback

In the joint modeling sessions, Hitachi experts highlighted that the SEM artifact provided immediate practical value because it consolidated corridor interaction situations and worker capabilities that are otherwise spread across informal operational descriptions. They emphasized that the explicit linkage between ontology/task information, initiator scenarios, corridor constraints, and required capabilities helped align the engineering discussion early and reduced ambiguity about what exactly is assumed about worker behavior in each corridor encounter type.

5.1.2 Critical behavior cause tree modeling

The application of the CBCT analysis and modeling processes resulted in a CBCT for critical behavior of human workers (Figure 5.3) with two SubCBCT models related to the two initiator scenarios: one SubCBCT *IS-PASS* for the passing scenario and *IS-OVTK* for the overtaking scenario. Within the case study scope, these two SubCBCTs address the corridor hazardous situation

HS1: unsafe distance between human worker (HW) and autonomous mobile robot (AMR) in narrow corridors. The human worker CBCT serves two purposes for the evaluation: First, it makes the causal structure of agent critical behavior explicit and reviewable through modeled logical behavior risk scenarios. Second, it provides the basis for verifying via minimal cut set analysis that selected risk implausibility hypotheses (RIHs) are justifiably safe to use for safety constraint adaptation in the context of the system safety concept. The CBCT modeling activities and minimal cutset analyses in the remainder of this subsection were conducted with the SINADRA modeling feature in the *safeTbox*[8] safety engineering tool. In the following, the CBCT results will be explained along the passing scenario. The overtake scenario is structurally very similar, therefore the respective CBCT models and minimal cut sets are located for review in Figure C.1 and Figure C.2 as well as Table C.1 in Appendix C.1.

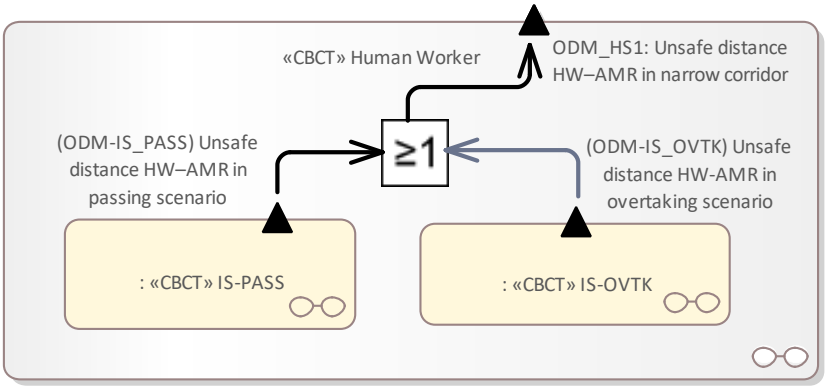


Figure 5.3 Top-Level Decomposition of Human Worker CBCT with the output deviation mode "ODM_HS1: Unsafe distance HW-AMR in narrow corridor". The hazardous situation occurs if a critical human worker behavior occurs in either of two initiator scenarios. The realization of the SubCBCT *IS-PASS* is located in Figure 5.4, the realization of SubCBCT *IS-OVTK* is located in Figure C.1 in Appendix C.1.

The realization of the SubCBCT *IS-PASS* is shown in Figure 5.4. For space reasons, the figure shows the final model *after* adding behavior risk scenarios *and* behavior risk measures. Given the hierarchical nature of the model, the realization of the four capability SubCBCTs is shown in Figure 5.5. In the following, the step-wise analysis and modeling is briefly described with reference to the four CBCT method steps in Section 4.2.2.

In Step 1, the **top-level CBCT structure** is derived from the SEM initiator scenarios and agent capabilities. The result contains two initiator scenario SubCBCTs *IS-PASS* and *IS-OVTK* containing the scenario presence indicators

[8] safeTbox safety engineering tool developed at Fraunhofer IESE: https://www.safetbox.de/

(PI_x) with four capability SubCBCTs (Focus, Perceive, Understand, Act) in each representing the capability types modeled for the scenarios in SEM.

In Step 2, the **behavior risk scenarios** are modeled supported by the deviation type guidewords and warehouse ontology and finally integrated into the CBCT logic via capability deviations (CD_x), agent deficiencies (AD_x) and situation deviation triggers (SDT_x). For *IS-PASS*, this resulted in four exemplary behavior risk scenarios involving a representative CD-AD-SDT combination for each capability type. These scenarios are represented in the minimal cut sets in Table 5.4, if risk implausibility hypotheses and safety concept measures are mentally removed from the individual cut sets (the table lists minimal cut sets after behavior risk measures are added).

For example, the human worker focus capability demands attention to the front space as a precondition to perceive the oncoming AMR, understand its intention to pass and act accordingly. A critical deviation of that capability is impaired attention (CD_PASS_FOCUS_1) caused by the combination of an active agent distraction (AD_CC_1, Figure 5.5) triggered by an immersive box carrying task (SDT_F_1). If the attention is impaired, all follow-up capabilities are irrelevant, because executing a safe behavior choice cannot be concluded if the HW is not aware of the coming AMR. The CBCT structure reflects this through the propagation to ODM-IS-PASS via the other capability SubCBCTs.

In the case study, more behavior risk scenarios were found based on a more detailed warehouse ontology. For the purpose of keeping the model tractable for understanding and evaluating the application of the SINADRA core concepts, the four scenarios were intentionally selected and extracted.

In Step 3, the **behavior risk measures** are modeled by means of risk implausibility hypotheses (RIH_x) and safety concept measures (M_x). *RIH_PASS* “Human Worker is aware of the AMR” intercepts the behavior risk scenarios related to *Focus* and *Perceive* capabilities. If we can detect this condition, deductive reasoning tells us that attention and perception must have worked properly for the human worker. However, even if *RIH-PASS* is valid, risk of unsafe distances influenced by HW behavior cannot be concluded due to the remaining behavior risk scenarios related to *Understand* and *Act* capabilities. Since mitigating the human worker’s agent deficiencies in *Understand* (tendency to misread intent / situation meaning, *AD_INT_1*) and *Act* (overconfidence in AMR stopping HW in path / too high risk taking, *AD_ACT_1*) is hardly possible via runtime monitoring, system safety measures are defined. Misreading intents can be addressed via training the human worker in what AMRs usually do in the warehouse and which behaviors are expected (*M_TRAINING*). The remaining *Act* deficiency is addressed via a safety-relevant AMR design measure. In this scenario, the human worker successfully perceives the AMR and interprets the situation correctly, but intentionally ignores it thinking the AMR will stop in any case (*AD_ACT_1*).

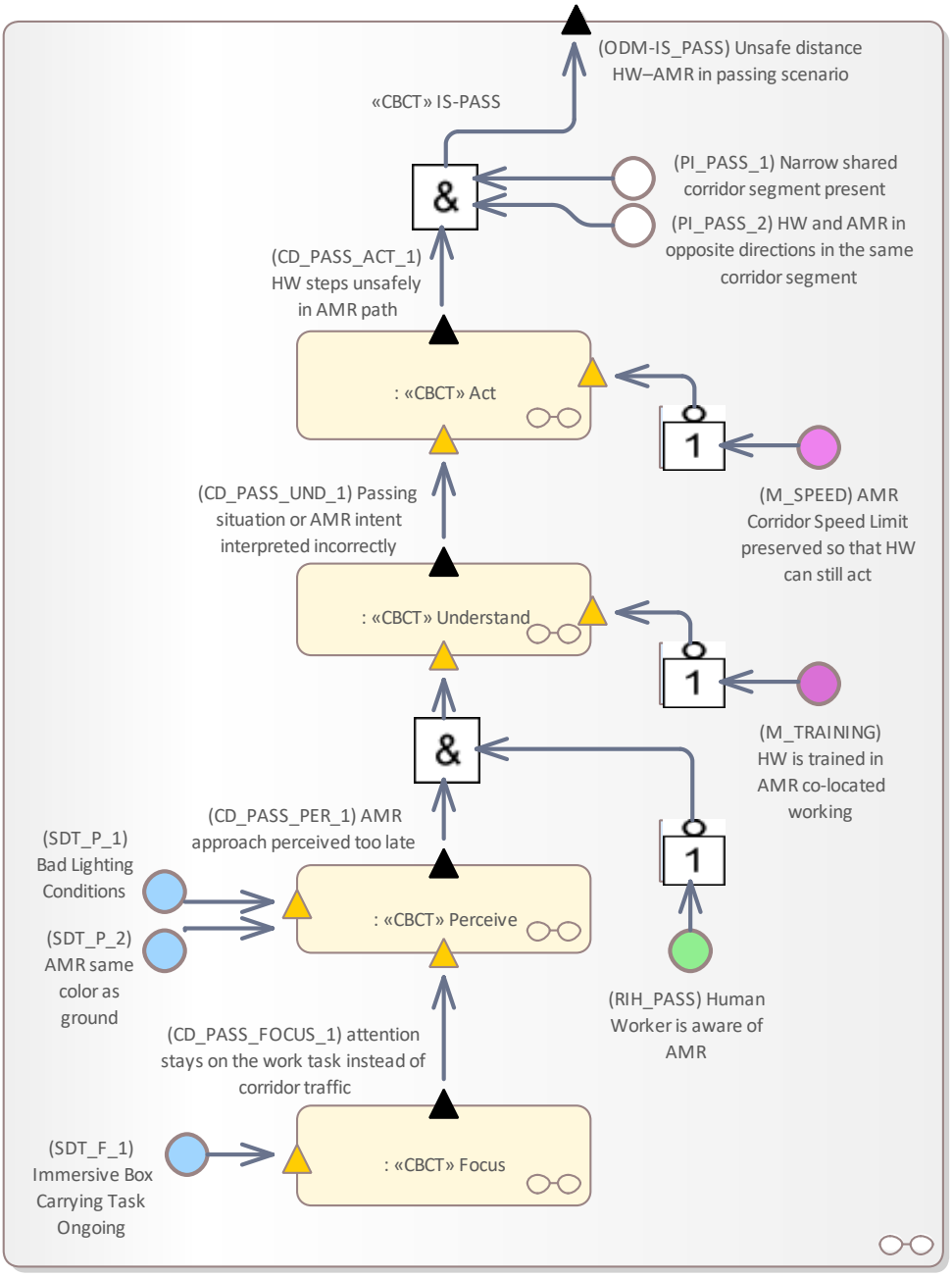


Figure 5.4 Realization of SubCBCT *IS-PASS* consisting of the output deviation mode (ODM_IS-PASS), capability SubCBCTs, capability deviations (CD_x), scenario presence indicators (PI_x), situation deviation triggers (SDT_x), risk implausibility hypotheses (RIH_x), safety concept measures (M_x). Agent deficiencies (AD_x) are not displayed, because they are part of the realization of the capability SubCBCTs in Figure 5.5.

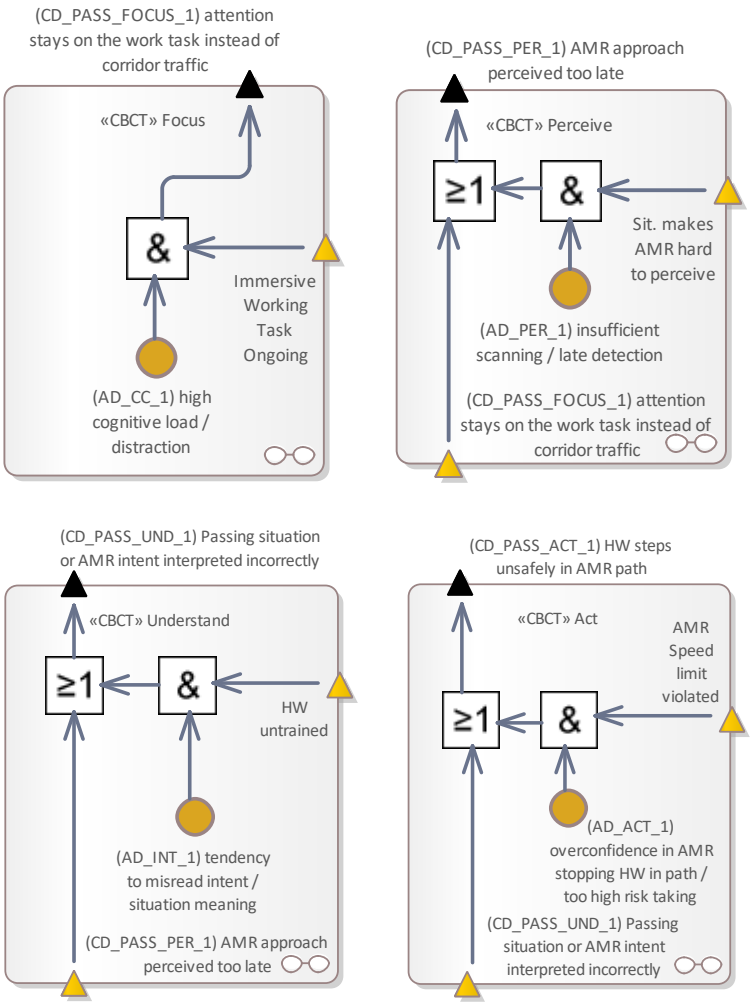


Figure 5.5 Realization of four SubCBCTs *Focus, Perceive, Understand, Act* of initiator scenario CBCT *IS-PASS*.

To increase HW controllability to still react to the AMR and reduce potential crash severity, the corridor speed needs to be limited for AMRs to an appropriate maximum value.

As already mentioned, the CBCT for the overtaking scenario is very similar to the passing scenario (Figure 5.6). The only major difference lies in the different focus and perceive capabilities, because the AMR is approaching the human worker from the rear area in the same direction. Consequently, the risk implausibility hypothesis *RIH_OVTK* focuses on the AMR awareness in the rear space. Although similar to *RIH_PASS,* this is decisive for the runtime

model transformation as different observed features are required to monitor awareness in both scenarios.

Overall, the behavior risk measure examples demonstrate that implausibility or absence of behavior risk scenarios cannot be argued based on monitored runtime risk implausibility hypotheses alone, but that a tight integration with system safety measures is indispensable. CBCTs were designed to make this integration explicit. Thus, in contrast to state-of-the-art approaches to argue adaptable safety monitor assumptions, CBCTs can deliver the *combined* evidence that monitored risk implausibility hypotheses *and* required safety concept measures justify safe permissiveness improvements.

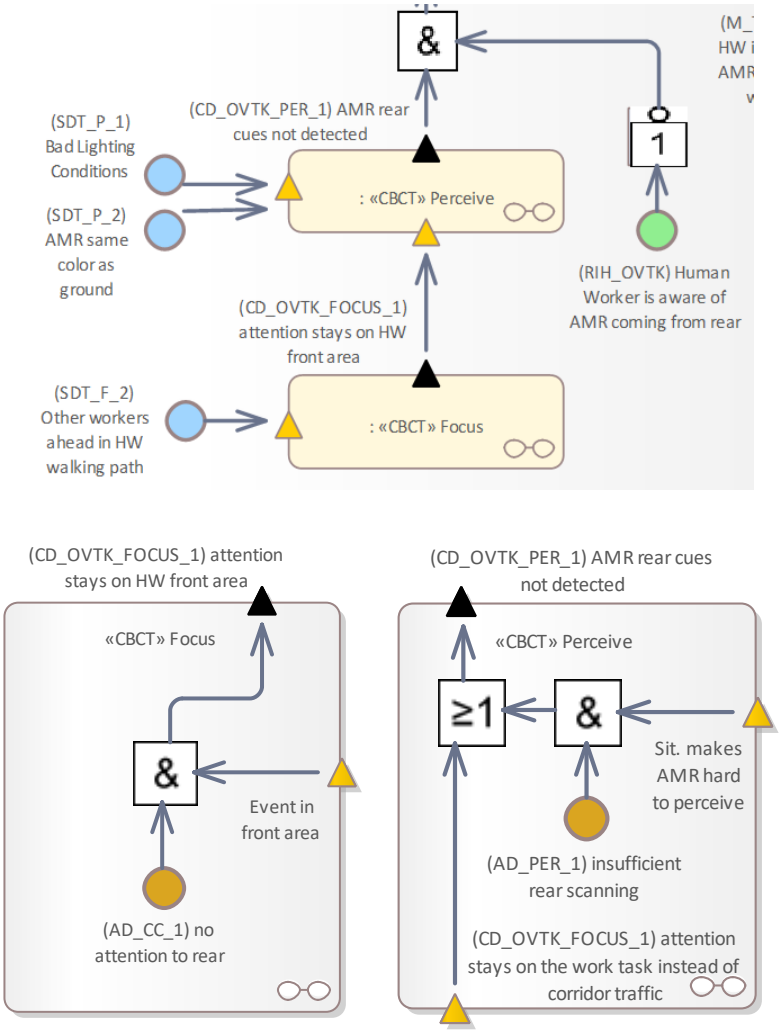


Figure 5.6 Differences in the realization of initiator SubCBCT *IS-OVTK* and its *Focus* and *Perceive* capability SubCBCTs.

Table 5.4 Four minimal cut sets of “ODM-IS-PASS: Unsafe distance HW-AMR in passing scenario”.

#	ID	Positive	Name	Event Type	Parent path
1	PI_PASS_2	True	HW and AMR in opposite directions in the same corridor segment	Presence Indicator	«CBCT» Human Worker :: «CBCT» IS-PASS
2	SDT_F_1	True	Immersive Box Carrying Task Ongoing	Situation Deviation Trigger	«CBCT» Human Worker :: «CBCT» IS-PASS
3	PI_PASS_1	True	Narrow shared corridor segment present	Presence Indicator	«CBCT» Human Worker :: «CBCT» IS-PASS
4	AD_CC_1	True	high cognitive load / distraction	Agent Deficiency	«CBCT» Human Worker :: «CBCT» IS-PASS :: «CBCT» Focus
5	RIH_PASS	False	Human Worker is aware of AMR	Risk Implausibility Hypothesis	«CBCT» Human Worker :: «CBCT» IS-PASS

#	ID	Positive	Name	Event Type	Parent path
1	SDT_P_2	True	AMR same color as ground	Situation Deviation Trigger	«CBCT» Human Worker :: «CBCT» IS-PASS
2	SDT_P_1	True	Bad Lighting Conditions	Situation Deviation Trigger	«CBCT» Human Worker :: «CBCT» IS-PASS
3	PI_PASS_2	True	HW and AMR in opposite directions in the same corridor segment	Presence Indicator	«CBCT» Human Worker :: «CBCT» IS-PASS
4	PI_PASS_1	True	Narrow shared corridor segment present	Presence Indicator	«CBCT» Human Worker :: «CBCT» IS-PASS
5	AD_PER_1	True	insufficient scanning / late detection	Agent Deficiency	«CBCT» Human Worker :: «CBCT» IS-PASS :: «CBCT» Perceive
6	RIH_PASS	False	Human Worker is aware of AMR	Risk Implausibility Hypothesis	«CBCT» Human Worker :: «CBCT» IS-PASS

#	ID	Positive	Name	Event Type	Parent path
1	PI_PASS_2	True	HW and AMR in opposite directions in the same corridor segment	Presence Indicator	«CBCT» Human Worker :: «CBCT» IS-PASS
2	PI_PASS_1	True	Narrow shared corridor segment present	Presence Indicator	«CBCT» Human Worker :: «CBCT» IS-PASS
3	AD_INT_1	True	tendency to misread intent / situation meaning	Agent Deficiency	«CBCT» Human Worker :: «CBCT» IS-PASS :: «CBCT» Understand
4	M_TRAINING	False	HW is trained in AMR co-located working	Safety Concept Measure	«CBCT» Human Worker :: «CBCT» IS-PASS

#	ID	Positive	Name	Event Type	Parent path
1	PI_PASS_2	True	HW and AMR in opposite directions in the same corridor segment	Presence Indicator	«CBCT» Human Worker :: «CBCT» IS-PASS
2	PI_PASS_1	True	Narrow shared corridor segment present	Presence Indicator	«CBCT» Human Worker :: «CBCT» IS-PASS
3	AD_ACT_1	True	overconfidence in AMR stopping HW in path / too high risk taking	Agent Deficiency	«CBCT» Human Worker :: «CBCT» IS-PASS :: «CBCT» Act
4	M_SPEED	False	AMR Corridor Speed Limit preserved so that HW can still act	Safety Concept Measure	«CBCT» Human Worker :: «CBCT» IS-PASS

In Step 4, minimal cut set analysis delivers this evidence (Table 5.4): Four behavior risk scenarios in *IS-PASS* are appropriately addressed by the combination of one risk implausibility hypothesis *RIH_PASS* and two safety concept measures *M-TRAINING* and *M-SPEED*. Every minimal cut set is addressed by either a risk implausibility hypothesis or a safety concept measure. This supports the safety verification claim that building a runtime adaptation scheme based on the risk implausibility hypotheses is justified.

The trustworthiness of the cutset analysis results depends on the quality of the SEM and CBCT models. Although an algorithmic verification of SEM and CBCT was not carried out, the basic model quality can be argued based on the following analytic evidence:

- the warehouse ontology and task decomposition model was used to guide the identification of initiator scenarios and scenario-specific human worker capabilities in the SEM model
- the CBCT base structure mirrors the initiator scenarios and capabilities of the SEM model by design and therefore implies architectural coverage regarding the environment aspects relevant for the hazardous situation
- the capability deviation analysis in CBCT was guided by the capability-specific deviation type guidewords to achieve capability deviation type coverage
- the warehouse ontology was used to guide the identification of potential situation deviation triggers contributing to capability deviations in the CBCT

5.1.3 Runtime model transformation

For each of the two corridor initiator scenarios, the case study resulted in an explicit Boolean runtime model that computes a risk implausibility hypothesis validity signal from perceived situation features (Figure 5.7).

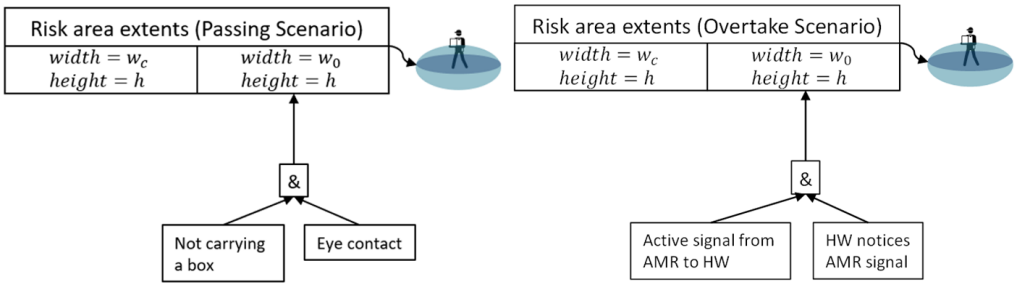


Figure 5.7 Boolean runtime models to operationalize SINADRA risk implausibility hypotheses related to the human worker in corridor passing (left) and overtaking (right) scenarios. The models define a conservative risk area width w_c and a reduced width w_0. The risk area width is switched based on the risk implausibility hypothesis validity dependent on the presence of situation features. Figure adapted from (Reich et al., 2022), with permission from Springer Nature. © 2022 The Author(s), under exclusive licence to Springer Nature Switzerland AG.

For the passing scenario, the runtime model captures the situation in which the behavior risk scenarios “impaired human worker attention” and “too late perception of oncoming AMR” are implausible. The implemented Boolean model for the CBCT risk implausibility hypothesis *RIH-PASS* evaluates two observable features that were agreed with the industrial partners as relevant and feasible in the warehouse setting. First, it evaluates whether the worker is currently carrying a box, which is used as an operational proxy for occlusion in front of the worker. Second, it evaluates whether eye contact is present as an indicator of HW awareness of the AMR in the corridor encounter. The model produces a validity signal that becomes true when the worker is not carrying a box and eye contact is observed. When the validity signal is true, the controller is permitted to select a reduced lateral risk area mode for the worker enabling a permissive AMR passing action. When the signal is false, the controller retains the worst-case risk area mode.

For the overtaking scenario, the runtime model operationalizes the monitoring logic of the CBCT risk implausibility hypothesis *RIH-OVTK and* evaluates whether an attention cue was issued and whether a corresponding worker reaction was observed. An active audio-visual attention cue issued by the AMR represents the operational mechanism used in the prototype to indicate AMR presence and overtake intent. The observed reaction represents a situation feature that indicates that the worker perceived and acknowledged the approaching robot. This feature is concretely observed as a change in the human worker’s eye gaze angle shortly after the audio-visual cue was signaled. The *RIH-OVTK* validity signal becomes true when both features hold. When it is true, a reduced lateral risk area mode can be selected to allow a more permissive overtake. When it is false, the system remains in the worst-case mode that assumes the worker may be unaware.

Discussions with the industrial partners about perception feasibility of these features resulted in the consensus that HMI devices like smart helmets or augmented reality glasses worn by human workers can indicate eye gaze direction/angle and present work tasks. In addition, infrastructure and AMR-local sensors can plausibilize the presence of carried boxes in addition to retrieving this information from the current working task.

5.1.4 Simulative feasibility demonstration

The SINADRA runtime models for both risk implausibility hypotheses were transformed into executable robot operating system (ROS) software components, which can perform the Boolean inference. The software components were integrated into the symbiotic safety system architecture (Figure 5.8), whose realization is based on ROS.

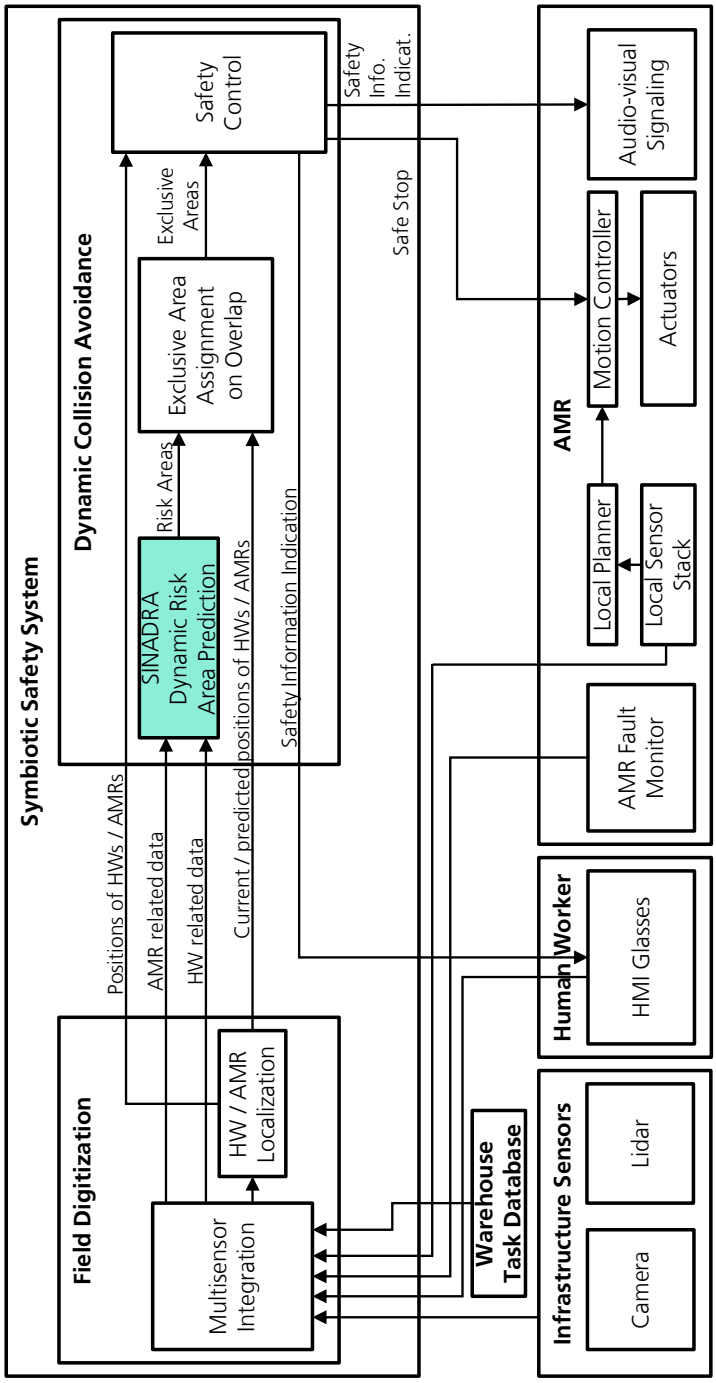


Figure 5.8 Integration architecture of the SINADRA runtime software component into the Symbiotic Safety System Architecture. Adapted from Fig. 5 in (Reich et al., 2022), with permission from Springer Nature. © 2022, under exclusive licence to Springer Nature Switzerland AG. Change: “Dynamic Risk / Capability Area Prediction” was renamed to “SINADRA Dynamic Risk Area Prediction.”

For the runtime feasibility demonstration, simple AMR and human worker models were created in the Gazebo simulator for passing and overtaking scenarios. Both scenarios were simulated and the AMR was controlled by the above architecture with integrated SINADRA runtime components. Videos of the demonstration scenarios are publicly available on YouTube for the passing[9] and overtake[10] scenarios. The relevant with/without SINADRA moments are shown in Figure 5.9.

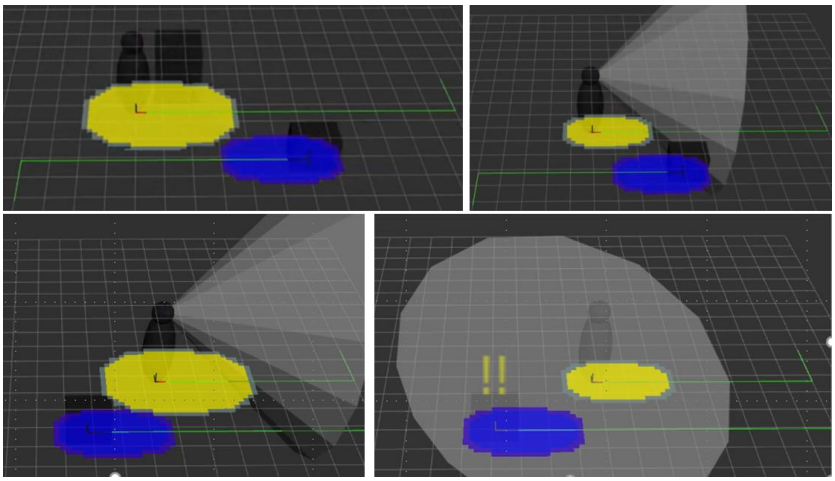


Figure 5.9 SINADRA runtime feasibility demonstration for passing (top) and overtaking (bottom) scenarios. The scenes on the left show the situations where the risk implausibility hypotheses are not confirmed leading to the worst-case size of the risk area around the human worker. In consequence, the AMR needs to stop until the human worker has passed (passing scenario) and needs to stay behind the human worker operating at a lower speed (overtaking scenario). The scenes on the right show the situations where the risk implausibility hypotheses are confirmed based on the SINADRA runtime inference. In consequence, the risk area width is decreased in both scenarios leading to more efficient passing and overtaking.

The demonstration shows an observable difference between conservative and situation-dependent operation in both corridor scenarios, driven only by whether the risk implausibility hypothesis validity signal turns true during the encounter. This provides feasibility evidence that the design-time hypotheses can be realized as executable runtime models that switch a safety monitor parameter related to human worker behavior risk, while keeping the worst case whenever the signal is missing or not satisfied.

5.1.5 Analytic permissiveness evaluation

In addition to the qualitative demonstration runs, the case study includes an analytic comparison that quantifies the expected permissiveness effect of

[9] Passing scenario demonstration video: https://www.youtube.com/watch?v=N3gYsAMkSVk

[10] Overtaking scenario demonstration video: https://www.youtube.com/watch?v=0Wlv1QXlkmw

switching between conservative and reduced risk area modes. The analysis is published in (Reich et al., 2022) and compares the expected time to traverse a corridor interaction under a worst-case configuration versus a configuration that allows reduced risk area modes when the corresponding hypothesis is valid. The intent of this analysis is not to claim measured operational performance. Its function in the evaluation is to provide a transparent, parameter-driven estimate of how often a hypothesis must be valid to yield a noticeable permissiveness effect in typical corridor encounters. The following paragraphs describe for both scenarios the analytic models with selected parameters. The analytic evaluation results in 2% (21%) improvement in AMR passing (overtaking) traveling time when using SINADRA compared to only using worst-case assumptions.

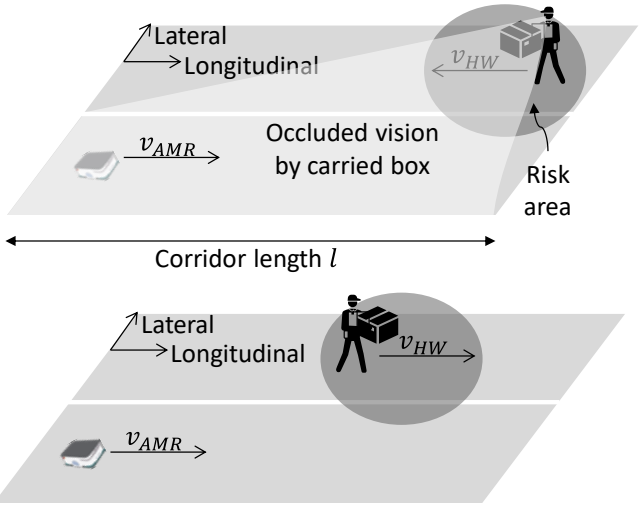


Figure 5.10 Passing (left) and overtaking (right) scenarios evaluated in the case study. Adapted from Fig. 6 in (Reich et al., 2022), with permission from Springer Nature. © 2022 The Author(s), under exclusive licence to Springer Nature Switzerland AG.

In the **passing scenario**, an AMR and a human worker (HW) move with speeds v_{AMR} and v_{HW} in opposite directions along a corridor with length l (Figure 5.10, left).

Two modes with different assumed width extents are modeled for the risk area models. The worst-case risk area extent is always selected with width w_c unless the risk-decreasing situation features “Not carrying a box” and “Eye contact” are present, where we assume w_0 to be the risk area’s width. Both situation features in combination indicate the absence of occlusion and a basic level of awareness, leading to the hypothesis that the human worker notices the AMR and will take this into account in order to behave safely. This turns the situation into a non-worst-case situation and represents efficiency potential.

As an evaluation benchmark, we computed the average AMR corridor passing time t_{WC}^{0} (Eq. 1) by assuming the constant presence of the HW risk area width w_c, where the AMR needs to stop and can only continue to move after the HW has stepped out of their capability area. For this computation, assumptions are required for the likelihood of an HW being present in the corridor simultaneously with the AMR $P(HW)$, the speeds of the AMR v_{AMR} and of the HW v_{HW}, the length of the AMR d_{AMR}, and the corridor length l.

$$t_{WC}^{0} = P(\overline{HW})\left(\frac{l}{v_{AMR}}\right) + P(HW)\left(\frac{l}{v_{AMR}} + \frac{d_{AMR}}{v_{HW}}\right) \qquad (1)$$

Assuming $P(HW) = 0.7,\ v_{HW} = 1.6\frac{m}{s},\ v_{AMR} = 5.0\frac{m}{s},\ d_{AMR} = 0.85m,\ l = 42m$, the benchmark average AMR corridor passing time is $t_{WC}^{0} = 8.77s$.

In our best-case scenario, the absence of boxes being carried and the presence of eye contact are taken into account to adapt the risk area extent to w_0, so that no stopping is required for the AMR, i.e., it can drive with maximum v_{AMR} along the full corridor length. In addition to the parameters of Eq. 1, further assumptions about the likelihood that an occlusion is present $P(OC)$ and the conditional probability that eye contact is present given no occlusion $P(EC|\overline{OC})$ are required for the average passing time t_{DYN}^{0}, which is given by Eq. 2.

$$\begin{aligned} t_{DYN}^{0} = {} & [P(\overline{HW}) + P(HW)P(EC|\overline{OC})P(\overline{OC})]\left(\frac{l}{v_{AMR}}\right) \\ & + P(HW)[P(\overline{EC}|\overline{OC})P(\overline{OC}) + P(OC)]\left(\frac{l}{v_{AMR}} + \frac{d_{AMR}}{v_{HW}}\right) \end{aligned} \qquad (2)$$

By assuming similar parameter values for $P(HW)$, v_{HW}, v_{AMR}, d_{AMR}, l and additionally $P(OC) = 0.5,\ P(EC|\overline{OC}) = 0.8$, the average AMR corridor passing time with dynamic risk area models is $t_{DYN}^{0} = 8.62s$.

For the assumed parameter values, the average passing time decreases by approximately **2% compared to the worst-case benchmark**. The reduction is modest but positive and suggests gains when aggregated over many interactions.

For overtaking in the same direction (Figure 5.10, right), the efficiency gain is stronger. Under worst-case assumptions, the AMR follows the worker at low speed and waits until the worker has covered enough distance to leave the AMR's risk area. With the worst-case assumed risk area around the HW, the AMR cannot overtake the HW with v_{AMR}. Instead, the AMR will constantly

drive behind the HW with a maximum v_{HW} until the end of the corridor. Here, the first term describes the time required by the AMR to reach the point s with its regular speed v_{AMR}. From this point on, it must reduce its speed to that of the HW, ensuring a sufficient distance to the HW over the remaining distance of the corridor. Using the same variables as in Eq. 1, the average passing time of an AMR t_{WC}^{1} is given by Eq. 3.

$$t_{WC}^{1} = P(HW)\frac{1}{l}\int_{o}^{l}\frac{s}{v_{AMR}} + \frac{l-s}{v_{HW}}ds + P(\overline{HW})\frac{l}{v_{AMR}} \tag{3}$$

Assuming the parameter values for $P(HW)$, v_{HW}, v_{AMR}, l to be as in Eq. 1, the average AMR corridor passing time with worst-case assumptions is $t_{WC}^{1} = 14.65s$.

The result of extending Eq. 3 with an additional parameter $P(AW|HW)$ indicating the likelihood of the HW's awareness of the AMR (=successful turn-around after active AMR signaling) leads to the average AMR corridor passing time with dynamic risk area models t_{DYN}^{1} (Eq. 4).

$$\begin{aligned} t_{DYN}^{1} = {} & P(HW)\frac{1}{l}\int_{o}^{l} P(\overline{AW}|HW)\left(\frac{s}{v_{AMR}} + \frac{l-s}{v_{HW}}\right)ds \\ & + P(AW|HW)P(HW)\frac{l}{v_{AMR}} + P(\overline{HW})\frac{l}{v_{AMR}} \end{aligned} \tag{4}$$

Assuming similar parameter values for $P(HW)$, v_{HW}, v_{AMR}, l and additionally $P(AW|HW) = 0.5$, the average AMR corridor passing time with dynamic risk area models is $t_{DYN}^{1} = 11.52s$.

For the assumed parameter values, the average passing time decreases on the order of approximately **21% compared to the worst-case benchmark**. The expected improvement in corridor traversal time in overtaking is substantially larger than in passing.

Together, the simulative demonstration and analytic evaluation results support the SINADRA runtime feasibility evaluation in two complementary ways. The simulation demonstrator shows that hypothesis validity can drive discrete and interpretable changes of dynamic safety constraints in the targeted corridor encounters. The analytic estimates then illustrate why the same mechanism can yield meaningful permissiveness effects in warehouse operations when the validity conditions hold with non-negligible frequency.

5.1.6 Evidence assessment and limitations

This case study provides evidence primarily for **EP1 (method feasibility)** and supporting evidence for **EP2 (runtime feasibility)** in a demonstrator scope. It starts from the abstract hazardous situation "unsafe distance between an autonomous mobile robot (AMR) and a human worker (HW)" with collision as the accident in scope, and targets avoiding unnecessary stops caused by worst-case worker motion assumptions.

Evidence against EP1. EP1a is addressed with a SINADRA application in the warehouse domain with different agents, tasks, and constraints compared to the automotive domain. EP1b is addressed by applying the design-time chain from SEM to CBCT: the SEM scopes the setting (ontology excerpt and task decomposition) and yields two initiator scenarios (passing and overtaking). The CBCT reuses these initiator scenarios and expands them into worker-centered behavior risk scenarios, with explicit risk implausibility hypotheses (RIH) and safety measures used as blockers in the same Boolean structure. For EP1c, the key justification evidence is the minimal cut set analysis: within the modeled scope, the selected hypotheses (RIH_PASS, RIH_OVTK) together with measures (M-TRAINING, M-SPEED) block the minimal cut sets for the modeled passing and overtaking risk scenarios, i.e., no unblocked minimal cut set remains in that model instance. Supporting justification evidence is provided by qualitative SEM and CBCT verification (guided scoping via ontology/task structure, guidewords for deviation coverage, structural reuse from SEM).

Evidence against EP2. EP2a–EP2c are supported by transforming selected RIHs into explicit Boolean runtime models based on observable situation features, integrating them as ROS components into Hitachi's symbiotic safety system, and demonstrating mode switching in simulation, with worst-case behavior as the default when the RIH validity signal is false. An analytic comparison reports an estimated gain in AMR corridor traveling time from using SINADRA of about 2% for passing and 21% for overtaking under stated assumptions.

Limitations and threats to validity. The internal validity is mainly limited by model scope. More behavior risk scenarios were identified, but only a subset was selected for explanation and analysis. Omitted paths could add new cut sets and require additional RIH or measures, which would weaken the sufficiency claim in practice. Therefore, the sufficiency claim holds for the modeled subset and stated assumptions, not as a completeness claim for the full warehouse ODD. The external validity is limited to two corridor interactions and a specific monitoring concept. Different warehouse layouts, tasks, and worker practices can change relevant initiator scenarios and observable features. Assumed sensing options (worker-worn devices or infrastructure sensing) may not be available in other deployments, which can

limit the feasibility of the shown runtime feature set outside the case study context.

5.2 Case Study 2: Urban ADS Driving

This second case study transfers SINADRA from a controlled warehouse domain to urban automated driving scenarios with human-driven vehicles as the main external agents influencing risk. The case study material was published in (Reich et al., 2021) and is complemented by an open implementation of the SINADRA computation pipeline in CARLA, including supplementary material in a public GitHub repository[11]. The supplementary material includes the Python component, exemplary Bayesian network models, OpenScenario scenario files for CARLA ScenarioRunner, and technical documentation of the runtime processing chain and its configuration.

The aim of this case study is to demonstrate runtime feasibility at component level for situation-aware dynamic risk assessment:

(1) inferring behavior intents of other traffic participants from situational cues using Bayesian networks,
(2) transforming intent likelihoods into trajectory distributions using behavior-specific motion models, and
(3) integrating these predictions into the probabilistic integral collision risk computation pipeline from (Eggert, 2018) to achieve a situation-aware dynamic risk estimate.

In contrast to the warehouse case study, this case study does not aim at an end-to-end SINADRA application with SEM and CBCT as the primary work products. Instead, it focuses on the runtime model operationalization and execution chain, and uses cause-effect intent models as an early precursor to CBCT-style reasoning.

Consistent with the adjusted evaluation scope in Table 5.1, the evidence provided here is that the component can identify lower-risk situation instances compared to worst-case intent assumptions at the level of the computed risk output. This is treated as a prerequisite for later safety constraint adaptation, while full safety justification and model verification are out of scope for this study.

5.2.1 Case study setup and produced artifacts

The case study instantiates SINADRA as a Python-based “risk sensor” coupled with the CARLA simulator. The component connects to a CARLA server,

[11] https://github.com/JaRei/sinadra

gathers scene data via the simulator API, and executes the SINADRA processing steps online during scenario execution. The resulting runtime chain is summarized in Figure 5.11.

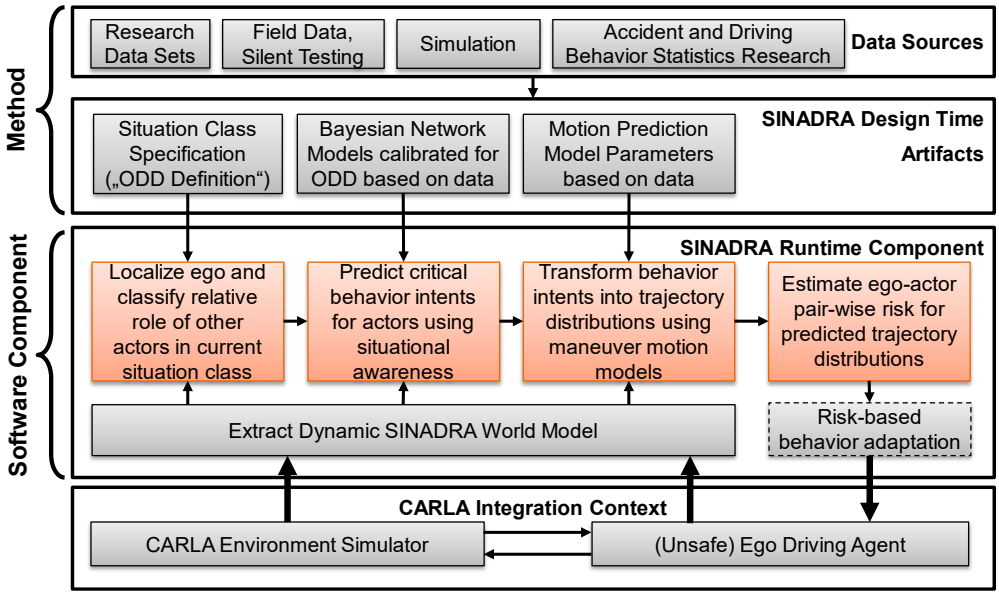


Figure 5.11 SINADRA demonstrator architecture, showing design-time inputs and the runtime chain from ego localization and actor-role detection through intent inference and trajectory generation to ego-actor risk estimation. Adapted from Fig. 1 in (Reich et al., 2021), with permission from Springer Nature. © 2021 Springer Nature Switzerland AG.

The case study contributes the following artifacts to the case study:

- **Situation class detection and actor role classification.** The component localizes the ego vehicle in its current situation class and classifies nearby vehicles into roles that are relevant for the ego vehicle in that context (e.g., front vehicle and side vehicle). These roles provide the basis for selecting which intent models must be evaluated for which actors.
- **Bayesian network models for behavior intent inference.** The case study uses Bayesian networks (BN) to infer discrete behavior intents of other traffic participants from runtime cues. The implementation supports modeling networks in a design tool and transforming them into an executable format for an open-source inference library. A generated configuration wrapper links network structure to runtime evidence provision.
- **Trajectory distribution generation.** For each nearby actor and behavior intent, trajectory distributions are generated using behavior-specific motion models. Rather than producing one deterministic future path per intent, the component generates distributions by sampling probabilistic parameters of the motion models to reflect variability.

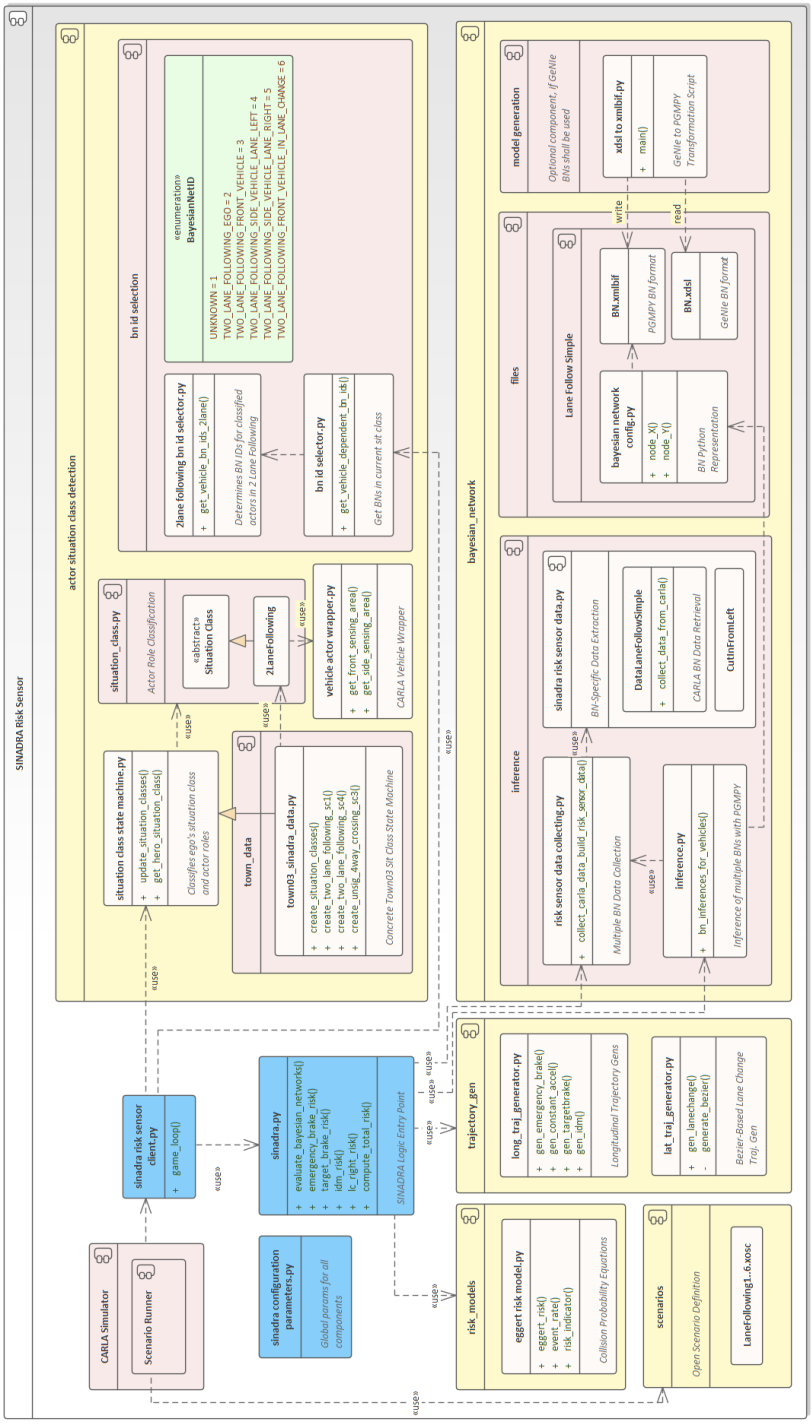


Figure 5.12 Component structure of the CARLA-coupled SINADRA Python client. Originally published in the SINADRA software repository © 2021 Jan Reich.

- **Risk computation and intent-weighted integration.** The predicted ego and actor trajectory distributions are used as input to a probabilistic, trajectory-based collision risk computation. Risk is computed per candidate intent, and a total risk output is formed by weighting behavior-specific risk values with the inferred intent likelihoods.

  The technical architecture and the main component responsibilities of the pipeline implementation are shown in Figure 5.12. In the provided demonstrator, the component runs as a Python client that orchestrates the runtime loop and produces a live risk plot while scenarios are executed in CARLA.

  The driving situations used for demonstration are lane-following scenarios in which the ego vehicle is exposed to relevant behavior variability from other vehicles, including braking variations of a front vehicle and lane change / cut-in behaviors of a side vehicle. The scenario set is specified in the OpenScenario format and executed using the CARLA ScenarioRunner mechanism.

5.2.2 Situation class scoping and actor role classification

The first step of the runtime chain is the ability to determine which parts of the scene are relevant for which intent models. In this case study, this is handled via situation classes and actor role classification. Methodically, situation classes serve the same purpose as the SEM scoping step in Section 4.1: they constrain which environmental context and which relative actor roles are considered when deciding which behavior intents are risk-relevant and therefore need runtime monitoring.

At runtime, the ego vehicle is first localized in its current situation class. Figure 5.13 shows, by example, what such situation classes look like for the Town03 map in CARLA. This situation class provides the context for searching for relevant actors in the ego vicinity and for classifying them into roles expressing relevance for the ego, such as a front vehicle or a side vehicle in an adjacent lane. Role definitions are encoded as conditions that must hold for an actor to be assigned to a role in a given situation class. Concretely, such role conditions are realized as speed-dependent spatial areas around the ego vehicle. If an actor is in that area, it is assigned the role associated with that area. The output of this step is a set of actor-to-role assignments that are used to select which intent inference models must be evaluated for which actor. Figure 5.14 illustrates this role classification step and the mapping to intent models.

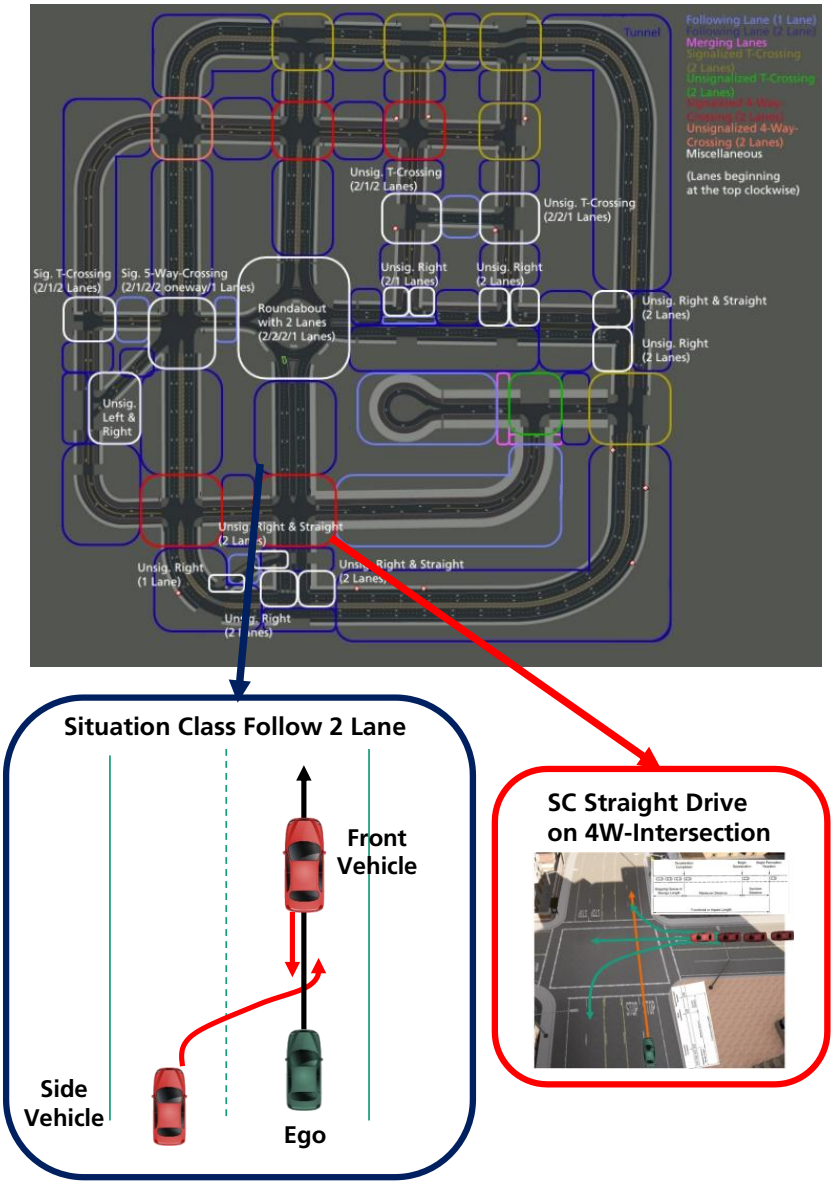


Figure 5.13 Situation class decomposition of the operational design domain, exemplified for the CARLA Town03 map. The ODD is decomposed into situation classes to scope relevant actor roles, for which critical behavior intents driving the risk of hazardous situations can be specified in a concrete situational context. Adapted from the figure originally published in the SINADRA software repository. © 2021 Jan Reich.

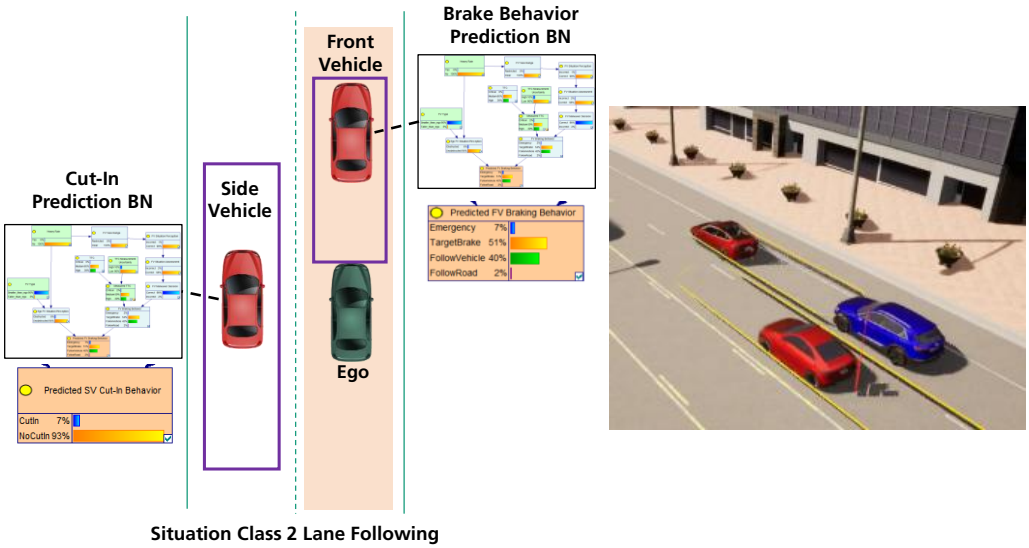


Figure 5.14 Situation class detection and actor role classification with intent-model selection. Ego localization in a situation class sets the context for searching and classifying relevant actors. Classified roles are mapped to Bayesian network identifiers that determine which intent models are evaluated for which actors. Adapted from the figure originally published in the SINADRA software repository. © 2021 Jan Reich.

5.2.3 Runtime model operationalization as Bayesian networks

Now that it is clear from the actor-role mapping which behaviors might become critical in a given driving situation, the second step is the realization of BNs, which can infer the relative likelihoods of the behaviors given the concrete situational context. To determine the relevant situational context for critical behaviors, the cause-effect structure of the possible behaviors needs to be analyzed.

As we saw in the warehouse case study in detail, the SINADRA structured environment and critical behavior cause tree modeling methods can deliver such a verified cause-effect structure. Given the ADS driving case study was chronologically conducted before the SEM and CBCT methods existed in their current form, the relevant situation influences and their causal effects on behaviors were derived through an extensive literature review for braking and lane-change behaviors (Wellstein, 2021). Excerpts of such qualitative causal models are shown for front vehicle braking in Figure 5.15, for side vehicle lane changes in Figure C.3 in Appendix C.2. The models in (Wellstein, 2021, Chapter 5) are much more extensive.

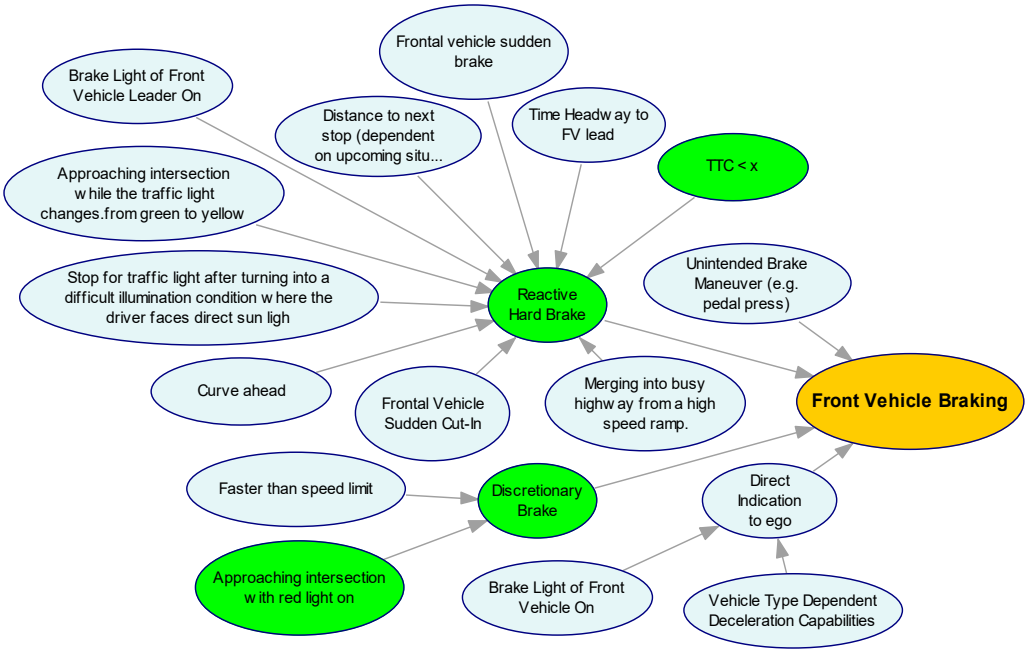


Figure 5.15 Qualitative causal influences for front vehicle braking behavior.

In Figure 5.15, different options of discretized braking behavior are shown along with a set of causal factors for them. For instance, *Reactive Hard Brake* is a different flavor of braking than *Discretionary Brake* or *Unintended Brake* and each of the three has different situational contexts in which they are caused. This matches the idea behind initiator scenario identification in the SEM process (Section 4.1.2).

To realize a minimal valid SINADRA model for risk implausibility hypothesis detection, one influence scenario per behavior intent to be differentiated is selected. For instance, the green nodes in Figure 5.15 show that *Discretionary Brake* is likely when the front vehicle is *Approaching intersection with red light on*, while *Reactive Hard Brake* is likely to occur when the time-to-collision *TTC* between a vehicle and its front vehicle is smaller than a threshold. The task of the BN runtime model is to monitor the presence or absence of such scenarios and in consequence determine the relative likelihood of critical behavior intents. Note that in the context of SINADRA, it is valid to pick only one scenario, because the default is operating based on worst-case assumptions. If a low likelihood of one scenario is inferred, we can use this knowledge to relax safety constraints only in this detected scenario.

In the terminology of Section 4.3, the BN output can be read as a probabilistic runtime validity signal for an intent-focused risk implausibility hypothesis of the form “the worst-case behavior intent is unlikely in the current situation”. This plays the same functional role as a Boolean hypothesis validity signal in

the warehouse case study, but it supports relative behavior likelihoods and explicit uncertainty handling.

Figure 5.16 shows an example BN that realizes this concept for front vehicle braking, an example BN for side vehicle cut-in behavior is located in Figure C.4 in Appendix C.2. The node front *FV Braking Behavior* discretizes braking behaviors with different levels of criticality for the ego vehicle:

- *Emergency* braking is most critical and in those scenarios, worst-case braking behavior of a front vehicle is expected.
- *Target Brake* is a discretionary braking maneuver for a target like a red light or traffic jam queue.
- *Follow Vehicle* does not represent a braking maneuver per se, but represents typical braking strengths, when a vehicle is following another in normal traffic.
- *Follow Road* is the default behavior without any particular actor or initiator scenario in the vicinity and therefore low deceleration is expected.

In the example, the behavior intent classification in *FV Braking Behavior* is performed primarily based on the time to collision (*Measured TTC)* between the front vehicle (FV) and the vehicle, which is followed by the FV. In the case of absence of a "front-front vehicle" (FFV), *Target Brake* and *Follow Road* likelihood are determined by map information. Now, even if the TTC between FV and FFV objectively suggests particular likelihoods for the upcoming behavior, the actual behaviors of real human drivers still vary due to different driving styles and capabilities. Therefore, these variations are explicitly modeled through the node chain *FV View Range* → *FV Situation Perception* → *FV Situation Assessment* → *FV Maneuver Decision*. It represents a runtime version of the set of agent capabilities (SEM concept in Section 4.1) and captures the likelihood of critical capability deviation (CBCT concept in Section 4.2). For instance, even though a situation is assessed correctly by the FV (output of *FV Situation Assessment*), the probability of still taking the wrong maneuver decision can be modeled in the BN (*FV Maneuver Decision* "Incorrect"*)*. In addition to BNs enabling these capability variations to be modeled explicitly and probabilistically, situation influences can be modeled to condition the capabilities. This is done for the *FV View Range* capability, which is exemplarily conditioned on *Heavy Rain*: If heavy rain is present, we assume the FV cannot properly detect the FFV and therefore the probability of taking the wrong maneuver decision in consequence is increased.

In real-world applications, perfect sensing and perfect knowledge cannot be assumed. Thus, aleatoric uncertainties related to imperfect sensing and epistemic uncertainties related to imperfect knowledge need to be handled.

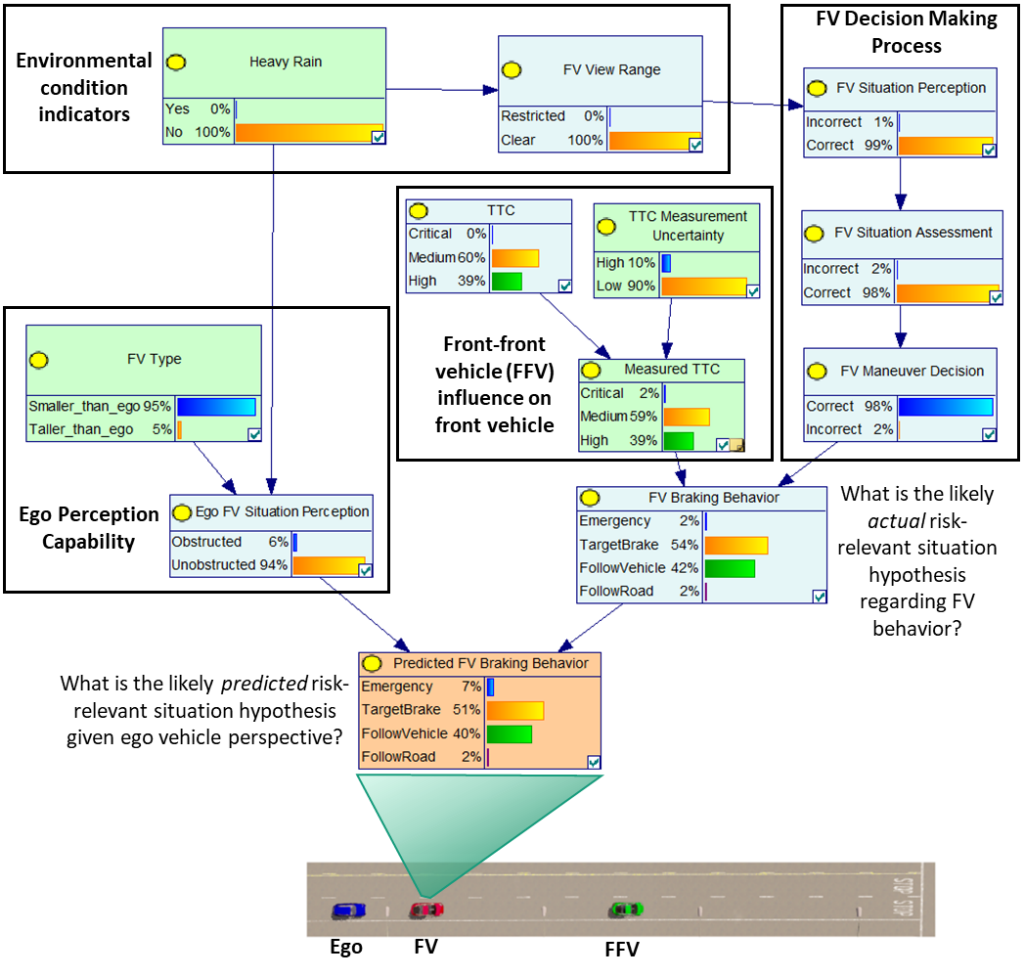


Figure 5.16 Bayesian network model for front vehicle critical braking behavior intent prediction. The model was created with the GeNIe modeling software[12]. Adapted from the figure originally published in the SINADRA software repository. © 2021 Jan Reich.

The node set *TTC, TTC Measurement Uncertainty, Measured TTC* represents a pattern on how to include aleatoric uncertainties. The true *TTC* is noised with a *TTC Measurement Uncertainty* probability distribution that can be derived from knowledge about sensor uncertainties or even dynamically determined based on environment factors.

To address imperfect ego knowledge and the resulting epistemic uncertainties, the situation-dependent ego perception limits are explicitly modeled. For instance, if the front vehicle type (*FV Type*) is a truck and

[12] GeNIe Bayesian network modeling and inference tool from BayesFusion: https://www.bayesfusion.com/genie/

therefore larger than the ego, the ego vehicle cannot perceive potential front-front vehicles due to them being occluded. Likewise, *Heavy Rain* affects the *Ego FV Situation Perception*, too. Such epistemic uncertainties related to ego perception capabilities can be modeled in BNs, too. This inclusion of ego perception limits is also the reason why the output of *Predicted FV Braking Behavior* (orange node) is used as inference result and not *FV Braking Behavior* alone.

Overall, the discussion of the example front vehicle braking BN demonstrates that using BNs as runtime models in the SINADRA context makes it possible to express probabilistic variations related to 1) the agent capability chain, 2) imperfect sensing due to measurement limits and 3) imperfect knowledge due to ego observability limits. This turns BNs into the formalism of choice to use in real-world applications.

To transform the design-time BNs modeled in the .xdsl format in GeNIe into executable software components performing the behavior intent inference fed by data to determine the evidence nodes, the process in Figure 5.17 was used. The BN file generation automatically produces the Python code with placeholder stubs, where data inputs from the situation class detection or CARLA scene inputs are manually mapped to the BN node states. The concrete intermediate formats are shown in Figure C.5 in Appendix C.2.

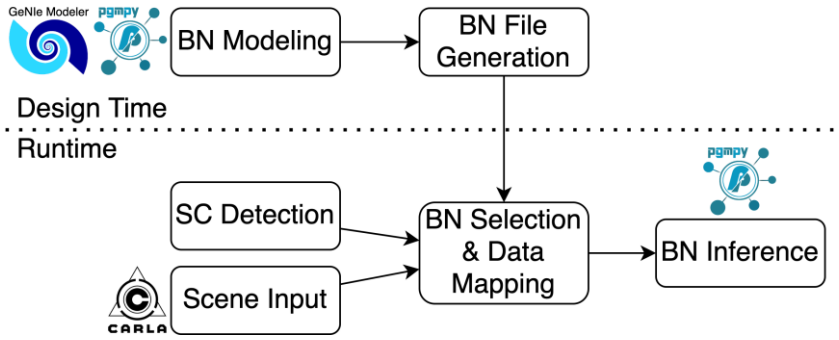


Figure 5.17 Process of semi-automatically transforming a design-time Bayesian network model in the GeNIe modeling tool into an executable Python inference component based on the pgmpy library.

5.2.4 Motion models and trajectory distributions

Intent likelihoods are not directly usable by trajectory-based risk computation methods like (Eggert, 2018). Therefore, this case study used an explicit intent-to-trajectory distribution transformation step that maps each behavior intent into a probabilistic prediction of future positions of the actor. Figure 5.18 illustrates the transformation from intent to trajectory distribution, exemplified for the emergency brake behavior.

The implementation includes a set of longitudinal and lateral motion models that enable trajectory generation for different intents. Longitudinal behaviors include *Emergency* and *Target Brake* behaviors, which were realized based on (Schreier et al., 2016), as well as *Follow Road* and *Follow Vehicle* behaviors, whose trajectory distributions were generated with a probabilistic variant of the Intelligent Driver Model (IDM) based on (Treiber et al., 2000). Trajectory distributions for lane change behaviors are generated with a parametric cubic Bézier curve approach. For each behavior, the component generates a distribution of possible future positions over a prediction horizon by sampling probabilistic parameters and evaluating the respective motion model.

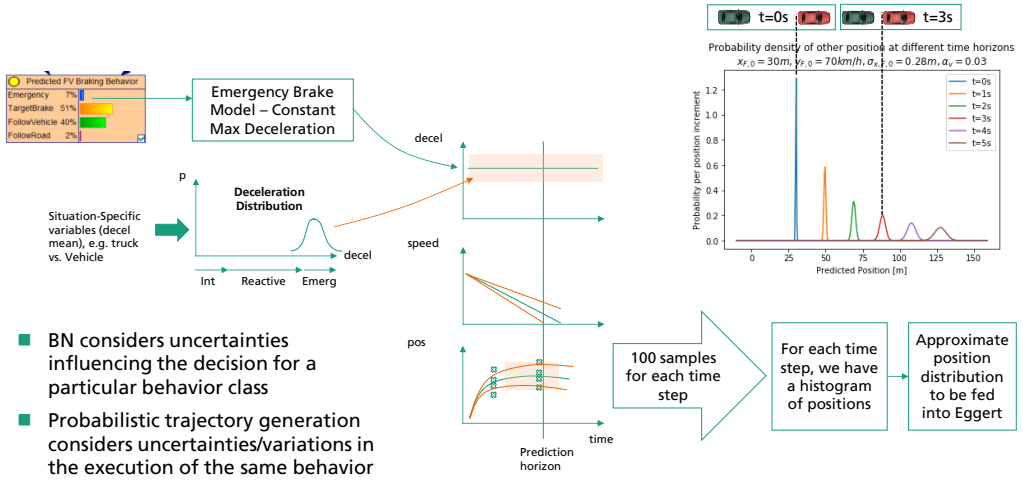


Figure 5.18 Transformation from behavior intent to trajectory distribution. Behavior-specific motion models generate distributions rather than single trajectories by sampling probabilistic parameters (e.g., deceleration limits or time gaps) to represent variability in intent execution. Originally published in the SINADRA software repository. © 2021 Jan Reich.

5.2.5 Risk computation and intent-weighted aggregation

Given predicted trajectory distributions for the ego vehicle and other relevant actors, the component computes collision risk using the probabilistic integral collision risk metric published in (Eggert, 2018). The output of this metric is the collision probability over time. The risk metric expects trajectory distributions from ego and an actor like the front vehicle as input. The risk metric is applied for each ego-actor-behavior intent triple leading to different risk values per actor behavior intent (Figure 5.19).

Finally, these intent-specific risk values are aggregated into a total risk score based on weighting the risk values by the behavior likelihoods inferred by the SINADRA BN. In consequence, behavior intents that lead to high risk like *Emergency braking* but have low likelihood contribute less to the total risk score than they would under a worst-case intent assumption (=expecting

emergency braking with 100% likelihood). In the following demonstration in Section 5.2.6, this worst-case baseline is instantiated by setting the critical intent likelihood to 1.0 and comparing the resulting aggregated risk curve against the intent-weighted aggregation produced by the BN.

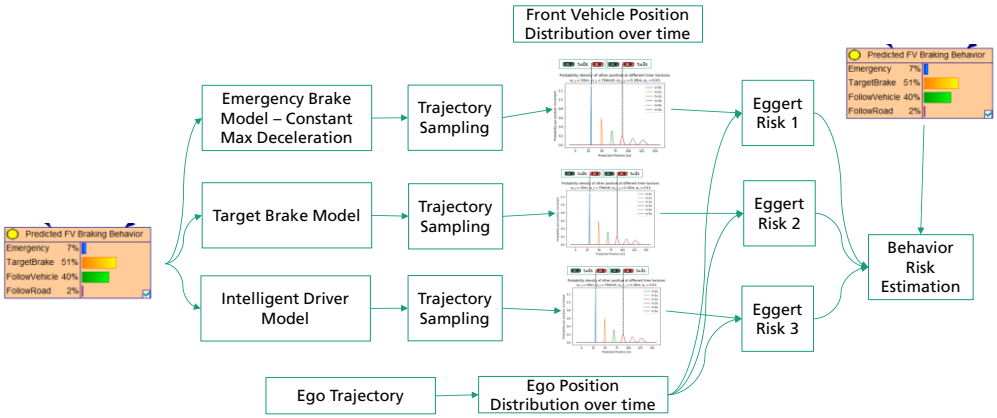


Figure 5.19 Behavior-specific risk computation based on inferred behavior intents from the SINADRA BN and generated trajectory distributions for each intent. The final situation-aware dynamic risk estimate is produced by weighting the behavior-specific risk scores with the estimated behavior likelihoods.

This aggregation mechanism is the core link between SINADRA-based intent inference and the final situation-aware dynamic risk output: Compared to worst-case assumptions, it makes the risk output sensitive to the risk-relevant drivers of the situation.

5.2.6 Runtime feasibility demonstration

This section demonstrates the runtime feasibility of the SINADRA component described in the previous subsections in a front vehicle following scenario in CARLA, as previously published in (Reich et al., 2021). The blue ego vehicle is following the red front vehicle (FV), which approaches another green front vehicle that has a lower traveling speed than ego and FV. The goal is to show that the integral collision risk between the ego and the red front vehicle is lower in a situation with non-worst-case behavior compared to using worst-case behavior assumptions.

Figure 5.20 shows the BN used in the demonstration. This is a reduced variant of the BN shown in Figure 5.16. The nodes in Figure 5.16 related to illustrating the measurement uncertainty pattern of TTC as well as the occlusion aspect are removed for clarity. The conditional probability tables of the BN are chosen to demonstrate the concept and are not verified. The supervised master's thesis in (Wellstein, 2021) compared approaches for learning

SINADRA BN parameters from real data and how they can be plausibilized by means of strength-of-influence and sensitivity analyses.

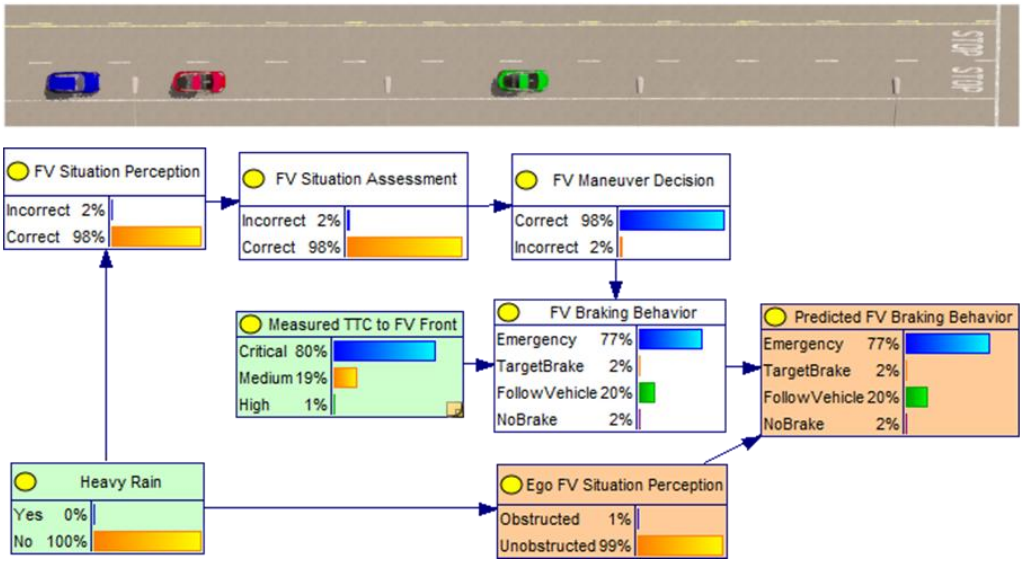


Figure 5.20 Example Bayesian network for SINADRA-based critical braking behavior intent prediction for the red front vehicle considering the slower green FV front vehicle. Adapted from Fig. 2 in (Reich et al., 2021), with permission from Springer Nature. © 2021 Springer Nature Switzerland.

In the beginning phase of the demonstration scenario, the time-to-collision between red and green vehicles is assumed to be medium, because the distance between both vehicles is still uncritical. As the scenario progresses to a later phase, the TTC decreases until a point where it reaches a critical value, so an emergency braking maneuver of the FV is likely.

The intermediate outcomes of the SINADRA runtime pipeline are shown in Figure 5.21 for the uncritical phase and in Figure 5.22 for the critical phase. Together, the figures show the full chain in one view: BN intent likelihoods drive different intent-conditioned trajectory distributions and behavior-specific risk curves, which then combine into an aggregated risk output that differs from the worst-case intent baseline when the situation cues indicate a non-critical situation.

The BN inference results in different relative braking behavior intent likelihoods for emergency braking and following the green vehicle with moderate braking expectations.

Subplots a1 and b1 show the predicted trajectory distributions for the possible behavior intents over time. Note that the Intelligent Driver Model (IDM) combines the *Follow vehicle* and *No Brake* (i.e., follow road) behavior intents.

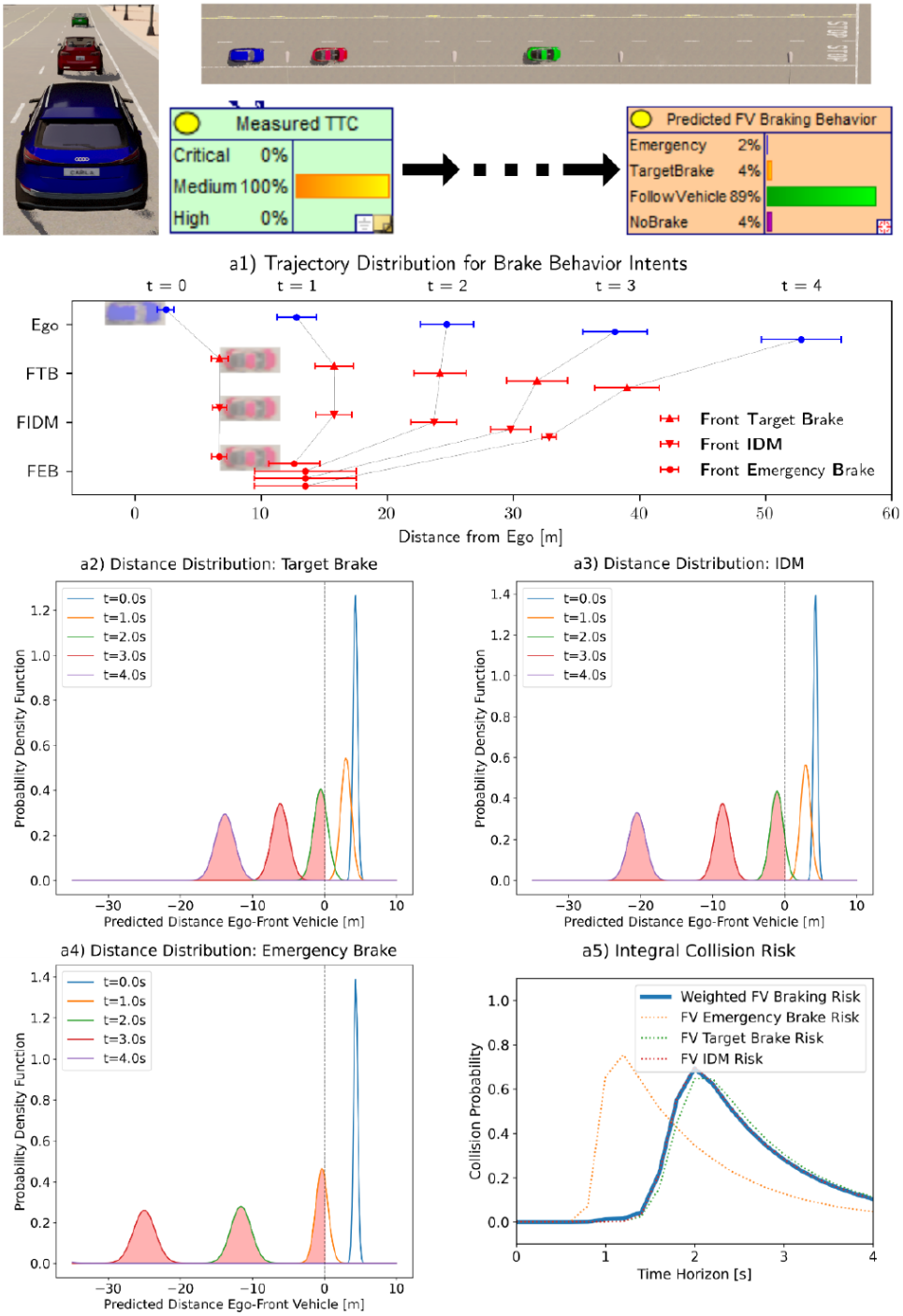


Figure 5.21 Situation-aware front vehicle braking dynamic risk assessment in uncritical scenario phase. Reproduced from Fig. 3 in (Reich et al., 2021), with permission from Springer Nature. © 2021 Springer Nature Switzerland.

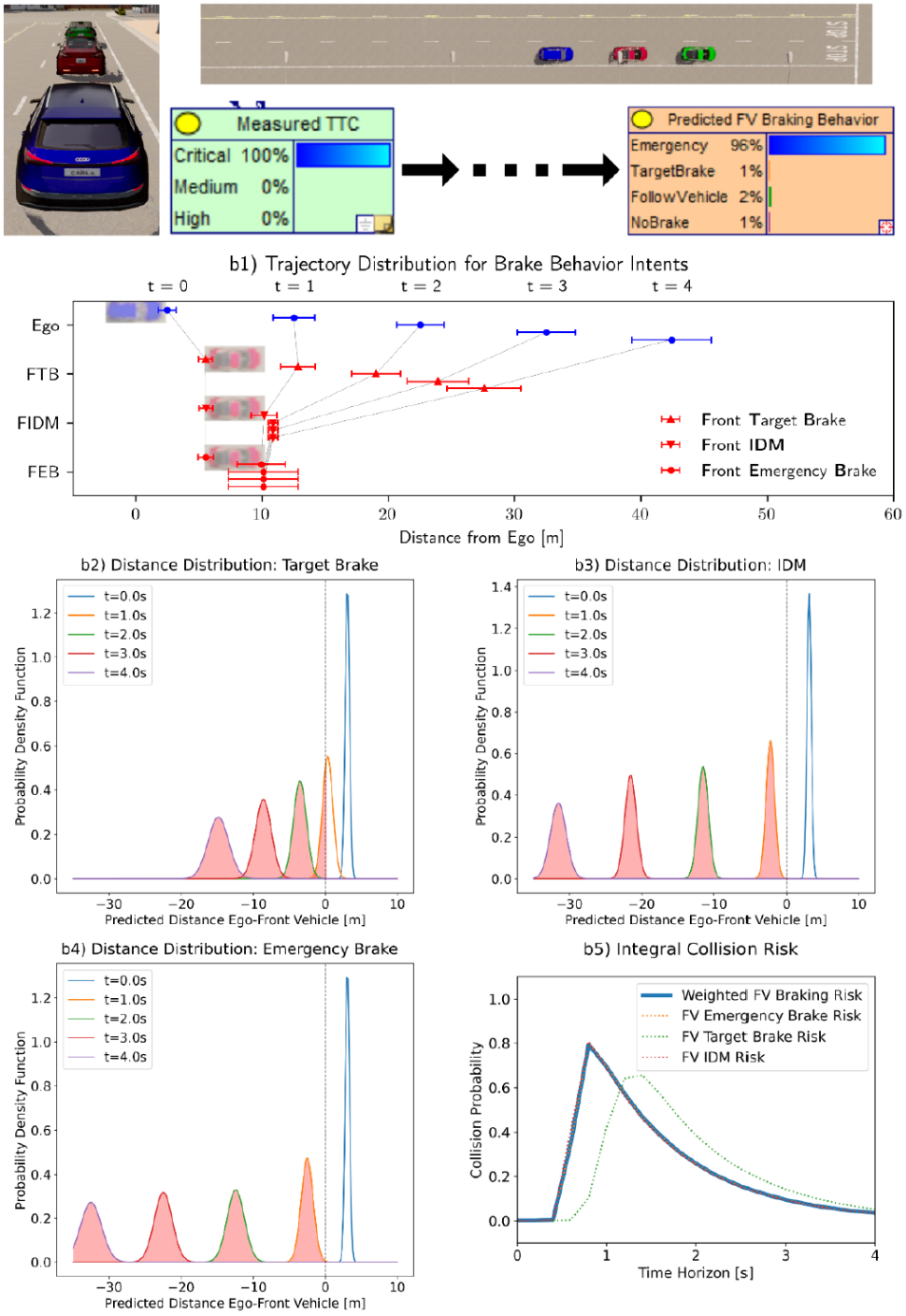


Figure 5.22 Situation-aware front vehicle braking dynamic risk assessment in critical scenario phase. Reproduced from Fig. 3 in (Reich et al., 2021), with permission from Springer Nature. © 2021 Springer Nature Switzerland.

The ego trajectory is constructed by assuming a constant speed in this example, while in reality the trajectory would be provided by the ego trajectory planner.

Subplots a2)-a4) and b2)-b4) show the distance distribution after combining the ego trajectory distribution with each actor's trajectory distribution. The red area below the probability density functions at different times indicates the cumulative probability of both vehicles being at the same position at the same time (i.e., crashed), which is the distance-based risk criterion used in (Eggert, 2018). It is interesting to note that in a4), a critical situation is predicted to happen at around 1.0 second due to assumed emergency braking, while in a2), the critical situation happens later only at around 2.0 seconds.

The final integral collision risk assessment is shown in plots a5) and b5), where the separate integral collision risks of front vehicle behavior intents (dashed) are weighted with the behavior likelihoods inferred from the BN into the thick aggregated risk graph. The resulting aggregated risk differences illustrate the value of SINADRA: While in the critical situation (b5), the weighted risk is close to all individual behavior risks, in the uncritical situation (a5), the weighted risk does not favor the high-risk score from the emergency braking behavior, as this behavior intent is very unlikely based on the current environmental situation.

The SINADRA "risk sensor" component implementation delivers this risk estimate continuously and thus can be the input signal for adapting the ego vehicle behavior to a more permissive mode in detected non-worst-case risk situations. Demonstrating such behavior changes is out of scope in this case study. However, the behavior planning components of (Eggert, 2018) consume arbitrary integral collision risk estimates and thus can be in principle parameterized with the output of the developed SINADRA component. This also illustrates how motion-based assumption adaptation approaches (Section 2.4) can benefit from SINADRA: the motion-based risk metric itself is unchanged, but its prediction input is made situation-aware by an explicit risk-driven decomposition of critical behavior intents enabled by the SINADRA engineering framework.

Technically, the runtime feasibility is additionally supported by the execution performance of the component in this setup: SINADRA computes a risk score every 75 milliseconds on average, on a typical Intel Core i5-7600K CPU, inferring braking and cut-in behavior BNs in parallel and generating trajectory distributions with 20 samples for each of the four different behavior intents. The component predicts the risk for two actors over a 4.0-second time horizon with a 0.2-second time-step resolution. This supports the feasibility of running intent inference, trajectory sampling, and risk aggregation online within a simulation step loop.

5.2.7 Evidence assessment and limitations

This case study provides evidence primarily for EP2 (runtime feasibility) and supporting evidence for EP1 (method feasibility) at component level. It targets collision risk in urban lane-following situations in which the ego vehicle's risk depends strongly on the behavior intents of nearby human-driven vehicles. The case study demonstrates a runtime chain that infers intent likelihoods from situational cues by means of Bayesian networks, translates the intents into trajectory distributions using behavior-specific motion models, and computes an intent-weighted collision risk output signal in a CARLA-based demonstrator.

Evidence against EP1. EP1a (cross-domain applicability) is addressed by transferring the SINADRA idea of external-agent behavior as a risk variability source from warehouse robotics to public-road traffic: instead of human workers, the relevant uncertainty source is human driver intent of other vehicles. EP1b (design-time artifact chain feasibility) is supported only in a reduced form. The case study uses qualitative cause–effect intent models as a design-time basis for selecting relevant intent classes and cues for runtime Bayesian network inference, but it does not execute the full SEM→CBCT modeling chain as primary work products. EP1c (design-time justification support) is treated at the level of documented modeling rationale and explicit assumptions for the runtime intent models, including observability limits and measurement uncertainty. This supports reviewability of the runtime modeling choices, but does not introduce an empirical validation claim for intent prediction accuracy.

Evidence against EP2. EP2a (runtime model feasibility) is addressed by operationalizing intent-focused risk implausibility hypotheses as executable Bayesian network inference components, including an explicit distinction between unobservable "true" intent and the ego-facing predicted output, and by incorporating measurement uncertainty for situation variables. A semi-automated transformation workflow from design-time Bayesian network models to runnable inference code supports feasibility beyond a single hand-coded model instance. EP2b (runtime integration feasibility) is supported by technical integration into a CARLA runtime pipeline that performs situation class localization, actor role selection, BN intent inference, intent-conditioned trajectory distribution generation, and probabilistic collision risk computation with subsequent intent-weighted aggregation. EP2c (observable runtime effects) is supported regarding the risk output: the demonstration differentiates between an uncritical and a critical phase and shows that, in the uncritical phase, the aggregated risk does not inherit the high risk of a worst-case emergency-braking assumption when that intent is inferred as unlikely. The produced risk signal is positioned as an input for later constraint relaxation or planning adaptation, while closed-loop behavior change is out of scope here.

Limitations and threats to validity. The internal validity is limited by demonstrator scope and model verification. The shown scenarios cover a restricted lane-following setting with a limited set of modeled intents and cues. A key limitation is that the Bayesian network structure and parameterization are used in a demonstrator sense and are not empirically validated against real-world driving data within this thesis. Therefore, the evidence supports runtime feasibility and the qualitative effect of intent-weighted risk aggregation in the shown scenarios, but it does not support claims about calibrated intent likelihoods for general road traffic. Trajectory prediction and uncertainty handling rely on stated simplifying assumptions, and the evaluation focuses on risk output differentiation rather than closed-loop system behavior change or system-level safety assurance. The external validity is limited by CARLA-specific sensing and scenario assumptions and by engineered situation class and actor role logic for a restricted ODD subset. These limitations bound the claims to component-level runtime feasibility under the stated assumptions, but do not undermine the main EP2 evidence provided by the executed runtime chain and the intent-weighted risk output behavior.

5.3 Chapter Summary

This chapter evaluated SINADRA against two perspectives: EP1 (methodological feasibility) and EP2 (runtime feasibility). Across the two case studies, the results jointly support the core claim of the dissertation: SINADRA can derive monitorable low-risk situation hypotheses from hazardous situations in a traceable way, and run these hypotheses at runtime to support more permissive operation than static worst-case assumptions, while keeping the safety reasoning grounded in design-time artifacts.

EP1 – Methodological feasibility

EP1a (cross-domain applicability) is addressed by applying the SINADRA idea of risk variability caused by external-agent behavior in two different operational design domains with different external agents: human workers in warehouse corridors and human drivers in urban driving. **EP1b (design-time artifact chain feasibility)** is demonstrated most strongly in the warehouse case study, where the full design-time chain is executed: hazardous situations are scoped via SEM, refined into behavior risk scenarios via CBCT, and turned into explicit risk implausibility hypotheses (RIH). **EP1c (design-time justification support)** is supported by verification evidence from the CBCT analysis: within the modeled scope, the minimal cut set analysis shows that the selected RIH together with additional safety measures blocks the relevant behavior risk scenarios for the passing and overtaking scenarios. This provides a concrete, reviewable argument why safety constraint relaxation based on

RIH is acceptable in the context of the system safety concept under the stated hypothesis conditions.

EP2 – Runtime feasibility

EP2a (runtime model feasibility) is shown in both case studies by transforming selected hypotheses into executable runtime inference components: Boolean runtime models in the warehouse demonstrator, and Bayesian-network-based intent inference (including uncertainty handling and a semi-automated model-to-code workflow) in the ADS demonstrator. **EP2b (runtime integration feasibility)** is supported by successful integration into representative runtime stacks: a ROS-based safety monitoring architecture for the warehouse system, and a CARLA-coupled pipeline for situation class detection, actor role selection, intent inference, trajectory distribution generation, and collision risk computation with intent-weighted aggregation. **EP2c (observable runtime effects)** is supported by observable, runtime-varying outputs: in the warehouse study, hypothesis validity drives discrete constraint mode switching with estimated corridor time gains. In the driving case study, the aggregated risk signal differs from a worst-case intent baseline and changes between uncritical and critical phases, demonstrating suitability as an input to increase the permissiveness of motion-based risk metrics.

6 Discussion and Conclusion

6.1 Key Findings

This dissertation addressed the question of how hazard analysis and risk assessment artifacts can be extended so that engineers can identify monitorable low-risk situations that justify adaptive, permissiveness-increasing constraint relaxation in runtime safety monitors for autonomous systems. The central result is that SINADRA provides an engineering method framework to derive and justify risk implausibility hypotheses from design-time artifacts, to transform these risk implausibility hypotheses into executable runtime models, and to use their inferred outputs as inputs for the situation-dependent adaptation of safety constraints at runtime.

The evaluation provides complementary evidence across two application domains. In the warehouse case study, the full design-time chain from structured environment modeling to Critical Behavior Cause Trees is executed, and the resulting risk implausibility hypotheses are linked to a reviewable justification argument within the modeled scope. In the ADS/CARLA case study, SINADRA is demonstrated as an executable runtime chain that performs critical behavior intent inference under uncertainty and integrates this inference into probabilistic collision risk computation. The runtime outputs of the chain show observable runtime differences relative to a worst-case baseline, suggesting that low-risk situations can be detected.

6.2 Threats to Validity

Construct validity. A primary threat to construct validity is a mismatch between the terms “permissiveness” and “safety” and what is actually evidenced. In this thesis, permissiveness is operationalized through explicit differences in the selection of safety constraint assumptions and through observable differences in the runtime risk output when situation-dependent information is incorporated. Safety is not claimed as a global guarantee of safe operation. Instead, the strongest safety-oriented evidence is the design-time argument that connects hazardous situations, the structured environment model, risk implausibility hypotheses and safety concept measures in Critical Behavior Cause Trees and that supports verification checks within this modeled scope. This threat remains relevant because the question “how safe is safe enough?” is not ultimately answered, and no concrete criteria are defined that would make the answer testable through concrete criteria on the safety engineering artifact level, e.g., regarding a “sufficient” coverage of environment factors in SEM and CBCT. The

achievable permissiveness depends on a testable definition of safety, therefore the latter needs to be clarified first.

Internal validity. The internal validity is threatened by the fact that demonstrator results are influenced by scenario selection, modeling choices, and parameter settings. For the warehouse case study, the main internal threat is that omitted causal paths could introduce additional minimal cut sets and thereby change the set of risk implausibility hypotheses that would be required for a comparable sufficiency argument. For the second case study, the key internal threat is that the Bayesian network design and parameterization serve a feasibility purpose in the demonstrator context and are not empirically validated against real-world driving data within this dissertation. As a consequence, the evidence supports runtime feasibility and qualitative scenario-dependent effects in the demonstrated setting, but it does not support claims about calibrated intent likelihoods or general performance across broad operational conditions.

External validity. External validity is limited by the scope of the operational design domains, the available sensing assumptions, and the selection of operational situations and actor roles that are instantiated in the case studies. The results support that SINADRA transfers as a method framework because it can be instantiated in different domains. However, transfer to a new system or ODD requires renewed structured environment modeling and renewed Critical Behavior Cause Tree modeling, and it requires renewed checks that the validity conditions of the risk implausibility hypotheses remain observable and usable under the target system architecture and sensing stack.

Reliability. A key reliability threat is the dependence on expert judgment in structured environment modeling and Critical Behavior Cause Tree modeling. Different analysts may select different causal factors, different decomposition depth, and different stopping criteria. The reliability therefore depends on the documentation of assumptions and constraints, explicit modeling guidelines and systematic reviews. Tool support and deterministic transformation steps can reduce variability for parts of the chain, but they do not remove the need for expert review of the model content and its scope. This holds especially true in the current situation, where the industry still lacks concrete and testable sufficiency and termination criteria for safety-oriented environment decomposition activities.

6.3 Methodological Limitations

The application of the SINADRA approach is most suitable when:

(1) risk is strongly influenced by situation-dependent factors,
(2) worst-case assumptions lead to frequent unnecessary permissiveness constraints,

(3) there exist observable cues that can support a defensible split into “worst-case remains necessary” versus “risk implausibility hypotheses are monitorable”.

If these preconditions are not met, SINADRA still supports conservative operation, because the worst-case remains the default, but the benefit in permissiveness can become marginal. This is especially true because engineering and verifying risk implausibility hypotheses and their transformation into runtime components introduce both additional complexity into the system and additional costs for engineering and assurance. These costs are only justified if the gains through added permissiveness outweigh the costs.

A first methodological limitation is **observability feasibility**. Risk implausibility hypotheses can only be operationalized if their validity conditions can be mapped to measurable features with uncertainty that can be handled safely. For many features related to basic object presence (e.g., “there is an object with size x in the vicinity”), existing perception stacks are reliable enough for safety-critical applications. However, many situation influence features relevant to monitor risk implausibility hypotheses refer to visual properties (e.g. carrying a box or eye gaze direction in the warehouse study) that require object classification. Classic algorithmic visual perception stacks have frequent problems with robust visual classification tasks of complex features, hence the connection to task databases and head-mounted smart helmets as proxies for such signals in the warehouse study. Machine learning-based (ML) perception can, in principle, increase the observability of such features. At the same time, this creates a conflict for safety assurance. If the validity of a risk implausibility hypothesis depends on ML-derived features, then the correctness and robustness of the perception component becomes part of the safety argument. In many application contexts, this justification is still limited because ML components often provide only weak safety guarantees or are even fully prohibited in safety-critical applications. This conflict does not invalidate the method, but it constrains the use cases where SINADRA can rely on learned perception today. In addition, it motivates future work on assured perception outputs, calibrated uncertainty estimates, and safety cases that explicitly address ML-based perception so that SINADRA runtime monitors can leverage their potential.

A second limitation is **scope control and completeness justification** during structured environment modeling and Critical Behavior Cause Tree modeling. The method supports systematic scoping and refinement, but it does not eliminate the fundamental challenge that additional operational situations and additional causal paths can always be added. Sufficiency arguments derived from Critical Behavior Cause Trees are therefore always bounded to the modeled subset. If relevant causal paths are omitted or forgotten, the modeled minimal cut sets can be incomplete, and additional risk

implausibility hypotheses or additional safety measures may be required. This limitation does not invalidate the method, but it constrains how results must be interpreted. SINADRA yields bounded, reviewable arguments under explicit assumptions rather than a claim of global completeness. Thus, the introduced SINADRA approach should be always complemented by adequate operational risk management systems making sure that the assumptions under which SINADRA models were created are valid and stay valid during operation. The model incompleteness limitation cannot be addressed by improving the SINADRA method itself, because its success is dependent on the *ontological knowledge* based on which environment factors are analyzed regarding their relevance for hazardous situations for a given operational domain. This knowledge base is an input to the SINADRA framework, e.g., in the form of environment ontologies. Thus, model coverage should be addressed by improving the knowledge base engineers use as input for applying SINADRA analyses. This requires first developing environment ontologies that better reflect the risk-relevant aspects of specific operational domains. Ongoing work aims to achieve this goal (Hillen & Reich, 2023; Hillen et al., 2025). Second, ontological models always have gaps due to their very nature of being models involving some degree of abstraction from the real world. Therefore, in addition to building more effective ontologies, approaches are required that explicitly acknowledge gaps in ontologies, e.g., due to unknown unknowns like edge and corner cases, but minimize the time from the observation of such cases to their addition to the knowledge base and the analysis of their impact on system safety.

A third limitation concerns the **assurance of probabilistic runtime models**. The ADS/CARLA case study demonstrates a feasible runtime chain with Bayesian network inference and intent-weighted risk aggregation, but it does not validate Bayesian network structure and parameters against real-world driving data within this thesis. This limits claims to runtime feasibility and qualitative effects in the shown scenarios. Methodologically, this means that SINADRA can specify *what must be inferred* and *how Bayesian inference is used in risk computation*, but further work is required to justify that the Bayesian inference outputs satisfy the safety requirements posed by the application. Although this passes the boundary of what this thesis can claim, we still believe that using Bayesian networks delivers the best compromise between expressiveness and verification/validation possibilities. They can be parameterized based on data through machine learning, but unlike other ML approaches like deep neural networks, the semantics of the BN model structure is reviewable by experts. This clear split between the causal model structure and the quantitative conditional probability parameters is what makes BNs appealing for potential use in safety-critical contexts.

Finally, a fourth methodological limitation concerns **scalability in complex environments**. SINADRA is deliberately structured to make assumptions and causal paths explicit, but complex operational design domains can still lead to

a rapid growth of operational situations, actor interactions, and candidate causal factors that need to be represented in structured environment models and Critical Behavior Cause Trees. In such settings, the effort required to build, maintain, and review these artifacts can become the dominant cost driver, and the risk of omissions increases with model size and heterogeneity. This limitation is not specific to SINADRA, but it is particularly relevant because the method's benefits depend on the completeness and clarity of the stated risk implausibility hypotheses and their validity conditions within the chosen scope. To address the limitation, automated tool support can assist in identifying candidate operational situations from logs or scenario descriptions, proposing initial causal factor sets or draft CBCT fragments from prior models. Where the environment and interaction structure is highly repetitive, reuse and pattern-based modeling become essential to avoid re-modeling similar situations from scratch. Such automation does not remove the need for expert judgment, but it can shift human effort from manual enumeration toward review, scoping decisions, and justification quality, which is the part of the process that most directly affects the validity of the resulting risk implausibility hypotheses and their runtime use.

6.4 Contributions

This dissertation provides the following contributions, mapped to where they are delivered in this thesis:

Contribution C1: A cross-domain engineering method (SINADRA) to derive and justify monitorable low-risk situation hypotheses from HARA artifacts.

C1.1 Structured Environment Modeling (SEM): method and artifact structure to scope hazardous situations into explicit, risk-driven environment models and initiator scenarios (Section 4.1, demonstrated in Section 5.1).

C1.2 Critical Behavior Cause Trees (CBCT): a modeling method with formal syntax/semantics that integrates behavior risk scenarios, risk implausibility hypotheses (RIH), and safety measures into one Boolean structure, enabling minimal cut set analysis as reviewable verification evidence for RIH justification (Section 4.2, demonstrated in Section 5.1).

C1.3 Runtime model transformation: a systematic path from risk implausibility hypotheses to executable runtime inference components using Boolean models and Bayesian networks (Section 4.3, demonstrated in Sections 5.1 and 5.2).

C2 Supporting prototype toolchain: a prototype toolchain as an implementation artifact that partially operationalizes SINADRA's method flow, including model handling and generation/configuration of executable Boolean and Bayesian runtime components based on the DDI/ODE workflow.

The toolchain is presented as feasibility support for the method rather than as an independent research contribution (Section 4.4).

C3: Two application studies that provide complementary evidence for methodological feasibility and runtime feasibility.

C3.1 Case Study 1 (warehouse robotics): end-to-end execution of the design-time chain and a demonstrator showing risk implausibility hypothesis-based mode switching with conservative default behavior plus an analytic permissiveness estimate under stated assumptions (Section 5.1).

C3.2 Case Study 2 (urban driving in CARLA): a component-level runtime chain for risk implausibility hypothesis inference with Bayesian networks and probabilistic collision risk computation, with situation-dependent risk output differentiation demonstrated in a selected scenario (Section 5.2).

6.5 Impact

The motivation of this dissertation is shaped by converging expectations across autonomy-relevant domains that safety must be demonstrated under changing conditions and that permissiveness losses resulting from too conservative safety monitoring approaches are unacceptable from business and societal acceptance points of view. In this context, the main impact of SINADRA is that it provides a reusable engineering method that makes risk variability an explicit design subject, ties it to established safety engineering artifacts, and yields executable runtime logic that can keep conservative operation as the default while permitting bounded relaxation under explicit, monitorable conditions.

SINADRA's **scientific impact** is that it closes a central gap identified in Section 2.6: although risk-based assumption adaptation aligns with safety engineering in principle, the literature has lacked a systematic method that makes permissiveness improvements at runtime defensible in terms of hazards and risk. This is significant for both safety and autonomous systems research communities because it provides a concrete proposal for what "justified permissiveness" can mean in technical terms in the sense of a defensible safety argument within a clearly stated scope. It supports more direct and fair comparison of adaptation approaches. It does so by shifting attention away from black-box, scenario-based validation performance results and toward the harder questions of safety justification. These questions include whether the chosen runtime conditions are meaningfully risk-relevant. They also include whether the considered set of low-risk situations is defensibly scoped within the operational design domain. Finally, they include whether the runtime decision to relax assumptions can still be traced back to hazard analysis and risk assessment artifacts, rather than being

driven only by correlation in data-based approaches or by prediction accuracy in motion-based approaches.

For industrial application, SINADRA's value lies in offering a structured method to create reviewable artifacts and bounded arguments that support safe monitor parameter adaptation in the context of established system safety engineering practice. The experiences gained in the SINADRA case studies position SINADRA as a practical method candidate for teams that need to argue when and why a conservative safety monitor may operate less restrictively, while keeping a conservative fallback whenever evidence supporting risk implausibility is missing. Beyond the engineering level, the evolution of industry standardization may profit from the concepts developed in this thesis, especially the future evolution of the standard series IEEE 2846-2022 and IEEE 3321-2024 dealing with assumptions on reasonably foreseeable behavior of other road users in safety-related models.

6.6 Future Research Directions

A first direction is **data-based quantification and assurance of Bayesian network runtime models** for intent inference and similar uncertainty-aware inference tasks. This includes collecting representative datasets for the features used in risk implausibility hypothesis validity evaluation, validating model structure and parameters, and defining acceptance criteria that are aligned with the intended safety use. It also includes explicit treatment of confidence and uncertainty so that conservative fallback behavior is triggered in a controlled and explainable way.

A second direction is **tool-supported assistance for structured environment modeling and Critical Behavior Cause Tree modeling**, including partial automation support. This includes support for literature extraction, proposal of candidate causal factors and initiator scenarios, consistency checking between modeled measures and the system safety concept, and change impact suggestions when the operational design domain or system functions evolve. The goal is to reduce modeling effort and reduce omissions while keeping human review and accountability explicit.

A third direction is to **raise the maturity level toward real deployments**. This requires replacing simulation-only assumptions with real sensing limitations and timing constraints, demonstrating closed-loop behavior adaptation beyond risk-output differentiation, and introducing in-service monitoring and feedback loops so that SEM and CBCT artifacts can be updated when novel conditions and residual risks are observed.

A fourth direction is **applying SINADRA in additional autonomy use cases**, including automated drones, off-road robotics, and driver monitoring in advanced driver control assistance systems (DCAS). This research should

establish tailoring rules for structured environment modeling and Critical Behavior Cause Tree modeling in domains with different hazardous situation types, sensing modalities and operational constraints. In addition, it should characterize which environmental aspects dominate risk variability in these domains and which permissiveness improvements are possible based on stated risk implausibility hypotheses.

6.7 Closing Remarks

Autonomous systems must remain safe under changing conditions, but they must also avoid unnecessary conservatism that prevents them from delivering their promised benefits. This dissertation contributes SINADRA as a traceable engineering method that links hazard analysis and risk assessment artifacts to monitorable risk implausibility hypotheses and executable runtime logic, and it demonstrates that situation-dependent safety constraints can be grounded in explicit assumptions while preserving conservative fallback behavior whenever conditions are not met. While further work is needed to scale the approach to more complex operational design domains and to strengthen assurance for probabilistic inference components, the results indicate a practical path forward. Advancing this path can help bring autonomy to roads and workplaces in a way that remains accountable, earns trust through transparent reasoning, and ultimately reduces harm while improving how people live and work.

Appendix

Appendix A Supplementary Method Material

A.1 Example Initiator Scenarios

In the following, a supplementary list of initiator scenario examples is provided as a result of applying the initiator scenario analysis (ontology items are formatted in *italics*).

- Environment source
  - *Squall* causes braking force
  - *Transition from low μ to high μ* causes braking force
  - Material on road (water → *aquaplaning*, *mud*) causes braking force
- Object source
  - *Fallen tree* on street causes braking force
  - *Fallen load* from truck causes braking force
- Road structure source
  - *Pothole* causes braking force
  - *Moveable bridge* causes braking force when up
- Vehicle source
  - Braking system *mechanical failure* causes unintended braking force
  - *Tire* blowout causes braking force
  - Braking system *software malfunction* causes unintended braking force
  - *ADAS malfunction* causes unintended braking force
  - *Security attack* causes unintended braking force
- Agent source
  - Unintentional: Unintentional brake due to *erroneous foot placement*
  - Example L1 (Road network / traffic guidance) initiator scenarios:
    - Intentional braking to reduce speed for *curve ahead*
    - Adapting to lower *speed limit*
    - Avoiding getting speeding ticket at speeds that are too high in response to *speed limit*
    - Intentional braking for *traffic signal* change
    - Intentional braking to avoid vehicle damage due to detected *pothole*
    - Intentional braking for *end of road*
  - Example L2 (Roadside structures) initiator scenarios:
    - Intentional braking to avoid collision with vehicles emerging from *parking spaces*

        - Intentional braking to be cautious in *school zone* or *kindergarten* areas.
    - Example L3 (Temporary L1 & L2 Modifications) initiator scenarios:
        - Intentional braking to switch lanes due to *end of lane* before *construction zone*
        - Intentional braking to avoid unstable vehicle dynamics at *piles of tree leaves* or *oil on road surface*
        - Road oscillation due to *earthquake*
    - Example L4 (Movable Object) initiator scenarios:
        - Giving way to *emergency vehicles*
        - Reacting to *front vehicle braking* maneuver
        - Reacting to *pedestrian jaywalking*
        - Reacting to side *vehicle cutting* in front of ego vehicle
        - Reacting to *traffic jam end / stop-and-go*
        - *Overtaking vehicle* in other direction
    - Example L5 (Environmental Condition) initiator scenarios:
        - Avoiding unstable vehicle dynamics in undriveable areas → *icy incline*
        - Reacting to lack of situational awareness due to *sun glare* / visibility change
        - Road condition change (braking to control *aquaplaning*)
    - Example L6 (Digital Information) initiator scenarios:
        - Braking due to signalized cooperative *hazard warning*

A.2 SINADRA Capability Deviation Analysis Guidewords

Focus Guidewords

Guideword	Description	Example Scenario
Missed Focus	Failure to pay attention to critical scene elements	Driver doesn't notice a pedestrian due to tunnel vision.
Distracted	Focus diverted by irrelevant stimuli	Driver looks at phone and misses oncoming traffic.
Over-focused	Excessive focus on one scene element, ignoring others	Driver fixates on a billboard, ignoring nearby vehicles.
Impaired Focus	Reduced attention due to fatigue, intoxication, or stress	Driver is drowsy and misses a stop sign.
Fluctuating Focus	Alternating between over-focus and distraction	Driver keeps alternating attention between traffic and GPS.
Delayed Focus	Focus shift occurs too late	Driver reacts too late after noticing brake lights.

Perceive Guidewords

Guideword	Description	Example Scenario
No Perception	Failure to detect relevant scene elements	Driver does not see a cyclist approaching an intersection.
False Perception	Misinterpretation of sensory input	Driver perceives a reflection as an obstacle.
Delayed Perception	Recognition of a hazard occurs too late	Driver notices brake lights too late to stop in time.
Partial Perception	Only some critical elements are detected	Driver sees oncoming traffic but misses a crossing pedestrian.
Over-perception	Detects non-existent hazards, leading to false alarms	Driver thinks a shadow on the road is a hazard.

Understand Challenge Guidewords

Guideword	Description	Example Scenario
Misunderstood Challenge	Incorrectly assessing the level or type of threat	Driver assumes a vehicle will stop but it continues moving.
Underestimated Challenge	Insufficient recognition of hazard severity	Driver underestimates the stopping distance needed on a wet road.
Overestimated Challenge	Overestimating the danger, leading to overreaction	Driver panics and brakes suddenly for a non-threatening shadow.
Unassessed Challenge	Failure to assess the situation entirely	Driver does not consider how fast another vehicle is approaching.
Delayed Assessment	Correct assessment is made but too late	Driver realizes too late that the curve ahead is sharper than expected.

Understand Reaction Capabilities Guidewords

Guideword	Description	Example Scenario
Inaccurate Assessment	Incorrect understanding of one’s own or vehicle limits	Driver overestimates their ability to stop in time.
Ignorance of Limits	Lack of knowledge about vehicle handling in different conditions	Driver doesn't realize that braking distance increases on icy roads.
Physical Impairment	Driver has physical limitations affecting response	Driver’s leg is injured, slowing their ability to press the brake.
Cognitive Overload	Mental overload prevents clear assessment of options	Driver is overwhelmed by multiple hazards at once.
Misjudged Feedback	Incorrect interpretation of system feedback or response time	Driver believes ABS will reduce stopping distance faster than it does.

Decide Guidewords

Guideword	Description	Example Scenario
Indecisive	Hesitation in making a decision	Driver cannot decide whether to brake or swerve, losing time.
Incorrect Decision	Decision made that is suboptimal for the situation	Driver decides to swerve instead of braking, causing a collision.
Impulsive Decision	Rash decision without proper evaluation	Driver brakes abruptly without checking mirrors.
Delayed Decision	Decision is made after the critical time window	Driver decides to brake only after passing the point of safe stopping.

Act Guidewords

Guideword	Description	Example Scenario
No Action	Failure to act despite deciding to act	Driver decides to brake but does not press the pedal.
Incorrect Action	Wrong action is performed	Driver presses the accelerator instead of the brake.
Delayed Action	Action is executed too late	Driver hesitates and brakes too late to avoid a collision.
Overreaction	Excessively strong action leading to secondary issues	Driver slams the brakes, causing skidding on a wet road.
Partial Action	Incomplete execution of the necessary action	Driver partially presses the brake pedal, slowing down insufficiently.

Appendix B Supplementary Toolchain Material

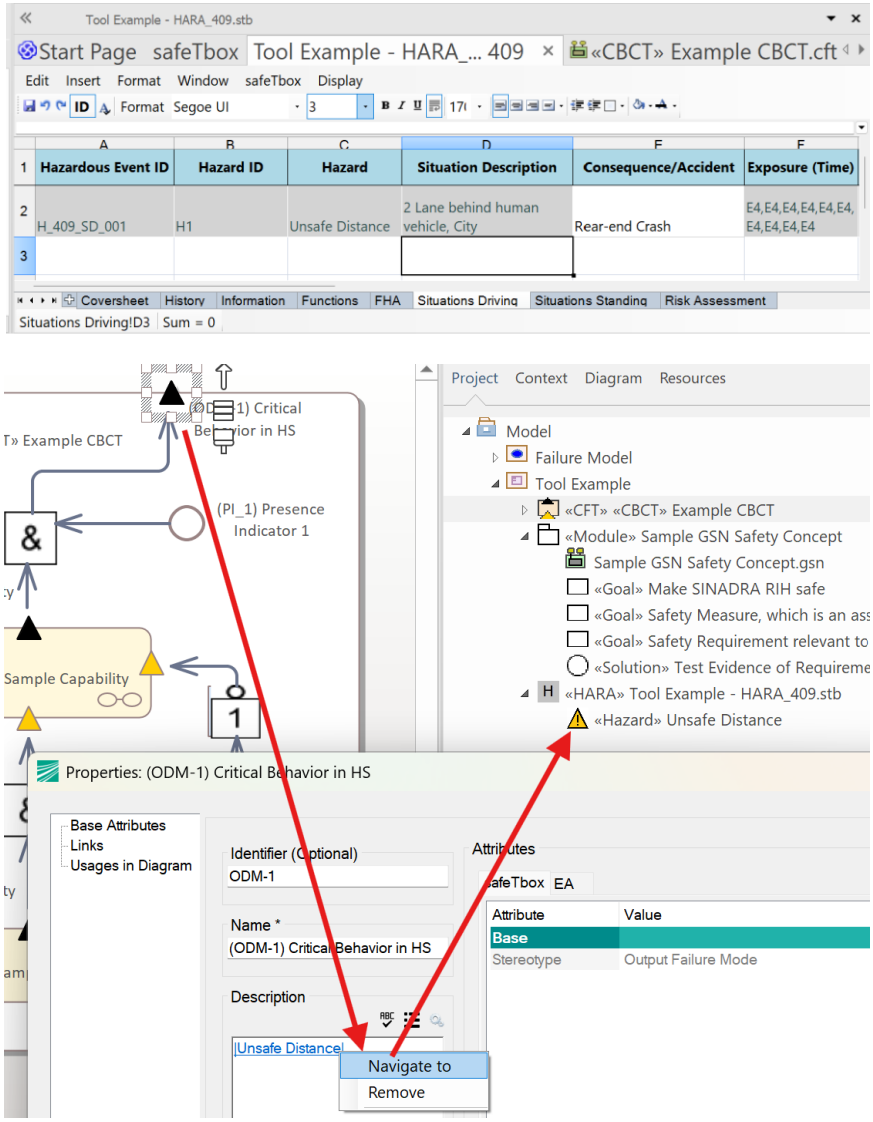


Figure B.1 Illustration of formal traceability between CBCT output deviation modes and hazard elements in the safeTbox HARA model.

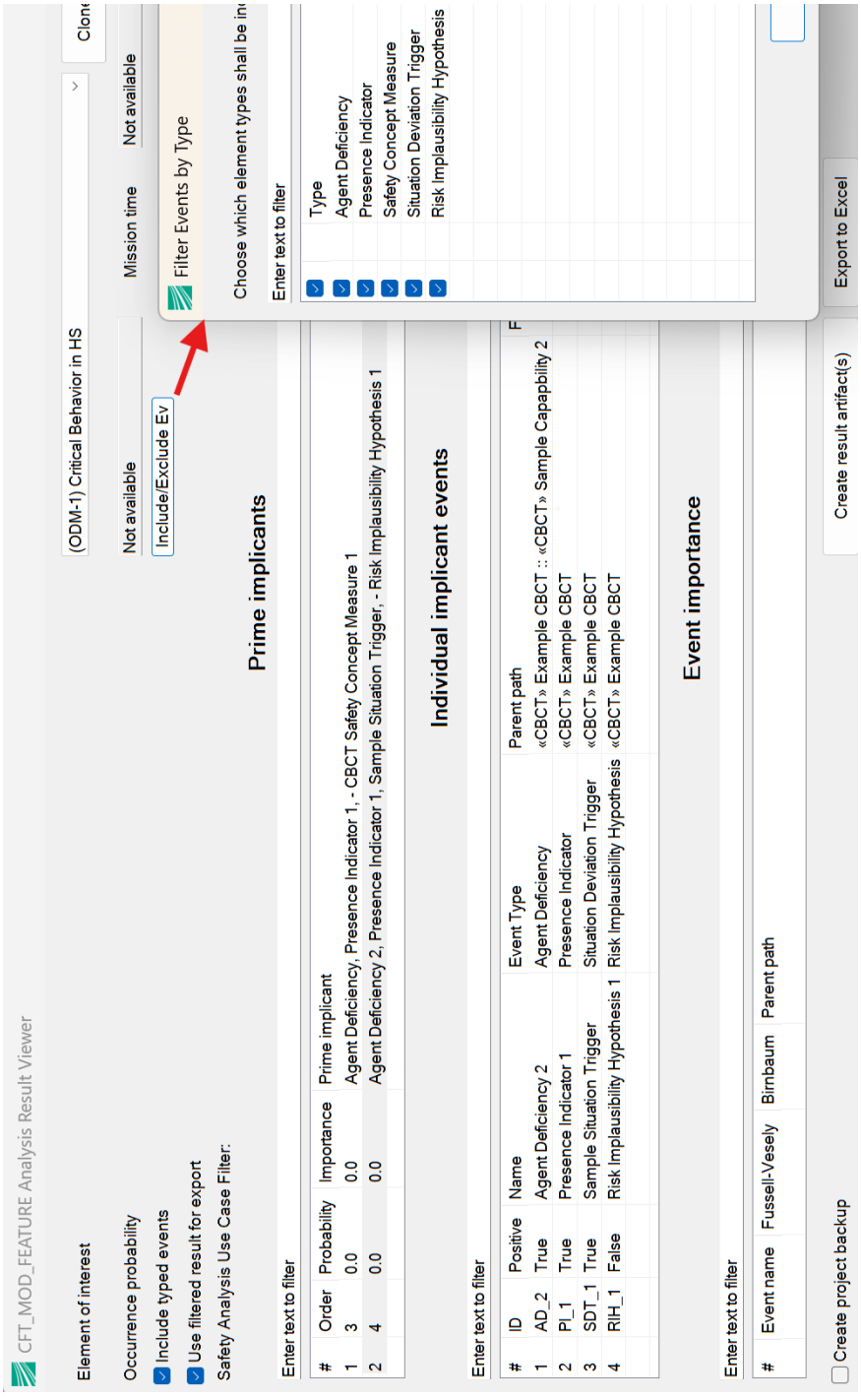


Figure B.2 safeTbox minimal cut set analysis result dialog with the possibility to filter and process CBCT minimal cut sets by using the formal CBCT element types.

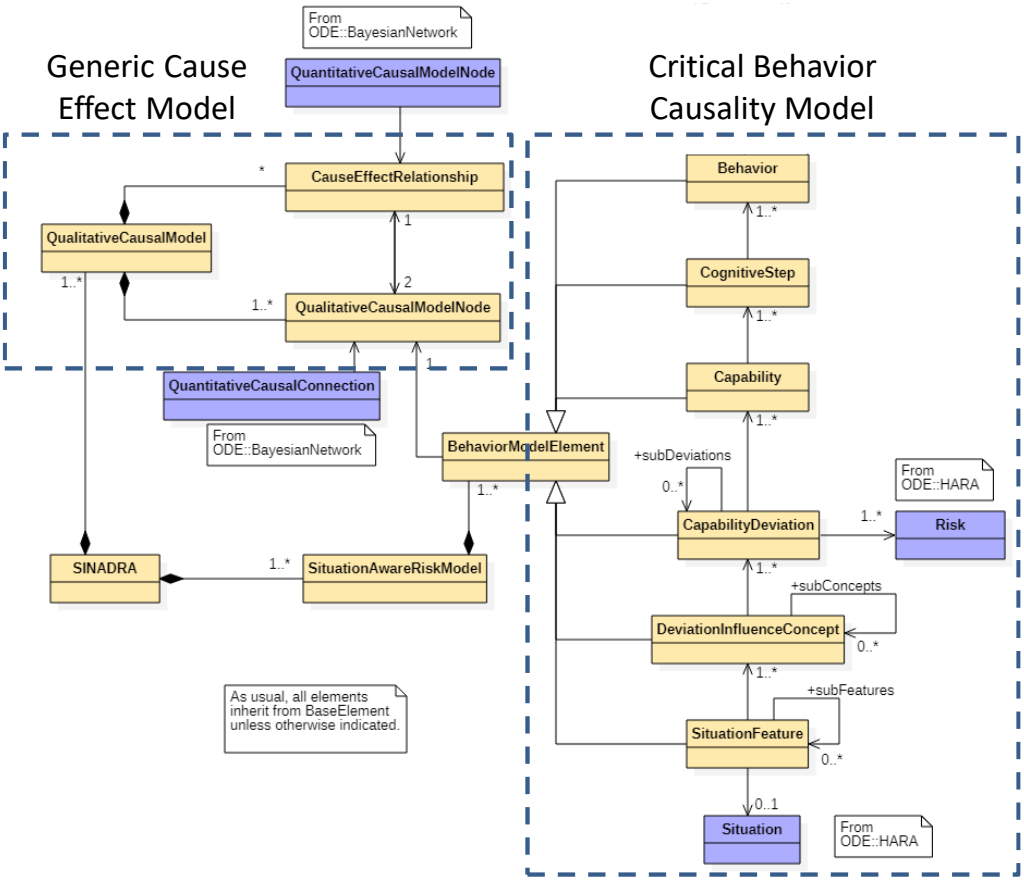


Figure B.3 ODE::SINADRA Metamodel in the Open Dependability Exchange (ODE) Format.

Appendix C Supplementary Case Study Material

C.1 Warehouse Case Study

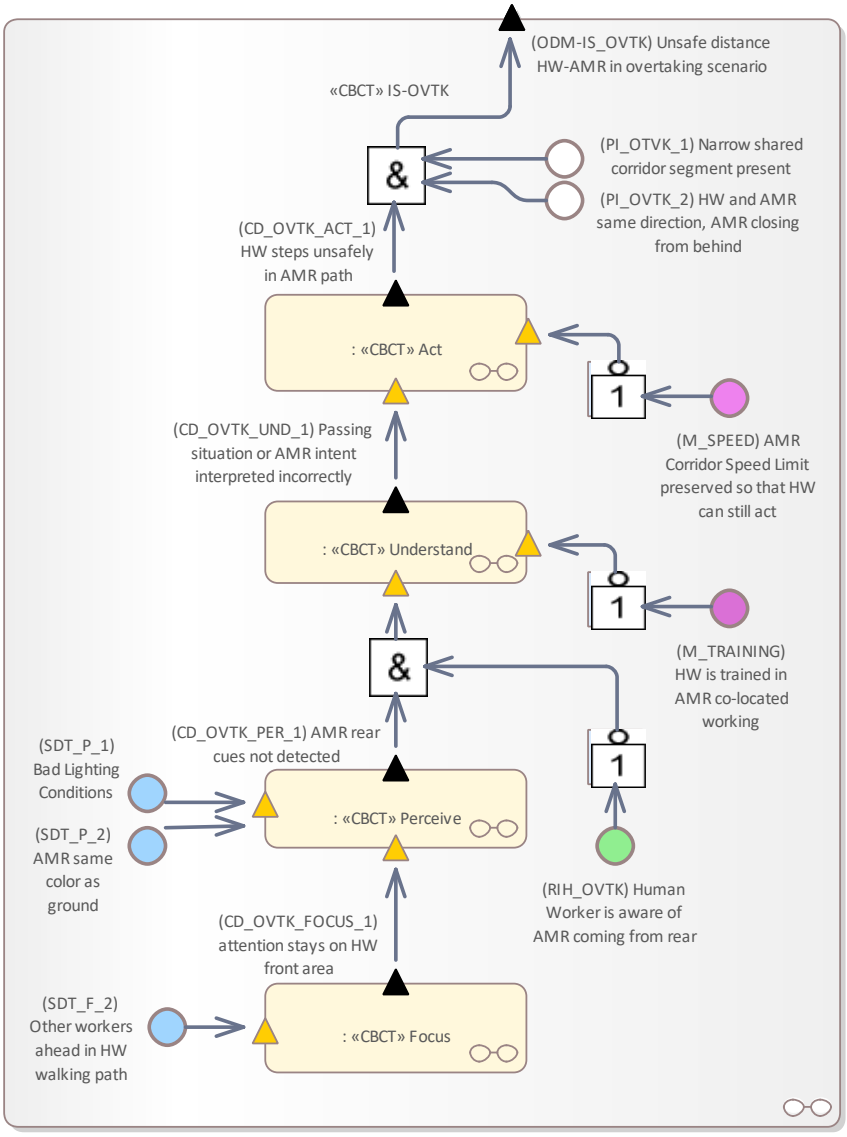


Figure C.1 SubCBCT *IS-OVTK* for the overtaking scenario in the warehouse case study.

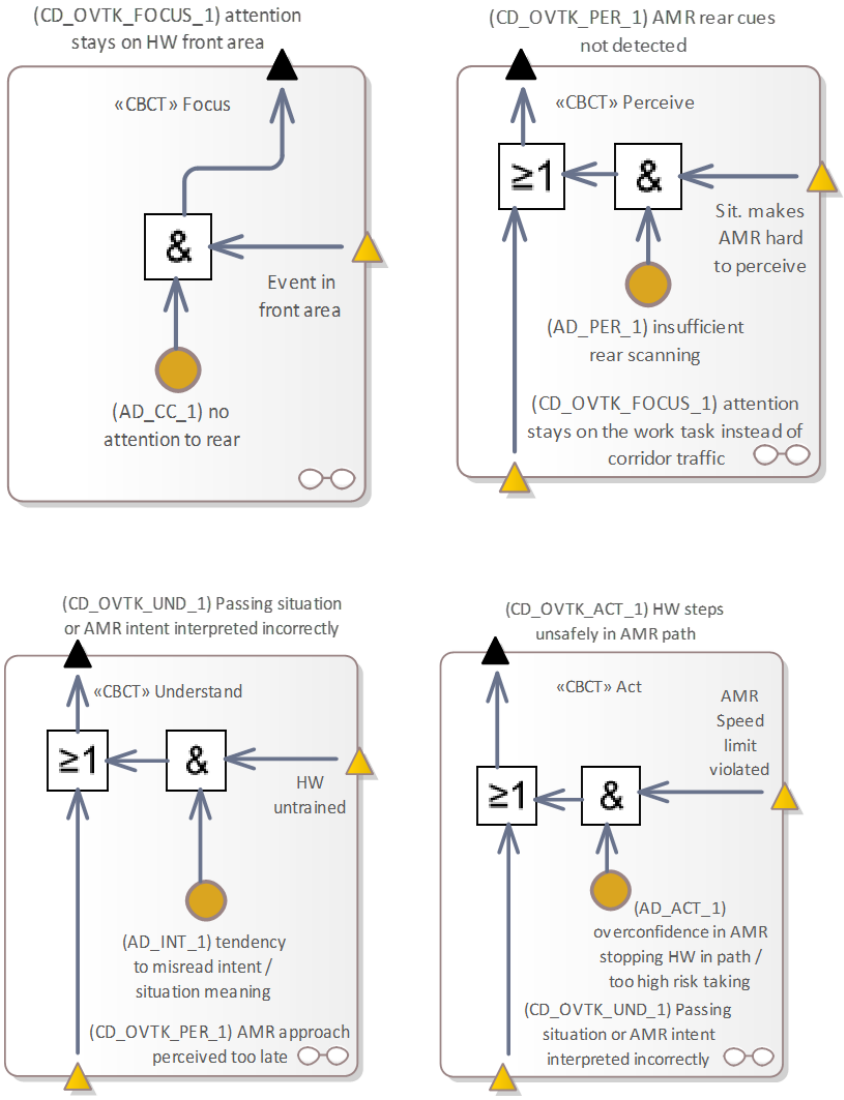


Figure C.2 Realization of four SubCBCTs Focus, Perceive, Understand, Act of initiator scenario IS-OVTK.

Table C.1 Four minimal cut sets of “ODM-IS-OVTK: Unsafe distance HW-AMR in overtaking scenario”.

#	ID	Positive	Name	Event Type	Parent path
1	PI_OVTK_2	True	HW and AMR same direction, AMR closing from behind	Presence Indicator	«CBCT» Human Worker :: «CBCT» IS-OVTK
2	PI_OTVK_1	True	Narrow shared corridor segment present	Presence Indicator	«CBCT» Human Worker :: «CBCT» IS-OVTK
3	SDT_F_2	True	Other workers ahead in HW walking path	Situation Deviation Trigger	«CBCT» Human Worker :: «CBCT» IS-OVTK
4	AD_CC_1	True	no attention to rear	Agent Deficiency	«CBCT» Human Worker :: «CBCT» IS-OVTK :: «CBCT» Focus
5	RIH_OVTK	False	Human Worker is aware of AMR coming from rear	Risk Implausibility Hypothesis	«CBCT» Human Worker :: «CBCT» IS-OVTK

#	ID	Positive	Name	Event Type	Parent path
1	SDT_P_2	True	AMR same color as ground	Situation Deviation Trigger	«CBCT» Human Worker :: «CBCT» IS-OVTK
2	SDT_P_1	True	Bad Lighting Conditions	Situation Deviation Trigger	«CBCT» Human Worker :: «CBCT» IS-OVTK
3	PI_OVTK_2	True	HW and AMR same direction, AMR closing from behind	Presence Indicator	«CBCT» Human Worker :: «CBCT» IS-OVTK
4	PI_OTVK_1	True	Narrow shared corridor segment present	Presence Indicator	«CBCT» Human Worker :: «CBCT» IS-OVTK
5	AD_PER_1	True	insufficient rear scanning	Agent Deficiency	«CBCT» Human Worker :: «CBCT» IS-OVTK :: «CBCT» Perceive
6	RIH_OVTK	False	Human Worker is aware of AMR coming from rear	Risk Implausibility Hypothesis	«CBCT» Human Worker :: «CBCT» IS-OVTK

#	ID	Positive	Name	Event Type	Parent path
1	PI_OVTK_2	True	HW and AMR same direction, AMR closing from behind	Presence Indicator	«CBCT» Human Worker :: «CBCT» IS-OVTK
2	PI_OTVK_1	True	Narrow shared corridor segment present	Presence Indicator	«CBCT» Human Worker :: «CBCT» IS-OVTK
3	AD_INT_1	True	tendency to misread intent / situation meaning	Agent Deficiency	«CBCT» Human Worker :: «CBCT» IS-OVTK :: «CBCT» Understand
4	M_TRAINING	False	HW is trained in AMR co-located working	Safety Concept Measure	«CBCT» Human Worker :: «CBCT» IS-OVTK

#	ID	Positive	Name	Event Type	Parent path
1	PI_OVTK_2	True	HW and AMR same direction, AMR closing from behind	Presence Indicator	«CBCT» Human Worker :: «CBCT» IS-OVTK
2	PI_OTVK_1	True	Narrow shared corridor segment present	Presence Indicator	«CBCT» Human Worker :: «CBCT» IS-OVTK
3	AD_ACT_1	True	overconfidence in AMR stopping HW in path / too high ri...	Agent Deficiency	«CBCT» Human Worker :: «CBCT» IS-OVTK :: «CBCT» Act
4	M_SPEED	False	AMR Corridor Speed Limit preserved so that HW can still...	Safety Concept Measure	«CBCT» Human Worker :: «CBCT» IS-OVTK

C.2 ADS Case Study

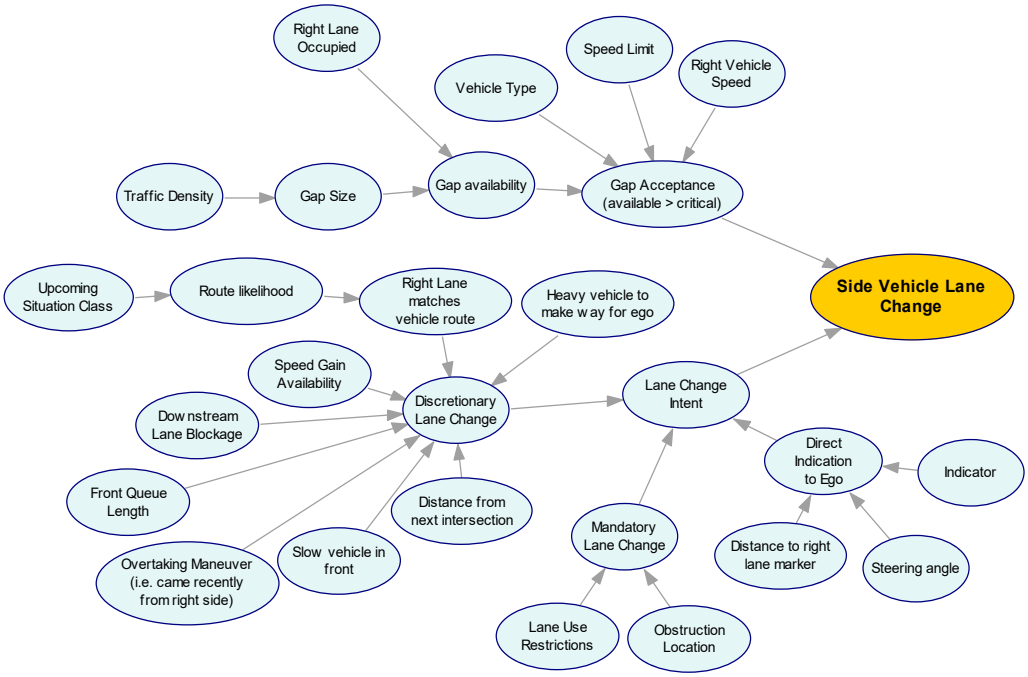


Figure C.3 Qualitative causal influences for side-vehicle lane change intent.

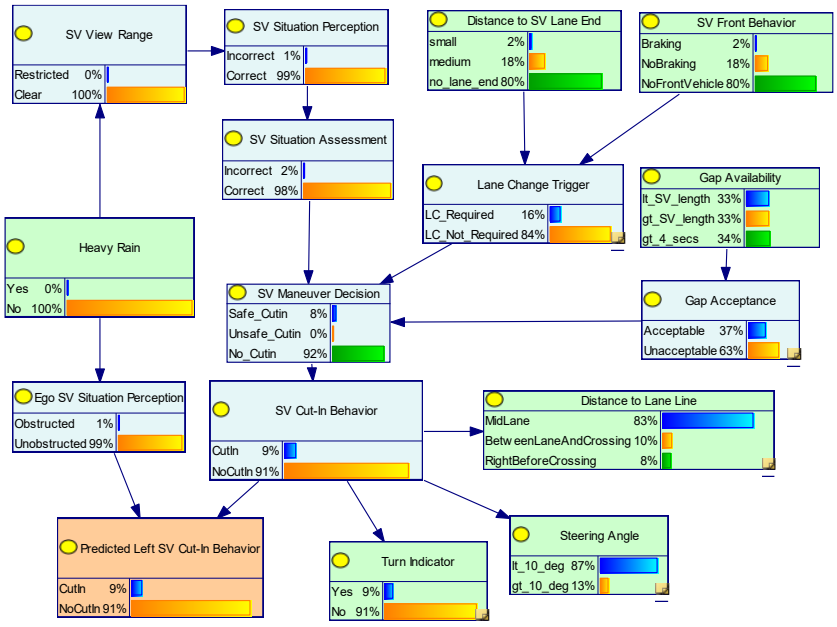


Figure C.4 Bayesian network model for side vehicle critical cut-in behavior intent prediction.

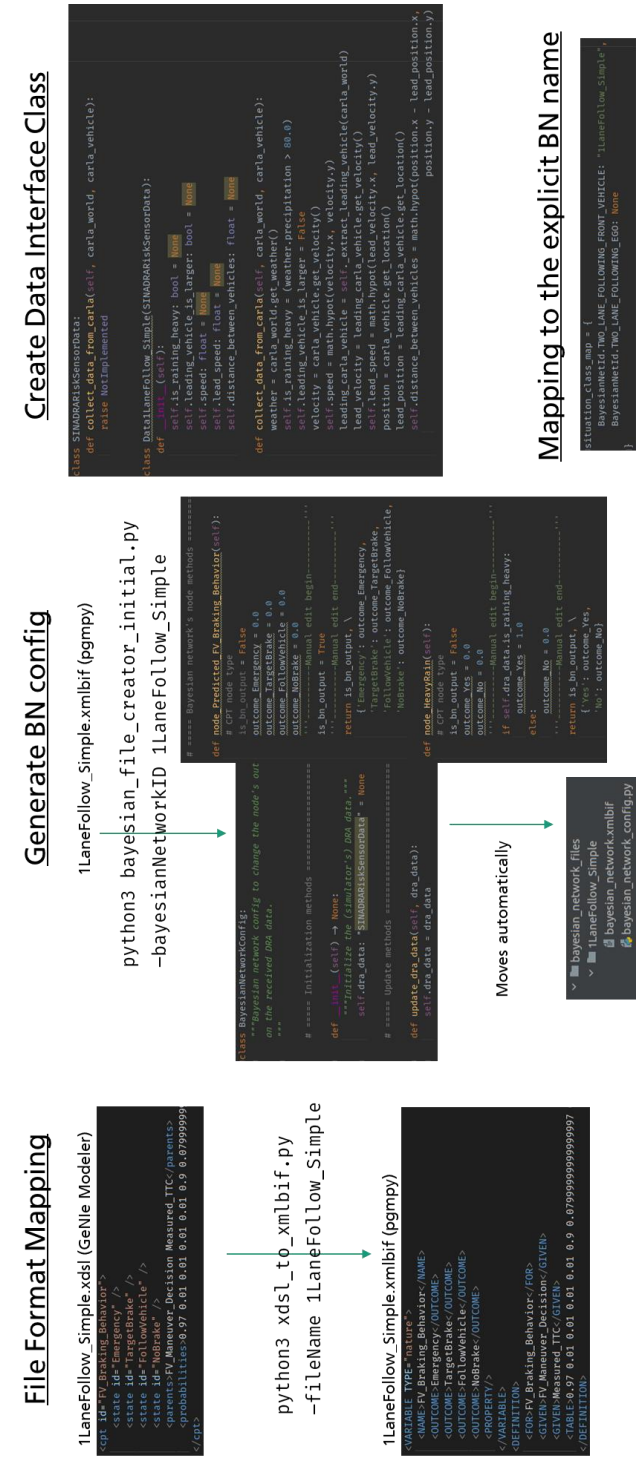


Figure C.5 Intermediate formats of semi-automated BN runtime inference Python components.

References

Adler, R., Feth, P., & Schneider, D. (2016). Safety engineering for autonomous vehicles. In *2016 46th Annual IEEE/IFIP International Conference on Dependable Systems and Networks Workshop (DSN-W)* (pp. 200–205). IEEE. https://doi.org/10.1109/DSN-W.2016.30

Adler, R., & Kemmann, S. (2014). Towards safe autonomic driving: Enhancing traditional hazard and risk analysis. In K. Berns, C. Schindler, K. Dreßler, B. Jörg, R. Kalmar, & G. Zolynski (Eds.), *Commercial vehicle technology 2014: Proceedings of the 3rd Commercial Vehicle Technology Symposium (CVT 2014)* (pp. 3–12). Shaker Verlag.

Adler, R., **Reich, J.**, & Hawkins, R. (2023). Structuring research related to dynamic risk management for autonomous systems. In J. Guiochet, S. Tonetta, E. Schoitsch, M. Roy, & F. Bitsch (Eds.), *Lecture Notes in Computer Science: Vol. 14182. Computer safety, reliability, and security. SAFECOMP 2023 workshops* (pp. 362–368). Springer Nature Switzerland. https://doi.org/10.1007/978-3-031-40953-0_30

aGrUM team. (2025, October 28). *aGrUM/pyAgrum*. https://agrum.gitlab.io/pages/agrum.html

Almusawi, A., Albdairi, M., & Qadri, S. S. S. M. (2024). Integrating Autonomous Vehicles (AVs) into Urban Traffic: Simulating Driving and Signal Control. *Applied Sciences*, *14*(19), 8851. https://doi.org/10.3390/app14198851

Arnez, F., Ollier, G., Radermacher, A., Adedjouma, M., Gerasimou, S., Mraidha, C., & Terrier, F. (2022). Skeptical Dynamic Dependability Management for Automated Systems. In *2022 25th Euromicro Conference on Digital System Design (DSD)* (pp. 118–125). IEEE. https://doi.org/10.1109/DSD57027.2022.00025

Avizienis, A., Laprie, J.-C., Randell, B., & Landwehr, C. (2004). Basic concepts and taxonomy of dependable and secure computing. *IEEE Transactions on Dependable and Secure Computing*, *1*(1), 11–33. https://doi.org/10.1109/TDSC.2004.2

BayesFusion. (2025, November 8). *SMILE Engine – BayesFusion*. https://www.bayesfusion.com/smile/

Chen, X., Xi, Z., Xu, Y., Xiong, Z., Ding, X., & Wang, H. (2025). Balancing safety and efficiency for autonomous vehicles at urban uncontrolled crosswalk: Challenges and countermeasures. *Accident Analysis & Prevention*, *220*, Article 108111. https://doi.org/10.1016/j.aap.2025.108111

Cioroaica, E., Blanco, J., & Rossi, B. (2022). Timing Model for Predictive Simulation of Safety-critical Systems. In *Proceedings of the 17th International Conference on Software Technologies* (pp. 331–339). SCITEPRESS - Science and Technology Publications. https://doi.org/10.5220/0011317000003266

Czarnecki, K. (2018a). *Operational World Model Ontology for Automated Driving Systems - Part 1: Road Structure.* Waterloo Intelligent Systems Engineering Lab (WISE) Report, University of Waterloo. https://doi.org/10.13140/RG.2.2.15521.30568

Czarnecki, K. (2018b). *Operational World Model Ontology for Automated Driving Systems - Part 2: Road Users, Animals, Other Obstacles, and Environmental Conditions.*

Waterloo Intelligent Systems Engineering Lab (WISE) Report, University of Waterloo. https://doi.org/10.13140/RG.2.2.11327.00165

Domis, D. (2012). *Integrating Fault Tree Analysis and Component-Oriented Model-Based Design of Embedded Systems* [Doctoral Dissertation]. Technical University Kaiserslautern, Germany.

Eggert, J. (2018). Risk estimation for driving support and behavior planning in intelligent vehicles. *At - Automatisierungstechnik*, *66*(2), 119–131. https://doi.org/10.1515/auto-2017-0132

Eggert, J., Damerow, F., & Klingelschmitt, S. (2015). The Foresighted Driver Model. In *2015 IEEE Intelligent Vehicles Symposium (IV)* (pp. 322–329). IEEE / Institute of Electrical and Electronics Engineers Incorporated. https://doi.org/10.1109/IVS.2015.7225706

Eggert, J., Klingelschmitt, S., & Damerow, F. (2015). The foresighted driver: Future ADAS based on generalized predictive risk estimation. In *Proceedings of the 3rd International Symposium on Future Active Safety Technology Towards Zero Traffic Accidents (FAST-zero 2015)* (pp. 93–100).

Feth, P. (2020). *Dynamic Behavior Risk Assessment for Autonomous Systems* [Doctoral Dissertation]. Technical University Kaiserslautern, Germany. https://doi.org/10.24406/publica-fhg-283031

Feth, P., Schneider, D., & Adler, R. (2017). A conceptual safety supervisor definition and evaluation framework for autonomous systems. In S. Tonetta, E. Schoitsch, & F. Bitsch (Eds.), *Lecture Notes in Computer Science: Vol. 10488. 36th International Conference, SAFECOMP 2017, Trento, Italy, September 13–15, 2017, proceedings* (pp. 135–148). Springer International Publishing. https://doi.org/10.1007/978-3-319-66266-4_9

German Association of the Automotive Industry (2023). *VDA-Empfehlung 702 Version 2.0:* Situationskatalog E-Parameter nach ISO 26262-3:2018.

Hasuo, I., Eberhart, C., Haydon, J., Dubut, J., Bohrer, R., Kobayashi, T., Pruekprasert, S., Zhang, X.-Y., Pallas, E. A., Yamada, A., Suenaga, K., Ishikawa, F., Kamijo, K., Shinya, Y., & Suetomi, T. (2023). Goal-Aware RSS for Complex Scenarios via Program Logic. *IEEE Transactions on Intelligent Vehicles*, *8*(4), 3040–3072. https://doi.org/10.1109/TIV.2022.3169762

Herd, B., Kreutz, A., & Trapp, M. (2025). SOTIF Analysis using Fuzzy Cause Trees. *Proceedings of EDCC*.

Hillen, D., & **Reich, J.** (2023). A pledge for safety-aware operational environment analysis [Position Paper]. In *SAFECOMP 2023, Toulouse, France.* HAL. https://hal.science/hal-04191761

Hillen, D., **Reich, J.**, Laxman, N., Frey, J., Otsuka, S., Ogata, T., & Di Paola, D. (2025). From Abstract to Action: Tailored Environment Taxonomies for More Complete ADS Safety Analyses. In P. Katsaros (Ed.), *Lecture Notes in Computer Science: Vol. 15755. Model-based safety and assessment* (pp. 429–443). Springer Nature Switzerland. https://doi.org/10.1007/978-3-032-05073-1_28

IEEE VT/ITS/AV Decision Making Working Group (2022). Literature Review on Kinematic Properties of Road Users for Use on Safety-Related Models for Automated Driving Systems.

Institute of Electrical and Electronics Engineers (2022). *IEEE 2846-2022: Assumptions in Safety-Related Models for Automated Driving Systems*.

Institute of Electrical and Electronics Engineers (2025). *IEEE 3321-2024: Recommended Practice for the Application of Assumptions on Reasonably Foreseeable Behavior of Other Road Users in Safety-Related Models*.

International Electrotechnical Commission (2010). *IEC 61508 2. Edition: Functional Safety of Electrical/Electronic/Programmable Electronic Safety-related Systems*.

International Organization for Standardization (2018). *ISO 26262 Second Edition:* Road Vehicles - Functional Safety.

International Organization for Standardization (2022). *ISO 21448:* Road vehicles — Safety of the intended functionality.

International Organization for Standardization (2023). *ISO 34503:* Road Vehicles — Test scenarios for automated driving systems — Specification for operational design domain.

Ishigooka, T., Yamada, H., Otsuka, S., Kanekawa, N., & Takahashi, J. (2022). Symbiotic Safety: Safe and Efficient Human-Machine Collaboration by utilizing Rules. In *2022 Design, Automation & Test in Europe Conference & Exhibition (DATE)* (pp. 280–281). IEEE. https://doi.org/10.23919/DATE54114.2022.9774687

Jan, Q. H., Wolf, P., Berns, K., **Reich, J.**, & Wellstein, M. (2022). Integration of human skeleton posture models into REACTiON for realizing dynamic risk management. In K. Berns, K. Dressler, R. Kalmar, N. Stephan, R. Teutsch, & M. Thul (Eds.), *Proceedings. Commercial Vehicle Technology 2022* (pp. 109–120). Springer Fachmedien Wiesbaden. https://doi.org/10.1007/978-3-658-40783-4_8

Johansson, R., & Nilsson, J. (2016). The need for an environment perception block to address all ASIL levels simultaneously. In *2016 IEEE Intelligent Vehicles Symposium (IV)* (pp. 1–4). IEEE. https://doi.org/10.1109/IVS.2016.7535354

Kemmann, S. (2015). *SAHARA—A structured approach for hazard analysis and risk assessments* [Doctoral Dissertation]. Technical University Kaiserslautern, Germany. KLUEDO. https://nbn-resolving.org/urn:nbn:de:hbz:386-kluedo-40270

Khastgir, S., Birrell, S., Dhadyalla, G., Sivencrona, H., & Jennings, P. (2017). Towards increased reliability by objectification of Hazard Analysis and Risk Assessment (HARA) of automated automotive systems. *Safety Science*, *99*, 166–177. https://doi.org/10.1016/j.ssci.2017.03.024

Khastgir, S., Sivencrona, H., Dhadyalla, G., Billing, P., Birrell, S., & Jennings, P. (2017). Introducing ASIL inspired dynamic tactical safety decision framework for automated vehicles. In *IEEE ITSC 2017: 20th International Conference on Intelligent Transportation Systems : Mielparque Yokohama in Yokohama, Kanagawa, Japan, October 16-19, 2017* (pp. 1–6). IEEE. https://doi.org/10.1109/ITSC.2017.8317868

Klingelschmitt, S., Damerow, F., Willert, V., & Eggert, J. (2016). Probabilistic situation assessment framework for multiple, interacting traffic participants in generic traffic scenes. In *2016 IEEE Intelligent Vehicles Symposium (IV)* (pp. 1141–1148). IEEE. https://doi.org/10.1109/IVS.2016.7535533

Königshof, H., Oboril, F., Scholl, K.-U., & Stiller, C. (2022). A Parameter Analysis on RSS in Overtaking Situations on German Highways. In *2022 IEEE Intelligent Vehicles Symposium (IV)* (pp. 1081–1086). IEEE. https://doi.org/10.1109/IV51971.2022.9827086

Koopman, P. (2022). *How Safe Is Safe Enough? Measuring and Predicting Autonomous Vehicle Safety*. Independently Published.

Koopman, P., Osyk, B., & Weast, J. (2019, October 31). *Autonomous Vehicles Meet the Physical World: RSS, Variability, Uncertainty, and Proving Safety (Expanded Version)*. https://arxiv.org/pdf/1911.01207

Koschi, M., & Althoff, M. (2021). Set-based prediction of traffic participants considering occlusions and traffic rules. *IEEE Transactions on Intelligent Vehicles*, *6*(2), 249–265. https://doi.org/10.1109/TIV.2020.3017385

Krajewski, R., Bock, J., Kloeker, L., & Eckstein, L. (2018). The highD Dataset: A Drone Dataset of Naturalistic Vehicle Trajectories on German Highways for Validation of Highly Automated Driving Systems. In *2018 IEEE Intelligent Transportation Systems Conference: November 4-7, Maui, Hawaii* (pp. 2118–2125). IEEE. https://doi.org/10.1109/ITSC.2018.8569552

Kramer, B., Neurohr, C., Büker, M., Böde, E., Fränzle, M., & Damm, W. (2020). Identification and Quantification of Hazardous Scenarios for Automated Driving. In M. Zeller & K. Höfig (Eds.), *Lecture Notes in Computer Science. MODEL-BASED SAFETY AND ASSESSMENT: 7th international symposium, imbsa 2020* (Vol. 12297, pp. 163–178). SPRINGER NATURE. https://doi.org/10.1007/978-3-030-58920-2_11

Kubasáková, I., Kubáňová, J., Benčo, D., & Fábryová, N. (2024). Application of Autonomous Mobile Robot as a Substitute for Human Factor in Order to Increase Efficiency and Safety in a Company. *Applied Sciences*, *14*(13), 5859. https://doi.org/10.3390/app14135859

Li, H., Sun, X., Zhuang, C., & Li, X. (2023, September 22). *On the Robotic Uncertainty of Fully Autonomous Traffic*. http://arxiv.org/pdf/2309.12611

Liu, S., Wang, X., Hassanin, O., Xu, X., Yang, M., Hurwitz, D., & Wu, X. (2021). Calibration and evaluation of responsibility-sensitive safety (RSS) in automated vehicle performance during cut-in scenarios. *Transportation Research Part C: Emerging Technologies*, *125*, 103037. https://doi.org/10.1016/j.trc.2021.103037

Machin, M., Guiochet, J., Waeselynck, H., Blanquart, J.-P., Roy, M., & Masson, L. (2018). SMOF: A safety monitoring framework for autonomous systems. *IEEE Transactions on Systems, Man, and Cybernetics: Systems*, *48*(5), 702–715. https://doi.org/10.1109/TSMC.2016.2633291

Masson, L., Guiochet, J., Waeselynck, H., Cabrera, K., Cassel, S., & Törngren, M. (2018). Tuning permissiveness of active safety monitors for autonomous systems. In A. M. Dutle, C. A. Muñoz, & A. J. Narkawicz (Eds.), *Lecture Notes in Computer Science: Vol. 10811. NASA formal methods: 10th International Symposium, NFM 2018, Newport News, VA, USA, April 17–19, 2018, proceedings* (pp. 333–348). Springer International Publishing. https://doi.org/10.1007/978-3-319-77935-5_23

Naumann, M., Wirth, F., Oboril, F., Scholl, K.-U., Elli, M. S., Alvarez, I., Weast, J., & Stiller, C. (2021). On Responsibility Sensitive Safety in Car-following Situations - A Parameter Analysis on German Highways. In *2021 IEEE Intelligent Vehicles Symposium (IV)* (pp. 83–90). IEEE. https://doi.org/10.1109/IV48863.2021.9575420

Nikolaou, P., Savva, A., Sorokos, I., Aslansefat, K., Missaoui, S., Naveed, A. M., Hillen, D., Lorenz, M., Walker, M. D., Papoutsakis, M., Gerasimou, S., Kolios, P., Papadopoulos, Y., Reich, J., Ioannidis, S., & Michael, M. K. (2025). Multi-Partner Project: Safe, Secure and Dependable Multi-UAV Systems for Search and Rescue Operations. In *2025 Design, Automation & Test in Europe Conference (DATE)* (pp. 1–7). IEEE. https://doi.org/10.23919/DATE64628.2025.10992739

Oboril, F., & Scholl, K.-U. (2020). Risk-Aware Safety Layer for AV Behavior Planning. In *2020 IEEE Intelligent Vehicles Symposium (IV)* (pp. 1922–1928). IEEE. https://doi.org/10.1109/IV47402.2020.9304563

Oboril, F., & Scholl, K.-U. (2021). RSS+: Pro-Active Risk Mitigation for AV Safety Layers based on RSS. In *2021 IEEE Intelligent Vehicles Symposium (IV)* (pp. 99–106). IEEE. https://doi.org/10.1109/IV48863.2021.9575731

Ollier, G., Arnez, F., Adedjouma, M., Lallement, R., Gerasimou, S., & Mraidha, C. (2023). Towards an Ontological Methodology for Dynamic Dependability Management of Unmanned Aerial Vehicles. In *DroneSE and RAPIDO: System Engineering for constrained embedded systems* (pp. 12–19). ACM. https://doi.org/10.1145/3579170.3579265

Peffers, K., Tuunanen, T., Gengler, C. E., Rossi, M., Hui, W., Virtanen, V., & Bragge, J. (2006). The design science research process: A model for producing and presenting information systems research. In *DESRIST International Conference on Design Science Research in Information Systems and Technology, Claremont, CA, USA, February 24–25, 2006*.

Pek, C., Koschi, M., Werling, M., & Althoff, M. (2017). Enhancing motion safety by identifying safety-critical passageways. In *2017 IEEE 56th Annual Conference on Decision and Control (CDC)* (pp. 320–326). IEEE. https://doi.org/10.1109/CDC.2017.8263685

Pek, C., Manzinger, S., Koschi, M., & Althoff, M. (2020). Using online verification to prevent autonomous vehicles from causing accidents. *Nature Machine Intelligence*, *2*(9), 518–528. https://doi.org/10.1038/s42256-020-0225-y

Reich, J., Gerber, P., Laxman, N., Schneider, D., Ogata, T., Otsuka, S., & Ishigooka, T. (2022). Engineering dynamic risk and capability models to improve cooperation efficiency between human workers and autonomous mobile robots in shared spaces. In C. Seguin, M. Zeller, & T. Prosvirnova (Eds.), *Lecture Notes in Computer Science: Vol. 13525. Model-based safety and assessment* (pp. 237–251). Springer International Publishing. https://doi.org/10.1007/978-3-031-15842-1_17

Reich, J., Hillen, D., Frey, J., Laxman, N., Ogata, T., Di Paola, D., Otsuka, S., & Watanabe, N. (2023). Concept and Metamodel to Support Cross-Domain Safety Analysis for ODD Expansion of Autonomous Systems. In *Computer Safety, Reliability, and Security: 42nd International Conference, SAFECOMP 2023, Toulouse, France, September 20–22, 2023, Proceedings* (pp. 165–178). Springer-Verlag. https://doi.org/10.1007/978-3-031-40923-3_13

Reich, J., & Trapp, M. (2020). SINADRA: Towards a Framework for Assurable Situation-Aware Dynamic Risk Assessment of Autonomous Vehicles. In *2020 16th European Dependable Computing Conference (EDCC)* (pp. 47–50). IEEE. https://doi.org/10.1109/EDCC51268.2020.00017

Reich, J., Wellstein, M., Sorokos, I., Oboril, F., & Scholl, K.-U. (2021). Towards a software component to perform situation-aware dynamic risk assessment for autonomous vehicles. In R. Adler, A. Bennaceur, S. Burton, A. Di Salle, N. Nostro, R. Løvenstein Olsen, S. Saidi, P. Schleiss, D. Schneider, & H.-P. Schwefel (Eds.), *Communications in Computer and Information Science: Vol. 1462. Dependable computing - EDCC 2021 workshops: DREAMS, DSOGRI, SERENE 2021, Munich, Germany, September 13, 2021, proceedings* (pp. 3–11). Springer. https://doi.org/10.1007/978-3-030-86507-8_1

Rodrigues de Campos, G., Kianfar, R., & Brännström, M. (2021). Precautionary Safety for Autonomous Driving Systems: Adapting Driving Policies to Satisfy Quantitative Risk Norms. In *2021 IEEE International Intelligent Transportation Systems Conference (ITSC)* (pp. 645–652). IEEE. https://doi.org/10.1109/ITSC48978.2021.9564879

Salvi, A., Weiss, G., Trapp, M., Oboril, F., & Buerkle, C. (2022). Safety Implications of Runtime Adaptation to Changing Operating Conditions. In *2022 IEEE 25th International Conference on Intelligent Transportation Systems (ITSC)* (pp. 2444–2449). IEEE. https://doi.org/10.1109/ITSC55140.2022.9922192

Schneider, D., **Reich, J.**, Adler, R., & Liggesmeyer, P. (2024, January 24). *Dynamic Risk Management in Cyber Physical Systems*. http://arxiv.org/pdf/2401.13539v2

Schneider, D., & Trapp, M. (2013). Conditional Safety Certification of Open Adaptive Systems. *ACM Transactions on Autonomous and Adaptive Systems*, *8*(2), 1–20. https://doi.org/10.1145/2491465.2491467

Schneider, D., Trapp, M., Papadopoulos, Y., Armengaud, E., Zeller, M., & Höfig, K. (2015). WAP: Digital dependability identities. In *2015 IEEE 26th International Symposium on Software Reliability Engineering (ISSRE)* (pp. 324–329). IEEE. https://doi.org/10.1109/ISSRE.2015.7381825

Scholtes, M., Westhofen, L., Turner, L. R., Lotto, K., Schuldes, M., Weber, H., Wagener, N., Neurohr, C., Bollmann, M., Körtke, F., Hiller, J., Hoss, M., Bock, J., & Eckstein, L. (2021). 6-Layer Model for a Structured Description and Categorization of Urban Traffic and Environment. *IEEE Access*, *9*, 59131–59147. https://doi.org/10.1109/ACCESS.2021.3072739

Schreier, M., Willert, V., & Adamy, J. (2016). An Integrated Approach to Maneuver-Based Trajectory Prediction and Criticality Assessment in Arbitrary Road Environments. *IEEE Transactions on Intelligent Transportation Systems*, *17*(10), 2751–2766. https://doi.org/10.1109/TITS.2016.2522507

SESAME Project Consortium. (2022a). *Deliverable D4.2: Safety-Targeted ODE and EDDI specification*. https://www.sesame-project.org/

SESAME Project Consortium. (2022b). *Deliverable D5.4 Tailorability of EDDIs*. https://www.sesame-project.org/

SESAME Project Consortium. (2022c). *Deliverable D7.1 Runtime Safety and Security Concept.*

SESAME Project Consortium. (2022d). *Deliverable D7.2 - Tools for Generation of Runtime EDDIs*. https://www.sesame-project.org/

SESAME Project Consortium. (2023). *Deliverable D4.6: Tools for Automated Safety Analysis of MRS and for Production of EDDIs*. https://www.sesame-project.org/

Shalev-Shwartz, S., Shammah, S., & Shashua, A. (2017). *On a Formal Model of Safe and Scalable Self-driving Cars.* Intel/Mobileye. http://arxiv.org/pdf/1708.06374v5

Snowden, D. (2005). Strategy in the context of uncertainty. *Handbook of Business Strategy*, *6*(1), 47–54. https://doi.org/10.1108/08944310510556955

Trapp, M., Schneider, D., & Weiss, G. (2018). Towards safety-awareness and dynamic safety management. In *2018 14th European Dependable Computing Conference (EDCC)* (pp. 107–111). IEEE. https://doi.org/10.1109/EDCC.2018.00027

Trautman, P., & Krause, A. (2010). Unfreezing the robot: Navigation in dense, interacting crowds. In *2010 IEEE/RSJ International Conference on Intelligent Robots and Systems* (pp. 797–803). IEEE. https://doi.org/10.1109/IROS.2010.5654369

Treiber, M., Hennecke, A., & Helbing, D. (2000). Congested traffic states in empirical observations and microscopic simulations. *Physical Review. E, Statistical Physics, Plasmas, Fluids, and Related Interdisciplinary Topics*, *62*(2 Pt A), 1805–1824. https://doi.org/10.1103/PhysRevE.62.1805

U.S. Department of Defense (2023). *MIL-STD-882E w/CHANGE 1: Department of Defense Standard Practice: System Safety*. https://www.cto.mil/wp-content/uploads/2025/07/MIL-STD-882E-w_CHANGE-1.pdf

Ulmen, J., Wellstein, M., Mark, C., Sundaram, G., Görges, D., & **Reich, J.** (2023). Stochastic MPC with Situation-Aware Dynamic Risk Assessment for Autonomous Driving. In *2023 IEEE 26th International Conference on Intelligent Transportation Systems (ITSC)* (pp. 3478–3484). IEEE. https://doi.org/10.1109/ITSC57777.2023.10421964

Wang, Q. (2024). Study on the impact of autonomous driving technology on the economy and society. *SHS Web of Conferences*, *208*, Article 01022. https://doi.org/10.1051/shsconf/202420801022

Warg, F., Gassilewski, M., Tryggvesson, J., Izosimov, V., Werneman, A., & Johansson, R. (2016). Defining autonomous functions using iterative hazard analysis and requirements refinement. In A. Skavhaug, J. Guiochet, E. Schoitsch, & F. Bitsch (Eds.), *Lecture Notes in Computer Science: Vol. 9923. Computer safety, reliability, and security: SAFECOMP 2016 workshops, ASSURE, DECSoS, SASSUR, and TIPS, Trondheim, Norway, September 20, 2016, proceedings* (pp. 286–297). Springer. https://doi.org/10.1007/978-3-319-45480-1_23

Warg, F., Skoglund, M., Thorsen, A., Johansson, R., Brännström, M., Gyllenhammar, M., & Sanfridson, M. (2020). The Quantitative Risk Norm - A Proposed Tailoring of HARA for ADS. In *2020 50th Annual IEEE/IFIP International Conference on Dependable Systems and Networks Workshops (DSN-W)* (pp. 86–93). IEEE. https://doi.org/10.1109/DSN-W50199.2020.00026

Wellstein, M. (2021). *Development of a Bayesian Network for Situation-Aware Lane Change Prediction based on the highD Dataset* [Master's Thesis]. Technical University Kaiserslautern, Germany.

Westhofen, L. (2025, March 9). *GitHub: The Automotive Urban Traffic Ontology*. https://github.com/lu-w/auto

Xu, X., Wang, X., Wu, X., Hassanin, O., & Chai, C. (2021). Calibration and evaluation of the Responsibility-Sensitive Safety model of autonomous car-following maneuvers using naturalistic driving study data. *Transportation Research Part C: Emerging Technologies*, *123*, 102988. https://doi.org/10.1016/j.trc.2021.102988

Zeller, M. (2022). Component Fault and Deficiency Tree (CFDT): Combining Functional Safety and SOTIF Analysis. In C. Seguin, M. Zeller, & T. Prosvirnova (Eds.), *Lecture Notes in Computer Science: Vol. 13525. Model-based safety and assessment* (Vol. 13525, pp. 146–152). Springer International Publishing. https://doi.org/10.1007/978-3-031-15842-1_11

Zeller, M., Sorokos, I., **Reich, J.**, Adler, R., & Schneider, D. (2023). Open Dependability Exchange Metamodel: A Format to Exchange Safety Information. In *2023 Annual Reliability and Maintainability Symposium (RAMS)* (pp. 1–7). IEEE. https://doi.org/10.1109/RAMS51473.2023.10088190

Zimmer, B. (2008). *Safety concept trees: An approach to a formalized depiction of safety concepts for safety-critical systems* [Diploma Thesis]. Technical University Kaiserslautern, Germany.

Curriculum Vitae

Education	2010-2014	Bachelor Applied Computer Science (Production and Automotive Technology) *Technical University Kaiserslautern, Germany*
	2014-2017	Master Applied Computer Science (Automotive Technology) *Technical University Kaiserslautern, Germany*
Work Experience	2017-2020	Scientist *Fraunhofer IESE, Kaiserslautern, Germany*
	2020-2023	Expert Scientist „Dynamic Assurances for Connected Autonomous Systems“ *Fraunhofer IESE, Kaiserslautern, Germany*
	2024-today	Department Head „Safety Engineering” *Fraunhofer IESE, Kaiserslautern, Germany*