Autonomous Driving – the “uncrashable” car?

What it takes to make self-driving vehicles safe and reliable traffic participants

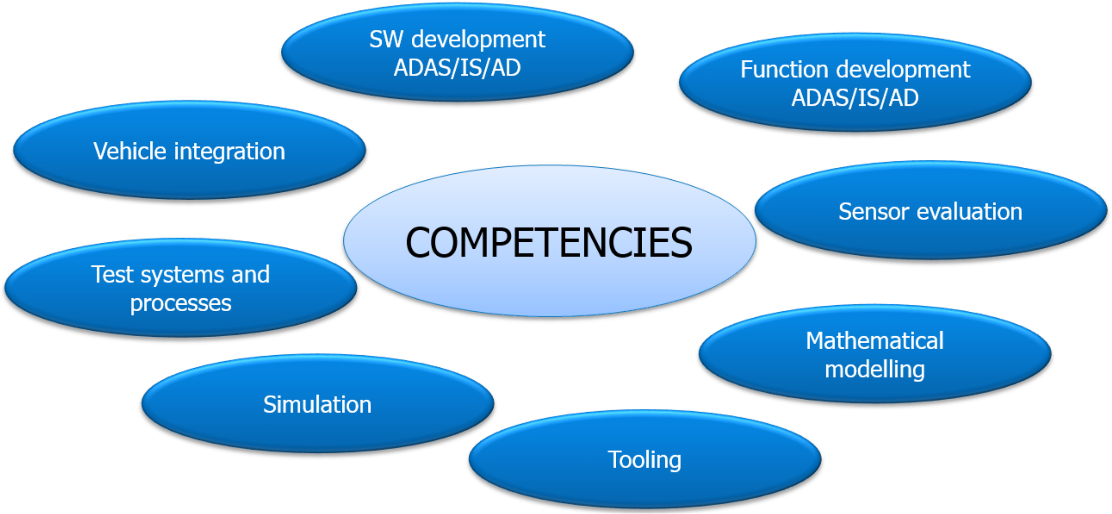
Dr. Frank Keck, MMB Conference 2018, Erlangen, Germany

Agenda

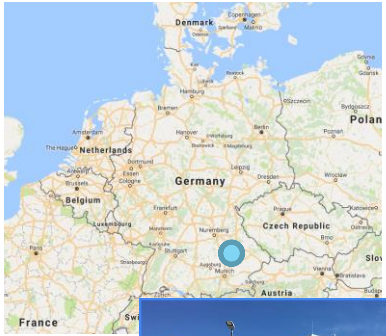
- Zukunft Mobility GmbH
- Motivation
- Traditional development efforts
- Novel approach
  - Use-case based specification
  - Simulation
  - Data analysis
  - Machine learning
- Virtualization techniques
- How to handle further uncertainties?
- Conclusion and outlook

Zukunft Mobility GmbH

- Founded in December 2017 as a transfer of enterprise from IEE Sensing Germany GmbH
- Subsidiary of ZF Friedrichshafen (via Zukunft Ventures GmbH)
- Located in Kösching, Germany
- 25 Employees as of February 2018
- Core competencies:

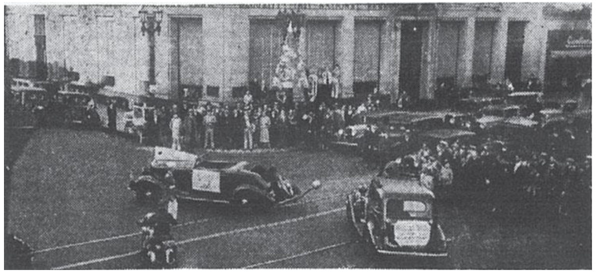


ZUKUNFT VENTURES
A COMPANY OF ZF



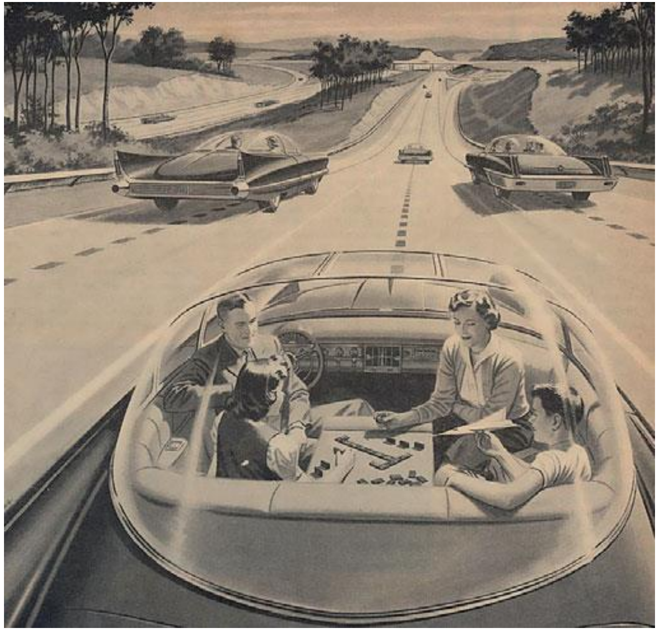
Motivation
Autonomous Driving – History

Remote-controlled car at Safety Parade, USA 1930*



„Prometheus Test Car" PROgraMme for a European Traffic of Highest Efficiency and Unprecedented Safety, BRD 1987

Autonomous driving vision, LIFE Magazine, USA 1956*



*Source: Winner (Hrsg.), "Autonomes Fahren", Springer Open, 2015

Motivation
Autonomous Driving – the “mega trend” in the spotlight…

Volvo to seek volunteers for self-driving car trial in UK

London testing of semi-autonomous vehicles on public roads is to be conducted in tandem with one in Gothenburg, Sweden

Source: The Guardian, Feb. 2017

vision Magazine   ZF.com

Vision Zero   Smart Mobility

Source: zf.com, Feb. 2018

Autonomous driving is today’s biggest game changer

Intel announced a collaboration with entertainment company Warner Bros. to develop in-cabin, immersive experiences in autonomous vehicle (AV) settings

November 30, 2017   DA Bureau   Interview/Opinion   0

Source: digianalysys.com, Feb. 2018

Self-Driving Cars Are Coming to Serve You

Auto companies are planning to launch autonomous vehicles into the service industry before selling them to consumers

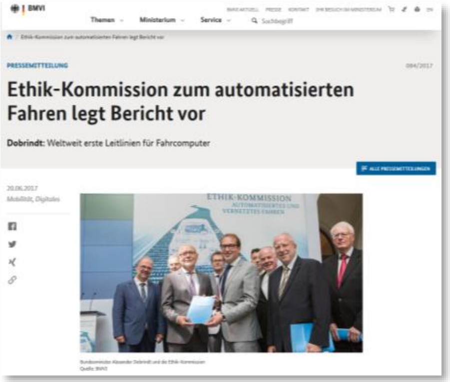
Source: consumerreports.org, Jan. 2018

Mercedes Robo-Taxi Coming Within A Few Months

Carscoops · Feb 6, 2018

Source: news.google.com, Feb. 2018

Motivation
… raising lots of questions, yet!



Source: bmvi.de, Feb. 2018

Source: news.google.com, Jan. 2018

Source: abcnews.com, Jun. 2017

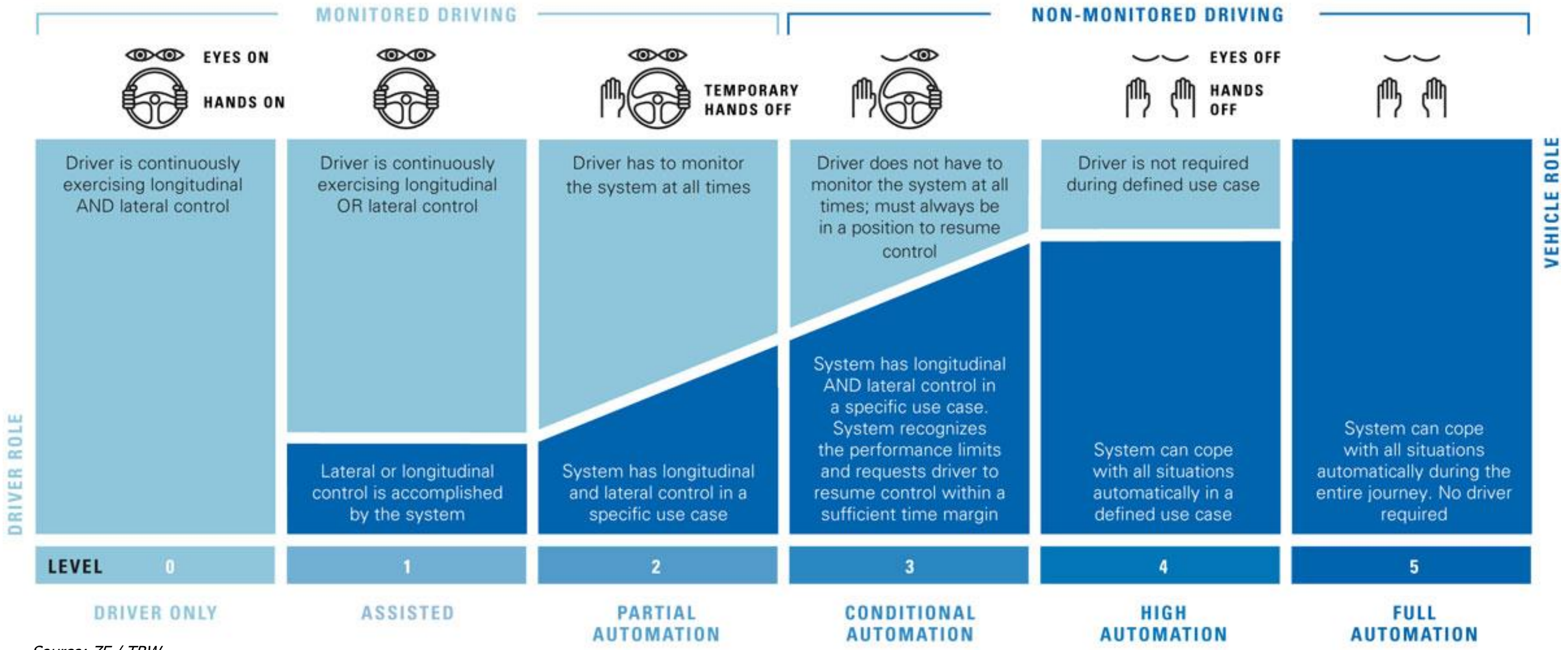
Autonomes Fahren: Crash eines Tesla Model S — 24.01.2018

Erneuter Tesla-Crash unter Autopilot

In Kalifornien hat ein Tesla Model S im Autopilot-Modus ein stehendes Feuerwehrauto gerammt. Der Fahrer und Tesla schieben einander die Schuld zu.

Source: autobild.de, Feb. 2018

Motivation
SAE Levels of Automation

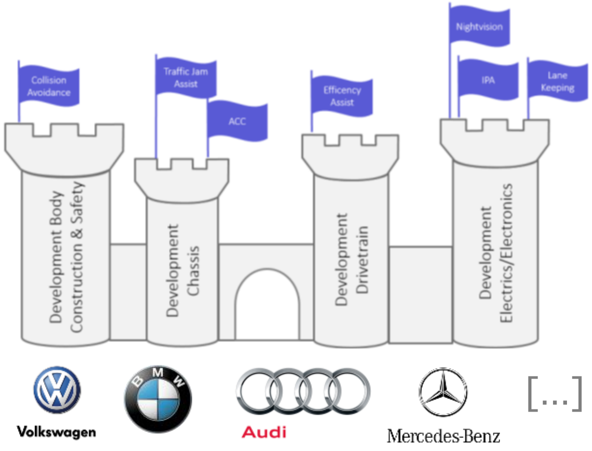


Source: ZF / TRW

Traditional development efforts – the fragmented approach

Complex functionality in very separated divisions

- OEMs need to monetarize every development step (ROI on development costs)
- Driver assistant systems are handled as every electrical/mechanical component



<u>Results:</u>

- Very slow development cycles for new products (>5 years)
- Development of ADAS is fragmented over several divisions
  - Very low reuse of existing software components
  - No general development of common software components
  - Many single-purpose systems developed outsourced at Tier-1s
  - Software companies (Google, Tesla Motors, …) are much faster with integrated approach
- OEMs are now starting to transform their development departments into fast sustainable software development entities

<u>Combination of ADAS functions is a huge challenge:</u>

- Complete testing of interaction between systems is not possible
- Interaction between separated systems can not be fully specified

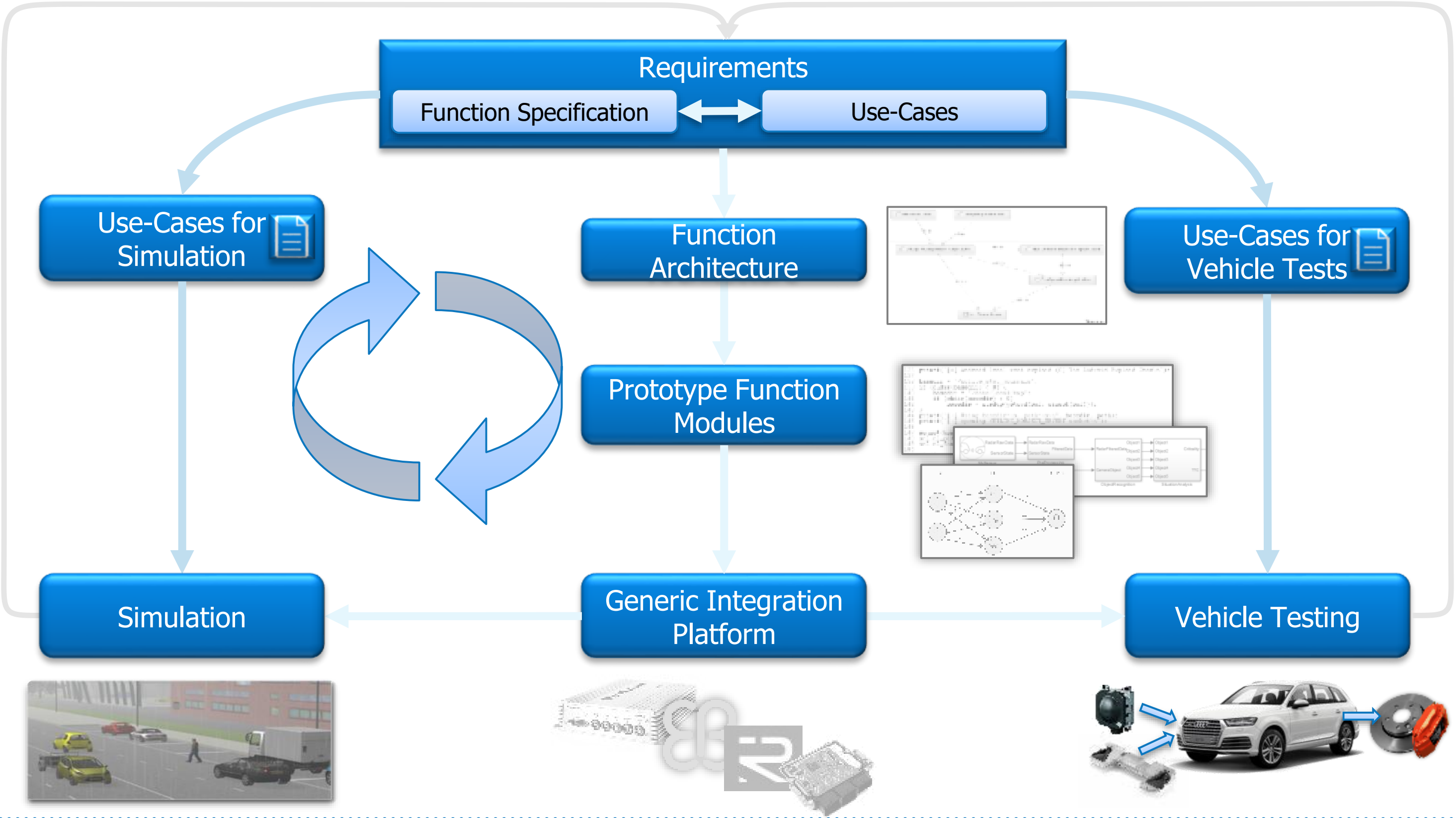
Example for the Complexity of a Function

"Advanced Emergency Braking" at Scania

1.5 million km driving - 80 terabytes of data

Source: J. Andersson: Model-Based Approach to Resource-Efficient Object Fusion for an Autonomous Braking System. In: Proc. MathWorks Automotive Conference 2015, Stuttgart.

Novel approach – the holistic approach
Iterative simulation based function development



Novel approach
Top-down function development

Define the functional goal

- make sure the description is complete, unambiguous, and consistent

Select or create scenarios that suit the functional goal

- address cases where something should happen, as well as cases where nothing shall happen
- enrich the scenarios with the ideal functional behaviour (providing the benchmark)

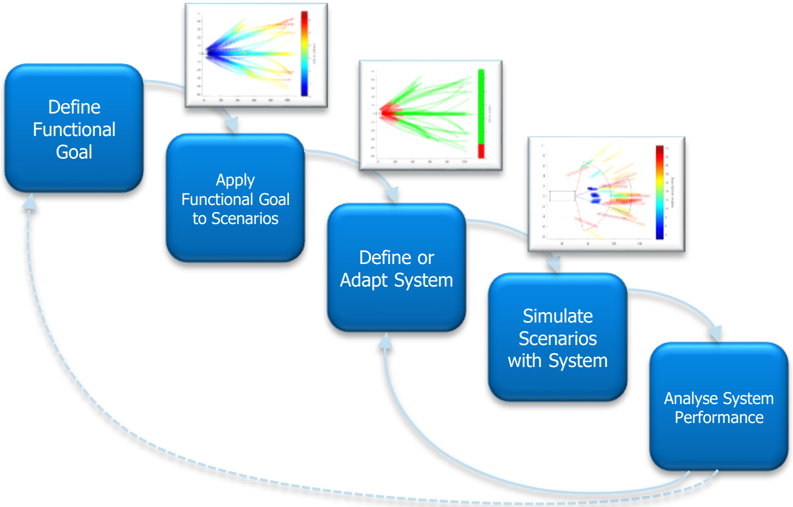
Define a system, including modules for sensor, algorithm, and actuator

- the modules can be ideal, modelled or real

Analyze the system performance on basis of the scenarios

- find out which module causes function performance loss
  → Improve that module if possible!

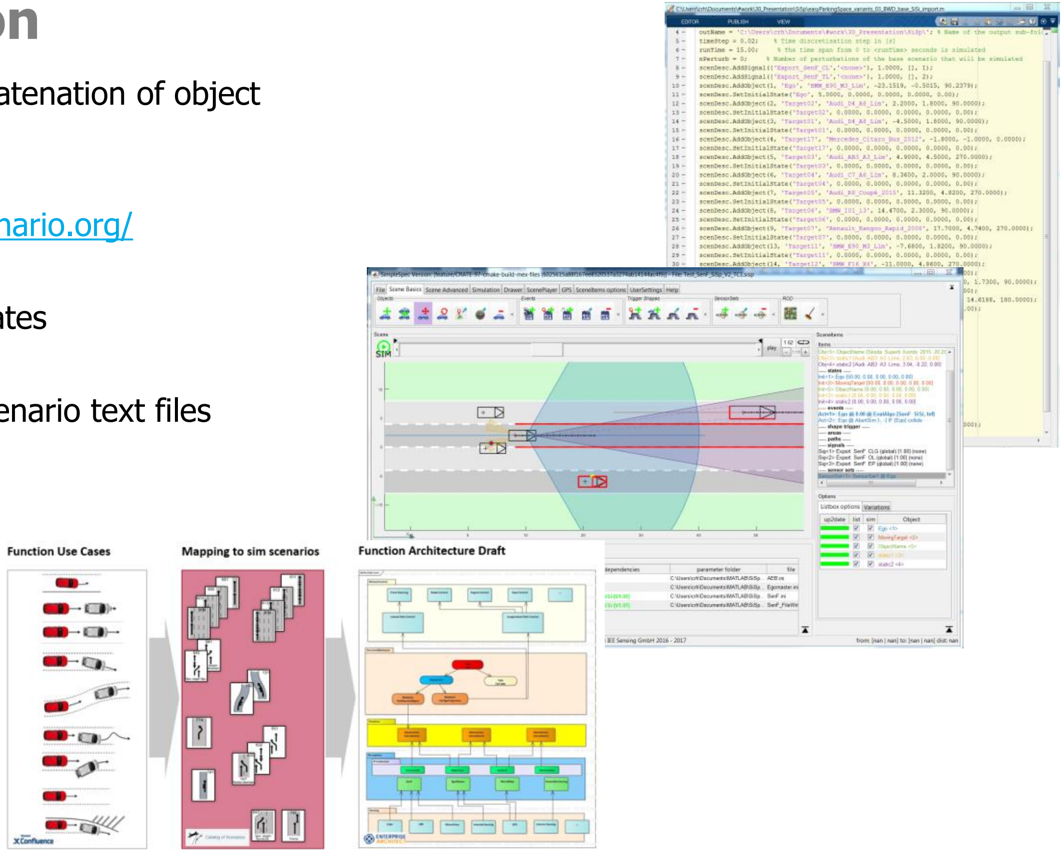
Know where to put your energy!
Loop back whenever and wherever necessary!



sensor	algorithm	actuator	performance
ideal	ideal	ideal	100%
model	ideal	ideal	90%
model	model	ideal	89%
…			
real	real	real	5%

Novel approach
Use-case based specification

- A “use-case” or “scenario” specification is a concatenation of object definition and behavior (maneuver based)
- Similar to the main ideas of http://www.openscenario.org/
- Definitions are done in global simulation coordinates
- Graphical definition of (still readable/editable) scenario text files
  - Set up environment and objects
  - Add events to objects, e.g. paths, triggers, …
  - Include sensor models and algorithms
  - Define stochastic variation parameters
  - Inspect simulation results, play as movie, …



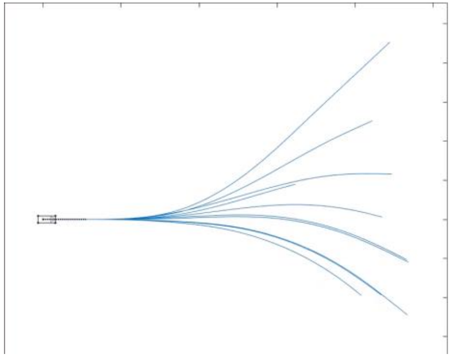
Novel approach
Simulation

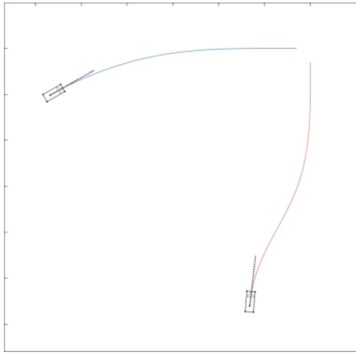
- Discrete event simulation backend for specified scenarios

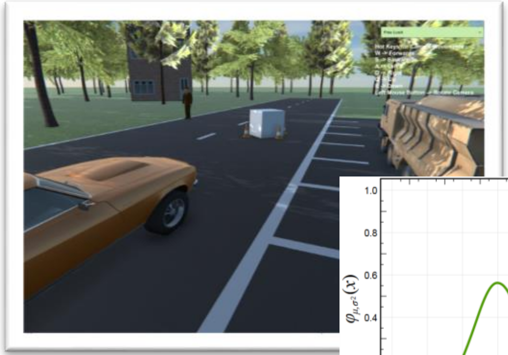
- Create the scenario's resulting signals
  - Object position
  - Velocities
  - …

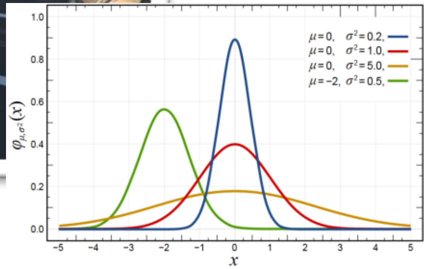
- Execute specific maneuvers (acceleration, braking, steering)
- Backward specification supported
- Follow a given path (list of points)
- Include additional algorithm(s) with closed-loop control feedback

- Variation of scenario parameters
  - Sampling from predefined stochastic distributions
  - Replication of stochastically independent scenarios
- 3D visualization of simulated scenarios







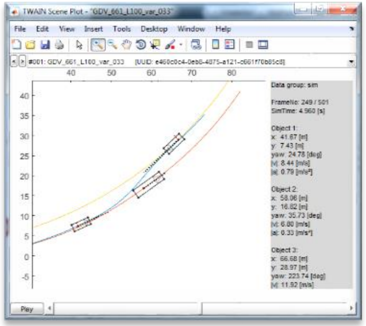


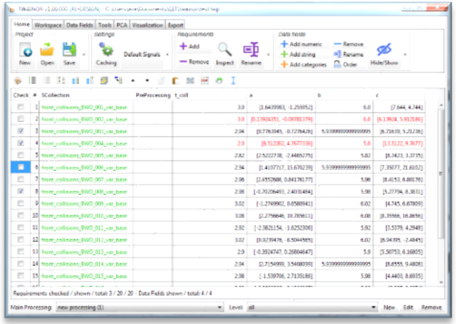
Novel approach
Data analysis

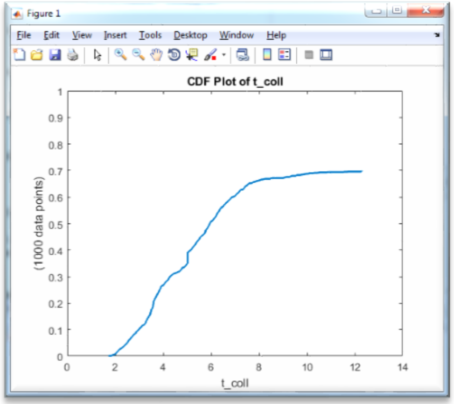
- Function development heavily depends on scenarios from
  - recorded real-world data
  - simulation data

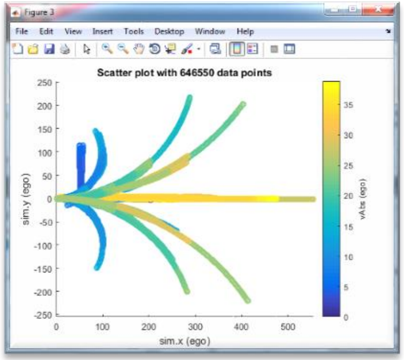
- Data analysis supports frequently used workflows
  - Add a simple sensor model based on simulation data
  - Calculate physical unavoidability of a collision scenario
  - Embed more/other function algorithms
  - Direct interface to closed-loop simulations

- Plot signals, animate scenario, create scatter plot
- Process signals (interpolation, filtering, …)
- Extract
  - time of collision from simulation data
  - signal values at a certain time
- Run tests and create test report









Novel approach
Machine learning

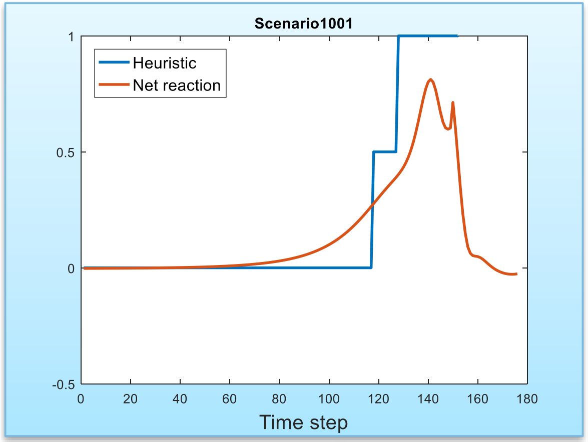
Goal:

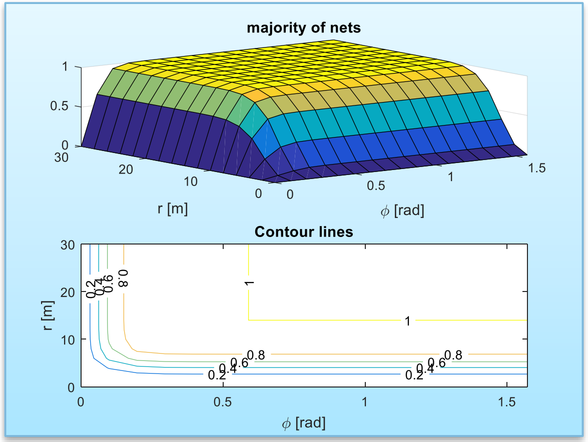
- Generate reference functionalities for model based solutions
- Use machine learning solutions as universal approximators

Two approaches:

- We know what the ideal behaviour should be and can formulate it mathematically.
  ➔ Train a machine learning reference

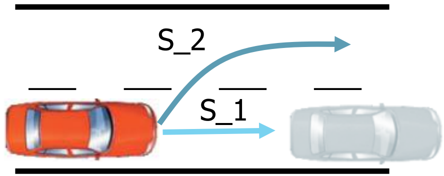
- We know which target we want to reach and what actions we can use to reach it.
  ➔ Learn a machine learning reference





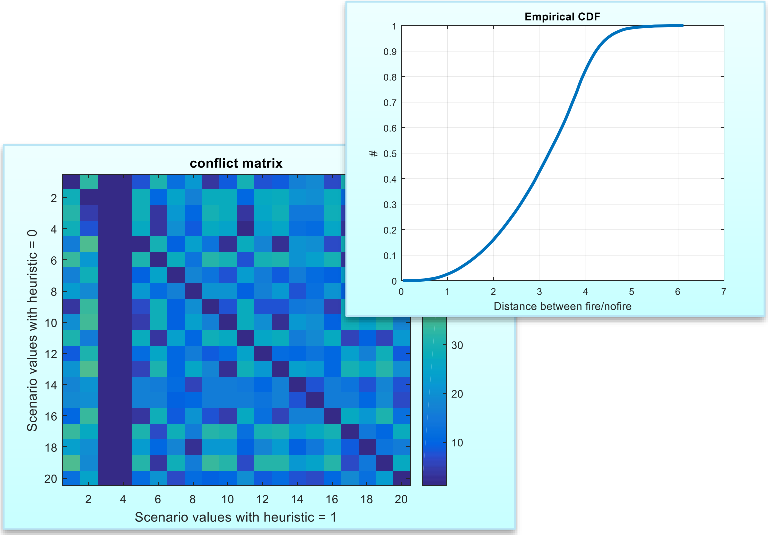
Novel approach
Machine learning

Example 1: Emergency braking function



Ideal solution: If a crash will happen one second into the future, we want to brake now (acausal formulation)

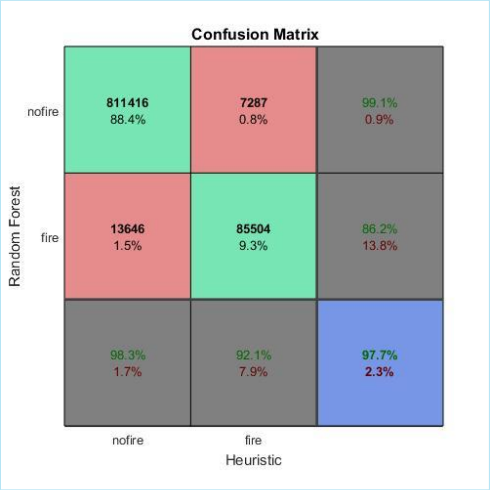
Problem: We don’t know what will happen in the future due to scenario similarities, leading to requirement conflicts.



Solution:

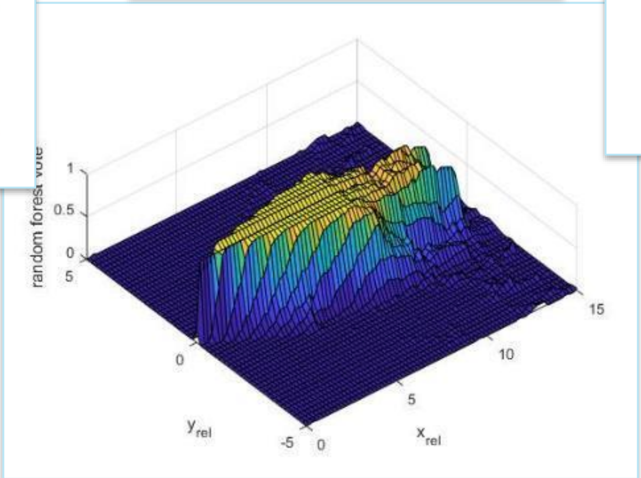
- Simulate the scenarios to generate reference data
- Analyse requirement conflicts on (raw) data level
- Train reference solutions (e. g. neural networks, random forests)
- Compare performance with model based solutions, e. g.
  - ROC-curve: trade off between field performance and misuse
  - Majority vote: Use statistics to judge stability of solutions

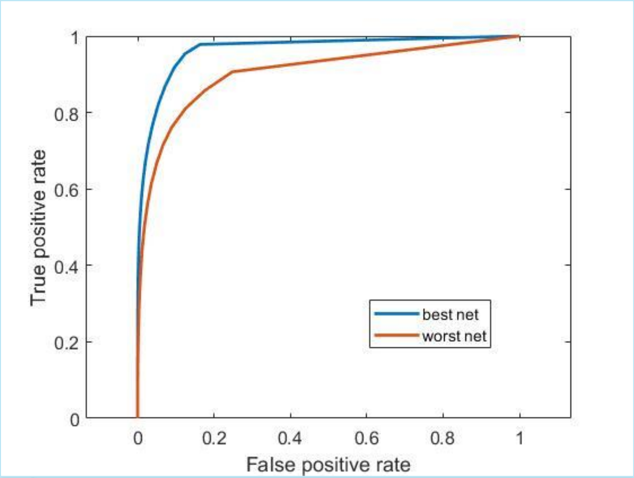
Novel approach
Machine learning



Performance estimation

Stability measure based on ensemble statistics





Trade-Off between Performance and Misuse

- FPR measures Misuse
- TPR measures Performance

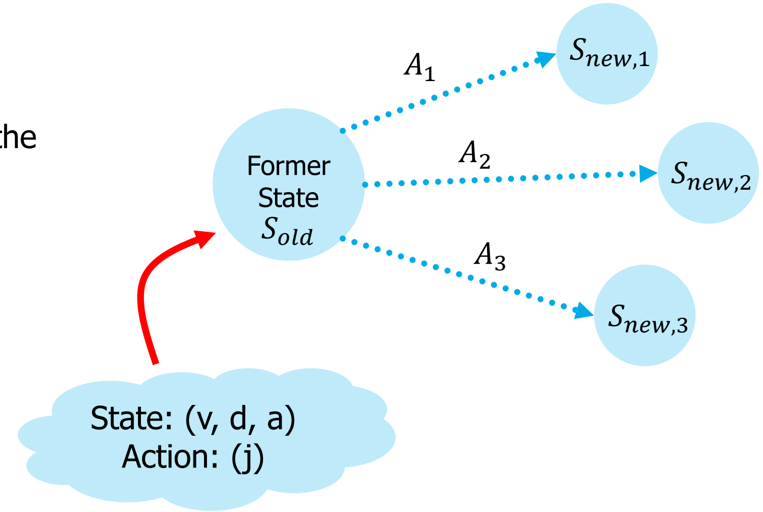
Comparison between different solutions possible

Novel approach
Machine learning

Example 2: Adaptive Cruise Control

Target: We want to reach a defined state without knowing the ideal strategy.

Solution:
- Use trial and error and a reward strategy to learn the optimal solution.
  ➔ Markov decision process, Q-learning
- Simulated annealing is used to balance exploration and exploitation.

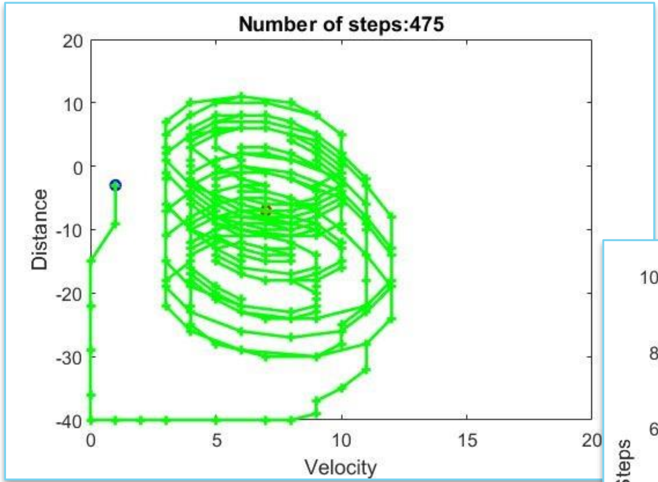


But: We have a very large state-action-space.

$$\underbrace{Q(S_{old}, A)}_{\text{Expected future reward}} \leftarrow Q(S_{old}, A) + \overset{\text{Learning rate}}{\lambda}\{ \underset{\text{Imminent reward}}{R} + \overset{\text{Discount factor}}{\Theta} \underbrace{\max Q(S_{new}, A')}_{\text{Max. expected future reward for following states}} - Q(S_{old}, A)\}$$

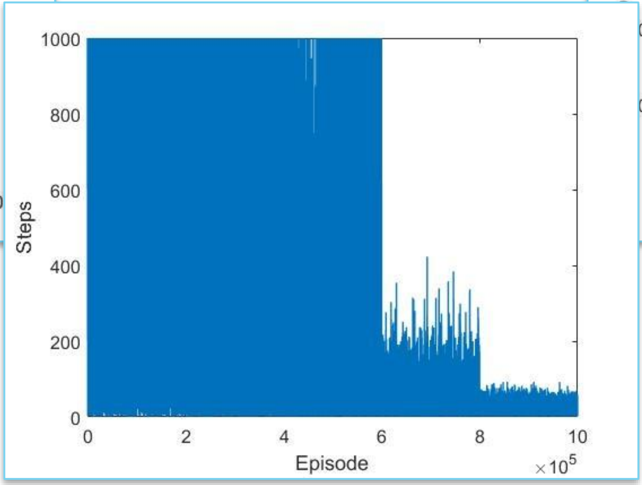
Solution:
- Deep Q-learning

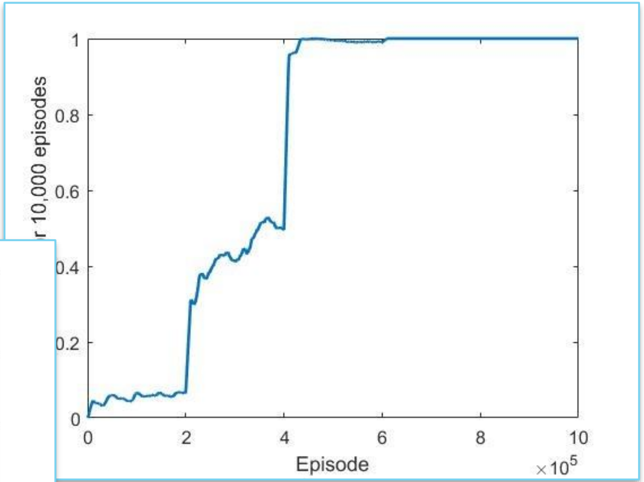
Novel approach
Machine learning



Training example

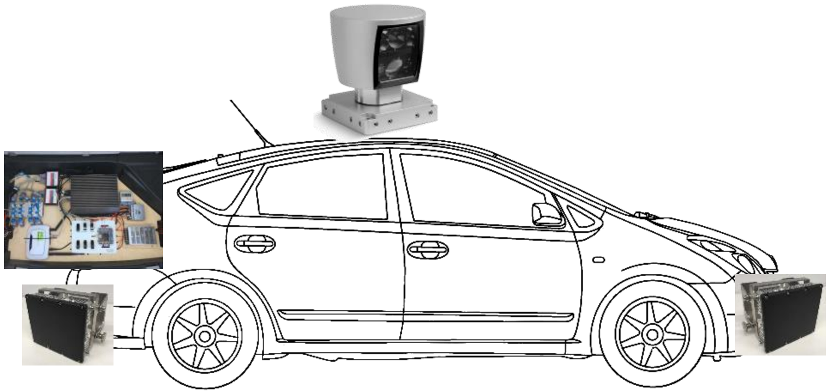
Training progress





Mean reward

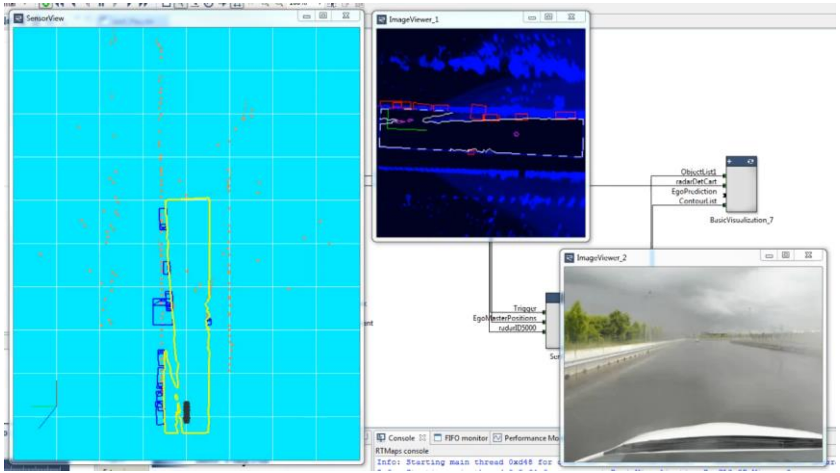
Novel approach
Real-life validation



Test vehicle equipped with

- Environment sensor systems (radar, lidar, vision, …)
- Car PC connected to bus system infrastructure
- Framework for integration of prototype function algorithms (RTMaps, ADTF, ROS, …)
- Positioning systems (GPS/DGPS)
- HDD data recorders

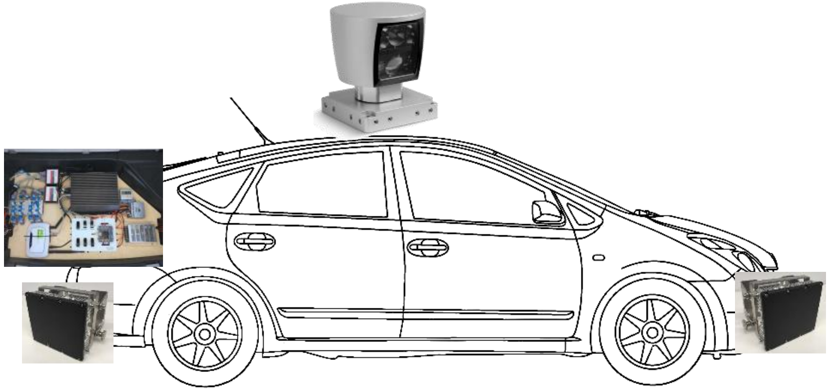
Set of defined test cases / driving manoeuvres

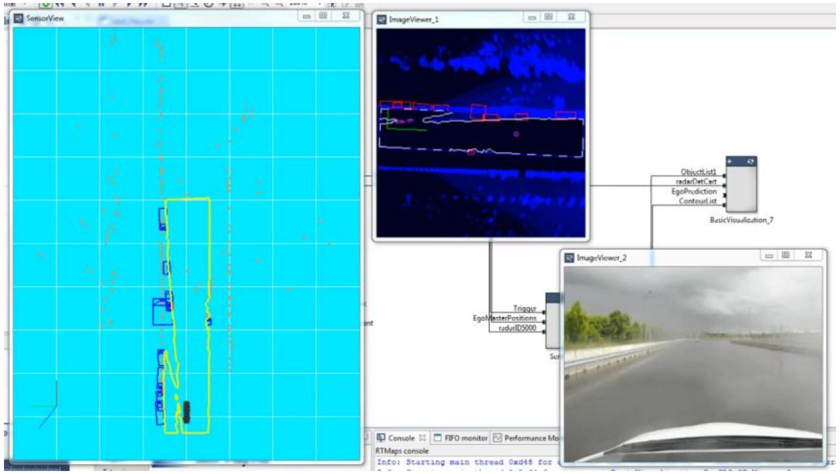


Novel approach
Real-life validation

Inevitable process step, yielding

- Validation of function prototypes
- Performance measures for sensor sets
- Recorded sensor and environment data for further processing
  - Input and training data for machine learning
  - Cross-validation of input data for simulation





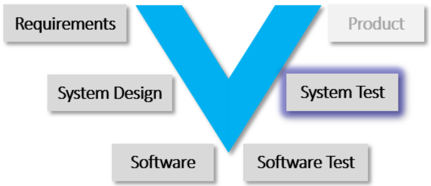
Virtualization techniques

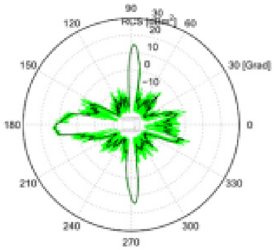
- Complex autonomous driving functions require sound validation and verification

- Real car testing:
  - late in development
  - expensive, time consuming
  - limited informational value (variations, reproducibility)

- Practicability?
  - Compliance to ISO26262 would require up to 5 bil. kilometers of test drives (assumption of 10^{-8}/h non-recovered failures[1])

  - But actually: This is a vanilla approach, we want to do it more intelligently.



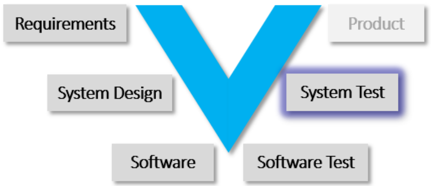


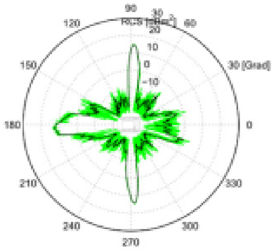
[1] *Winner, "Autonomes Fahren", 6. FAS-Tagung, München 2013*

Virtualization techniques

System Test 2.0:

- Early „offline" functional and ECU testing under real life conditions
- Demand for high-performant virtual test frameworks:
    - Comprehensive environment and perception models
    - Detailed scenario modeling capabilities (weather, surroundings, …)
    - Sophisticated sensor models (radar waves, laser beams, vision, ..)
    - Valid behavior models for traffic participants and partners in scenes
- Overall validation strategy taking into account all levels
    - MiL
    - SiL
    - HiL
    - ViL





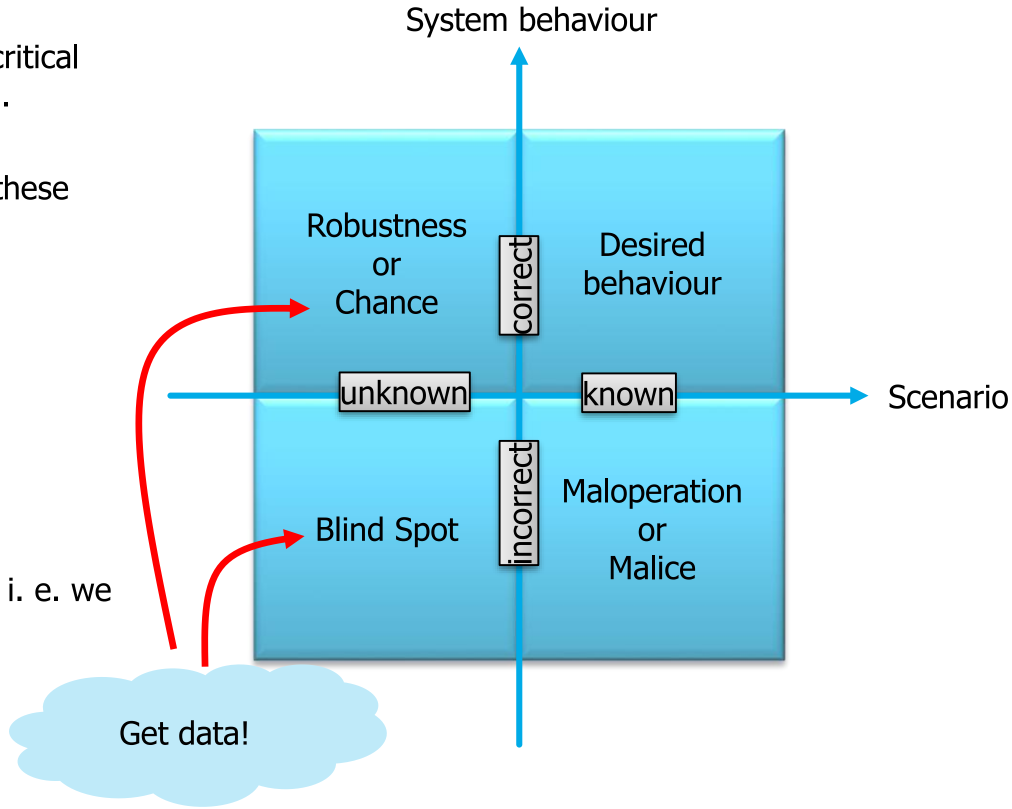
How to handle further uncertainties?

Despite our effort, we will not be able to address all critical scenarios that the system will encounter in real world.

We therefore have to assure that we can learn from these scenarios.

➔ Event data recorder with different trigger levels (critical scenarios, accident scenarios)
➔ data channel to acquire this data
➔ backend to process the data

We need an antifragile approach to handle this topic, i. e. we have to learn from our mistakes.



Conclusion and outlook

- Autonomous driving cars are one of the main disruptive topics for the future.
- Despite the hype and announcements, there is still a long way to go.
- Current OEMs have to change their processes dramatically to be able to cope with the looming challenges.
- We have shown you an alternative approach to reach our goal.
- You are invited to join the party.

Thank You!