This project has received funding from the European Union's Horizon 2020
research and innovation programme under grant agreement No 643921.

UnCoVer CPS

Unifying Control and Verification of Cyber-Physical Systems

Project Summary

Matthias Althoff, Technische Universität München

ARTEMIS Spring Event, 14 April 2016

Partners

Unifying Control and Verification of Cyber-Physical Systems
(UnCoVerCPS)

Funding: 4.9 mio Euro

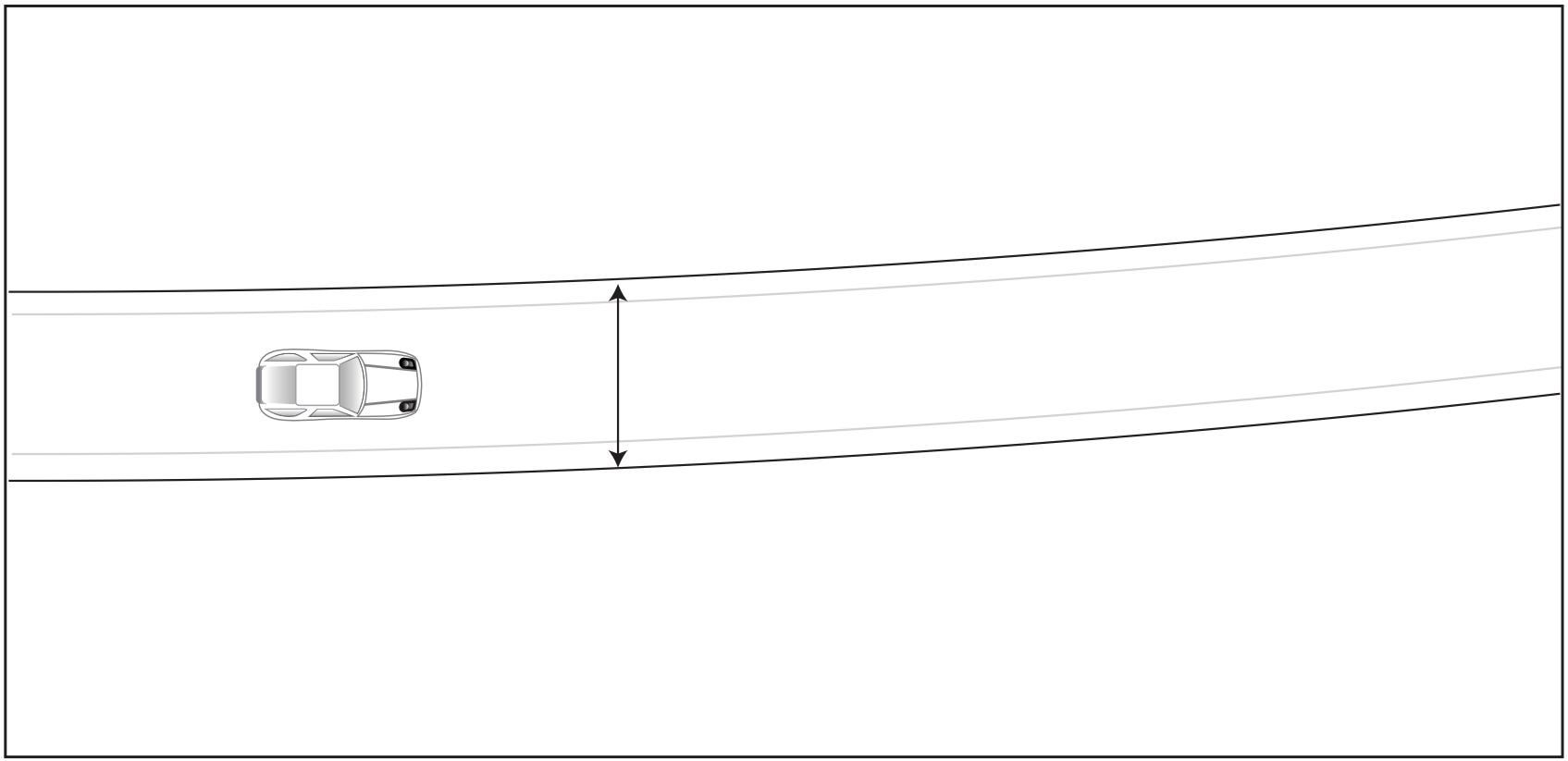
Participant organisation name	**Country**
Technische Universität München (TUM)	Germany
Université Joseph Fourier Grenoble 1 (UJF)	France
Universität Kassel (UKS)	Germany
Politecnico di Milano (PoliMi)	Italy
GE Global Research Europe (GE)	Germany
Robert Bosch GmbH (Bosch)	Germany
Esterel Technologies (ET)	France
Deutsches Zentrum für Luft- und Raumfahrt (DLR)	Germany
Tecnalia (Tec)	Spain
R.U.Robots Limited (RUR)	United Kingdom

Have We Considering All Situations?

The need for online verification can be motivated for any of our use cases. Automated driving is used in this presentation.

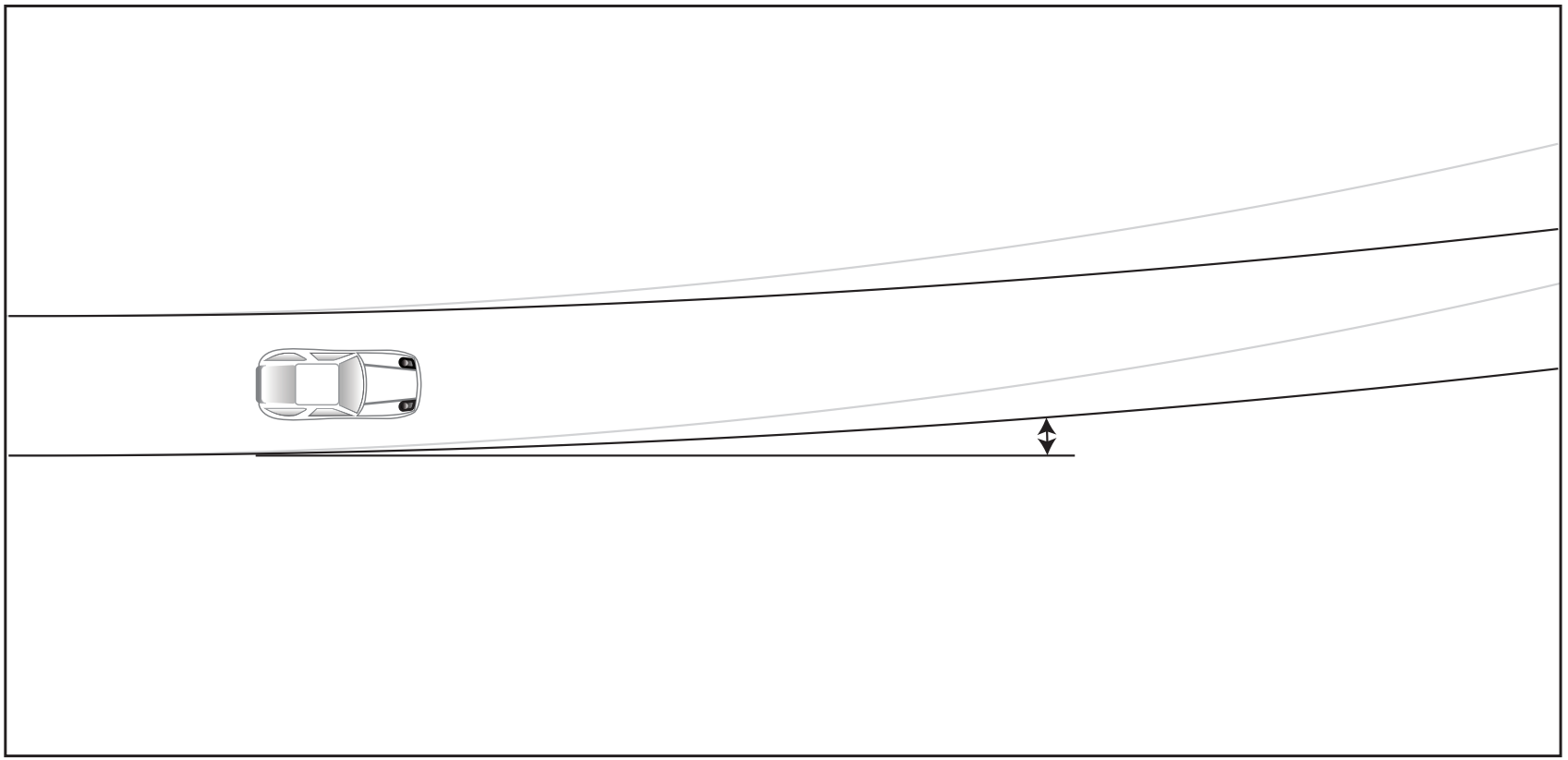



Possible Traffic Situations (1): Varying Lane Width



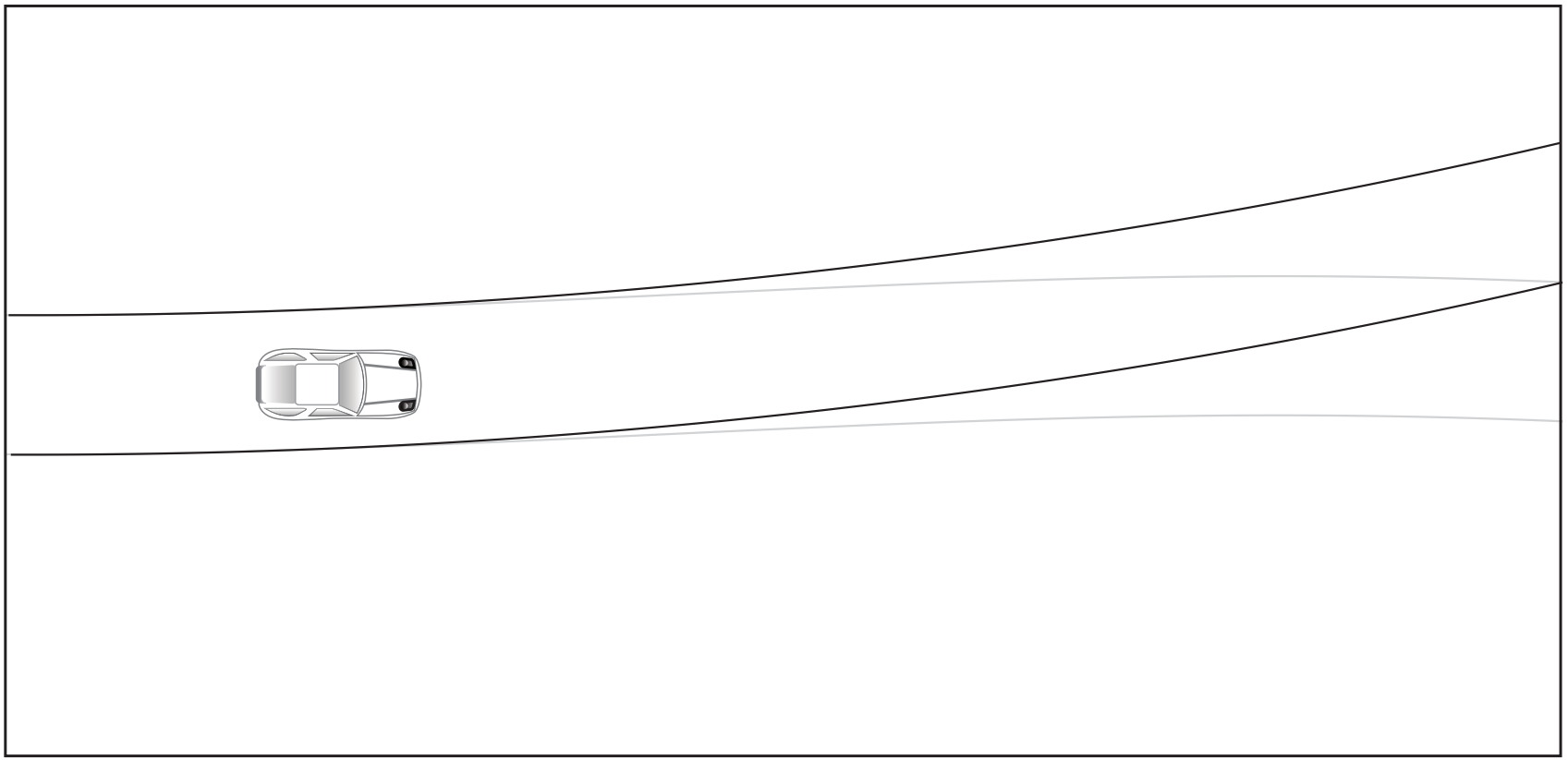
We assume that each variable has 20 possible values.
Number of scenarios: 20.

Possible Traffic Situations (2): Varying Lane Curvature



We assume that each variable has 20 possible values.
Number of scenarios: $20^2 = 400$.

Possible Traffic Situations (3): Change of Curvature

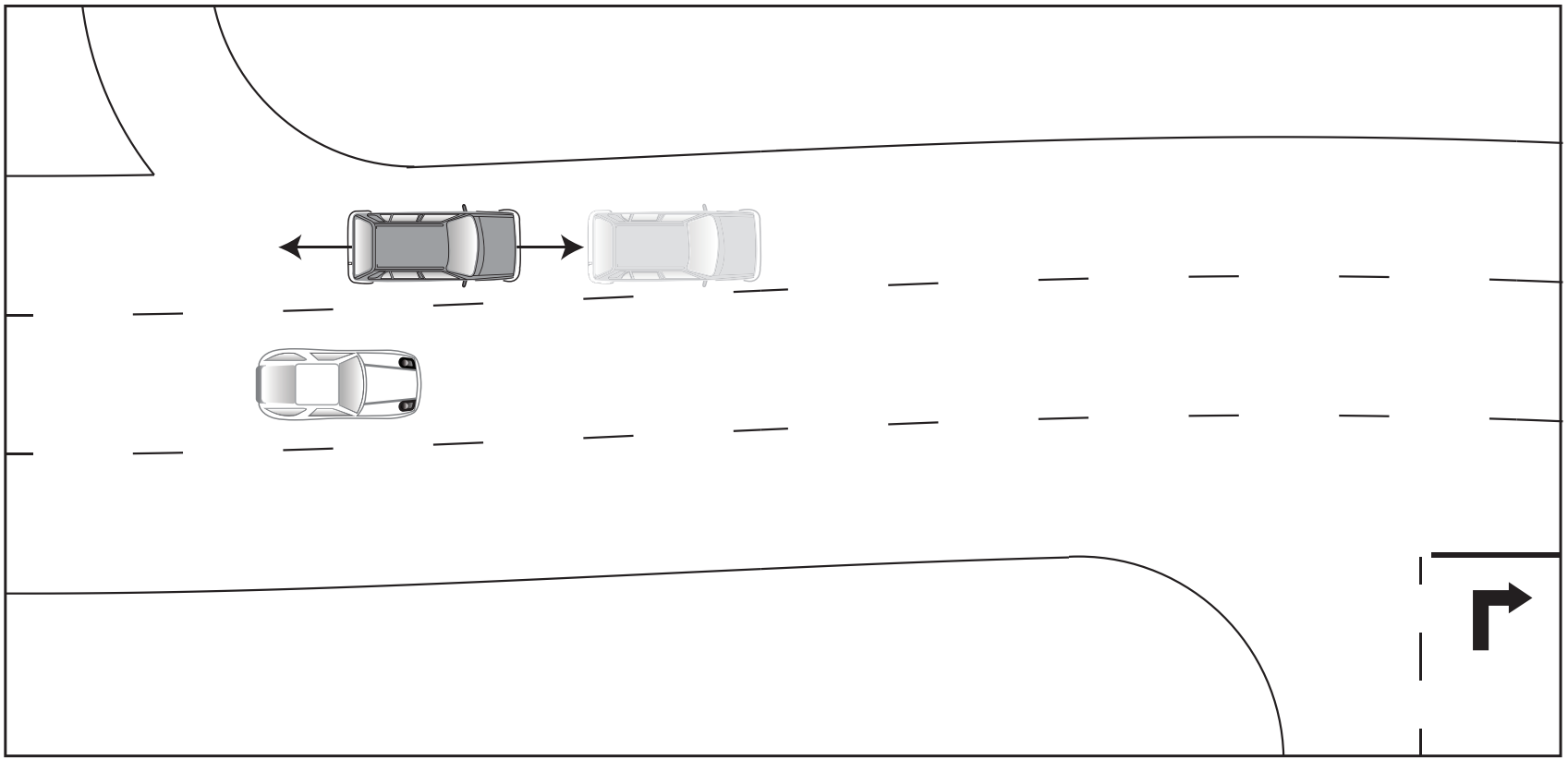


We assume that each variable has 20 possible values.
Number of scenarios: $20^3 = 8000$.

Possible Traffic Situations (4): Varying Number of Lanes

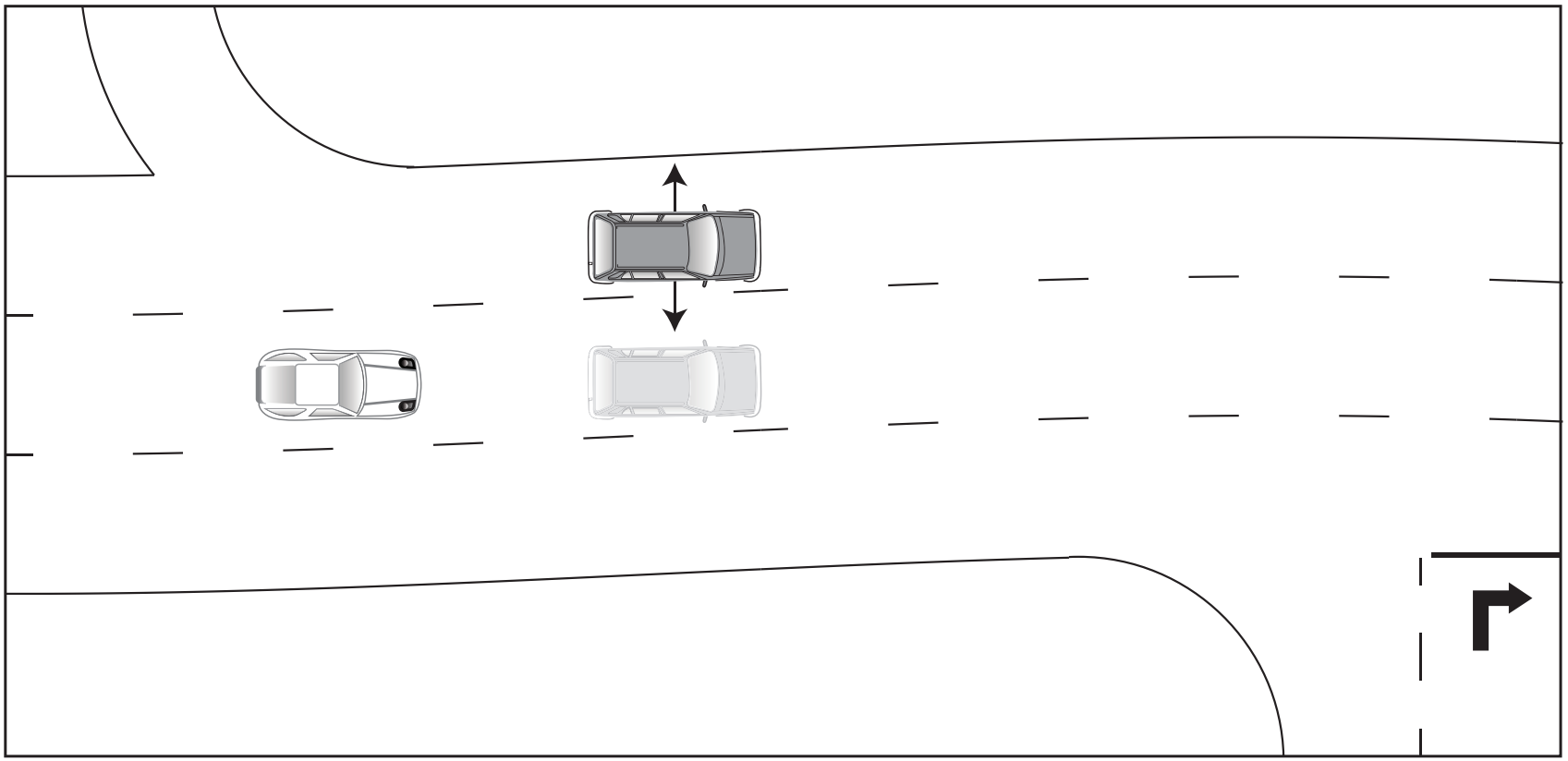
We assume that each variable has 20 possible values.
Number of scenarios: $(20^3)^5 = 3.3 \cdot 10^{19}$ (assumption: max. 5 lanes).

Possible Traffic Situations (5): Varying x-Position



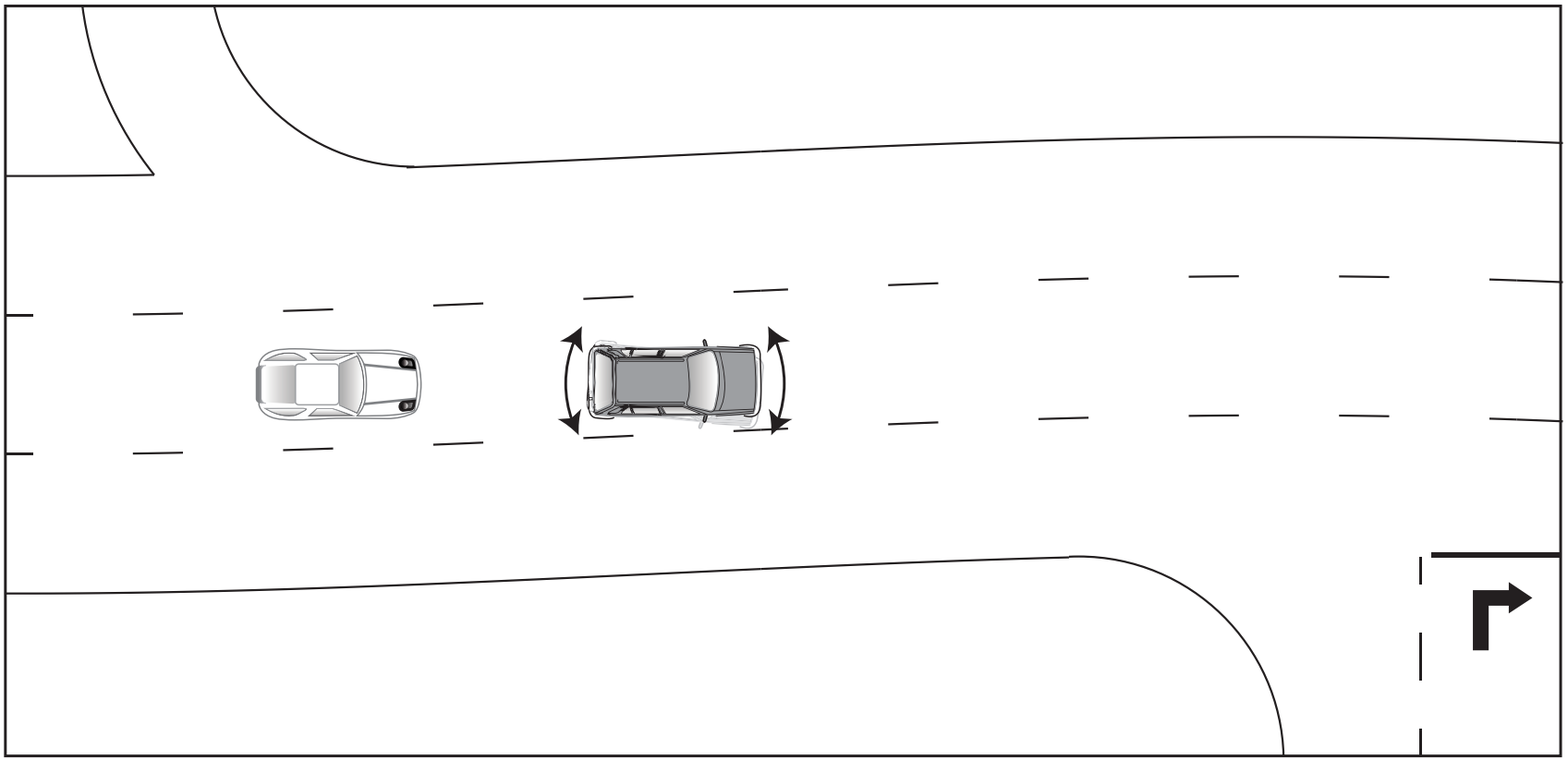
We assume that each variable has 20 possible values.
Number of scenarios: $(20^3)^5 \cdot 20 = 6.6 \cdot 10^{20}$.

Possible Traffic Situations (6): Varying y-Position



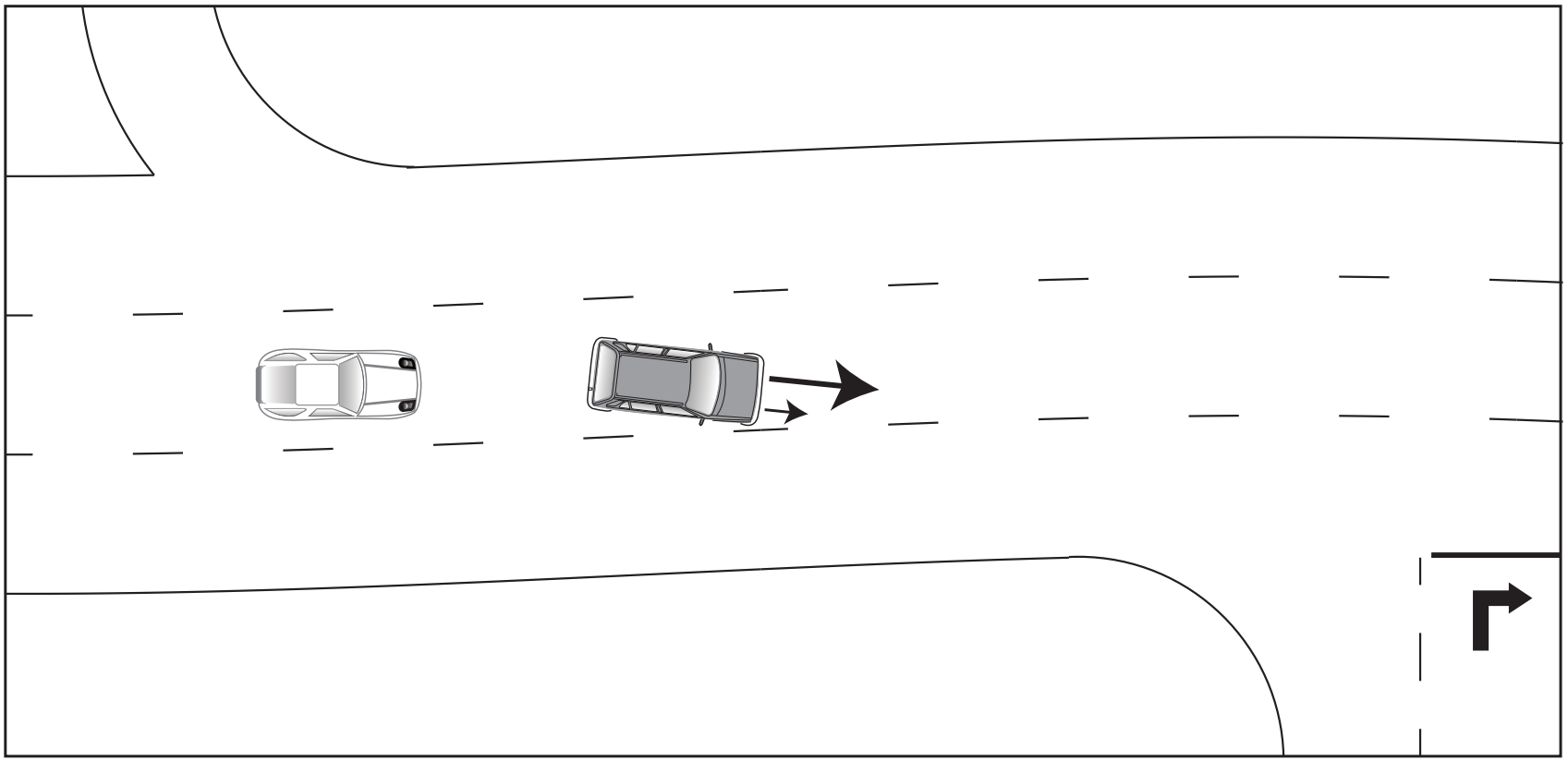
We assume that each variable has 20 possible values.
Number of scenarios: $(20^3)^5 \cdot 20^2 = 1.3 \cdot 10^{22}$.

Possible Traffic Situations (7): Varying Orientation



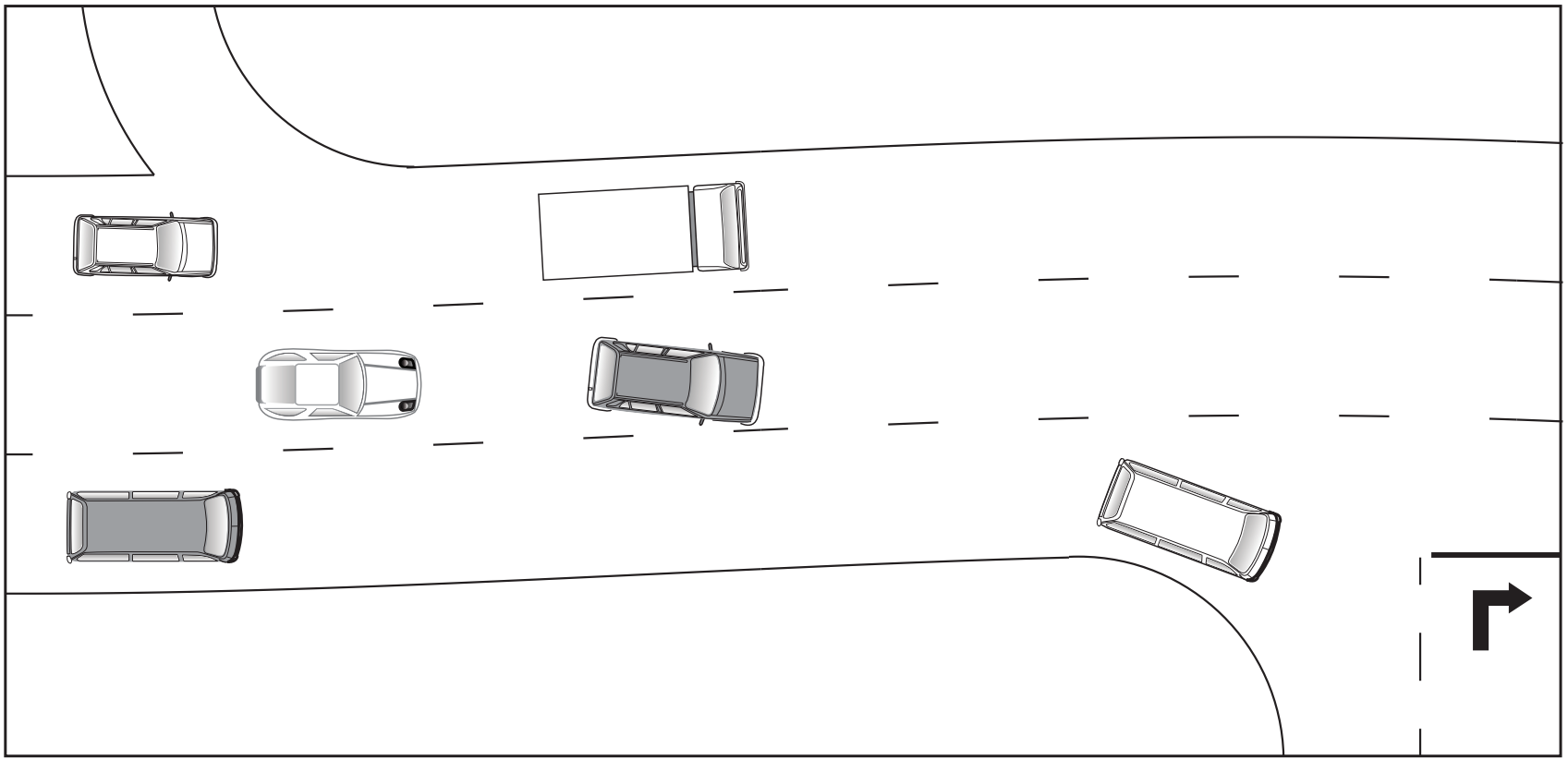
We assume that each variable has 20 possible values.
Number of scenarios: $(20^3)^5 \cdot 20^3 = 2.6 \cdot 10^{23}$.

Possible Traffic Situations (8): Varying Velocity



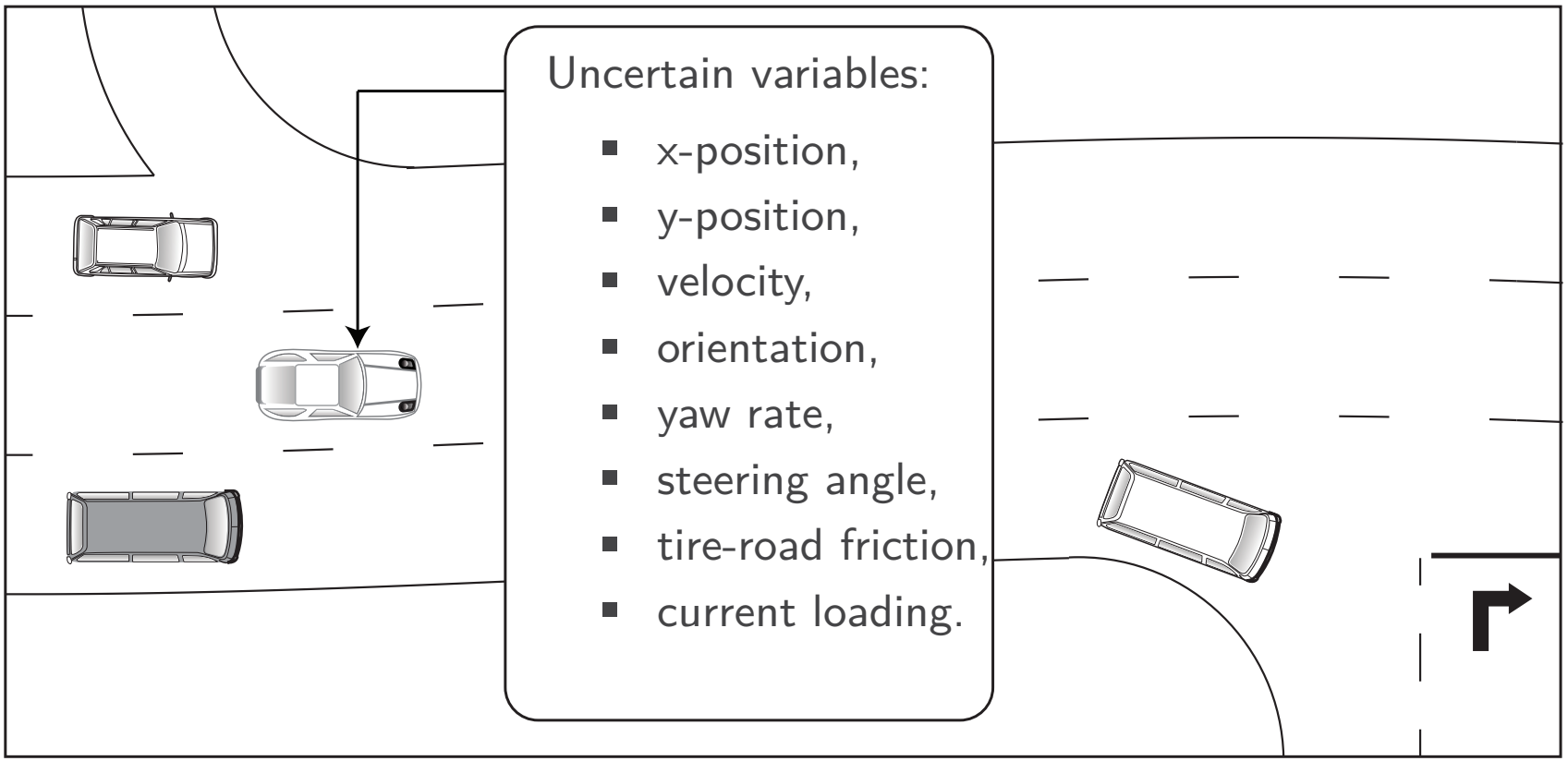
We assume that each variable has 20 possible values.
Number of scenarios: $(20^3)^5 \cdot 20^4 = 5.2 \cdot 10^{24}$.

Possible Traffic Situations (9): Varying Number of Vehicles



We assume that each variable has 20 possible values.
Number of scenarios: $(20^3)^5 \cdot (20^4)^{10} = 3.6 \cdot 10^{71}$ (max. 10 vehicles).

Possible Traffic Situations (10): Variables of Ego Vehicle



We assume that each variable has 20 possible values.
Number of scenarios: $(20^3)^5 \cdot (20^4)^{10} \cdot 20^8 = 9.2 \cdot 10^{81}$ (Atoms in the universe $\approx 10^{80}$). In reality, the number of situations is uncountable.

Online vs Offline Verification

Offline Verification

Verify a system based on a mathematical model before deployment.

Online vs Offline Verification

Offline Verification

Verify a system based on a mathematical model before deployment.

- ⊕ No real-time guarantees required.

Online vs Offline Verification

Offline Verification

Verify a system based on a mathematical model before deployment.

- ⊕ No real-time guarantees required.
- ⊕ Detected bugs can be repaired before deployment.

Online vs Offline Verification

Offline Verification

Verify a system based on a mathematical model before deployment.

- ⊕ No real-time guarantees required.
- ⊕ Detected bugs can be repaired before deployment.
- ⊖ All possible changes in the environment need to be verified upfront.

Online vs Offline Verification

Offline Verification

Verify a system based on a mathematical model before deployment.

- ⊕ No real-time guarantees required.
- ⊕ Detected bugs can be repaired before deployment.
- ⊖ All possible changes in the environment need to be verified upfront.

Online Verification

Verify a given system during its operation.

Online vs Offline Verification

Offline Verification

Verify a system based on a mathematical model before deployment.

- ⊕ No real-time guarantees required.
- ⊕ Detected bugs can be repaired before deployment.
- ⊖ All possible changes in the environment need to be verified upfront.

Online Verification

Verify a given system during its operation.

- ⊖ Real-time guarantees required.

Online vs Offline Verification

Offline Verification

Verify a system based on a mathematical model before deployment.

- ⊕ No real-time guarantees required.
- ⊕ Detected bugs can be repaired before deployment.
- ⊖ All possible changes in the environment need to be verified upfront.

Online Verification

Verify a given system during its operation.

- ⊖ Real-time guarantees required.
- ⊖ Detected bugs have to be dealt with during deployment; Safe backup strategy required (fail-safe).

Online vs Offline Verification

Offline Verification

Verify a system based on a mathematical model before deployment.

- ⊕ No real-time guarantees required.
- ⊕ Detected bugs can be repaired before deployment.
- ⊖ All possible changes in the environment need to be verified upfront.

Online Verification

Verify a given system during its operation.

- ⊖ Real-time guarantees required.
- ⊖ Detected bugs have to be dealt with during deployment; Safe backup strategy required (fail-safe).
- ⊕ All possible changes in the environment are automatically considered.

Online vs Offline Verification

Offline Verification

Verify a system based on a mathematical model before deployment.

- ⊕ No real-time guarantees required.
- ⊕ Detected bugs can be repaired before deployment.
- ⊖ All possible changes in the environment need to be verified upfront.

Online Verification

Verify a given system during its operation.

- ⊖ Real-time guarantees required.
- ⊖ Detected bugs have to be dealt with during deployment; Safe backup strategy required (fail-safe).
- ⊕ All possible changes in the environment are automatically considered.

In automated driving, online verification will be required!

Main objectives

- Novel on-the-fly control and verification concepts.

Main objectives

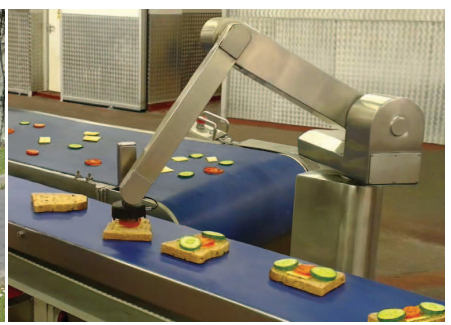
- Novel on-the-fly control and verification concepts.
- Unifying control and verification to quickly react to changing environments.

Main objectives

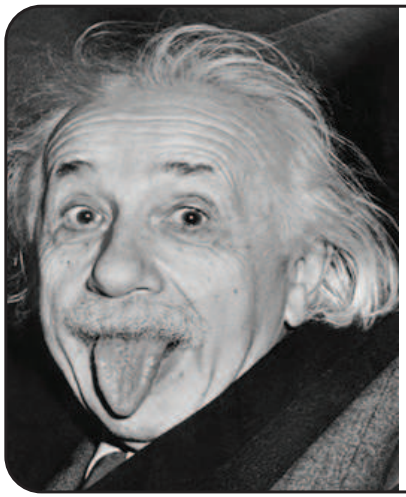
- Novel on-the-fly control and verification concepts.
- Unifying control and verification to quickly react to changing environments.
- A unique tool chain that makes it possible to integrate modeling, control design, formal verification, and automatic code generation.

Main objectives

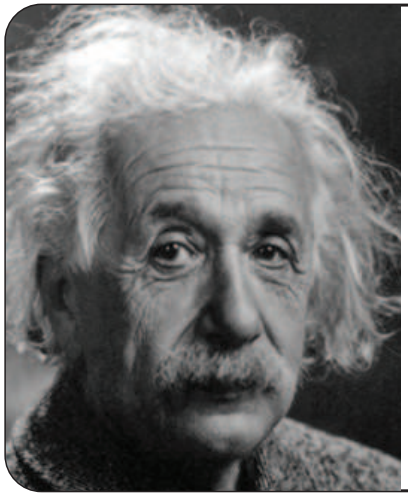
- Novel on-the-fly control and verification concepts.
- Unifying control and verification to quickly react to changing environments.
- A unique tool chain that makes it possible to integrate modeling, control design, formal verification, and automatic code generation.
- Prototypical realizations for automated vehicles, human-robot collaborative manufacturing, wind turbines and smart grids.

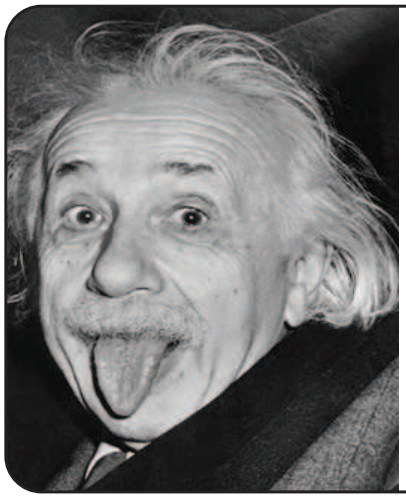
Safety Net



Playful development of artificial intelligence:

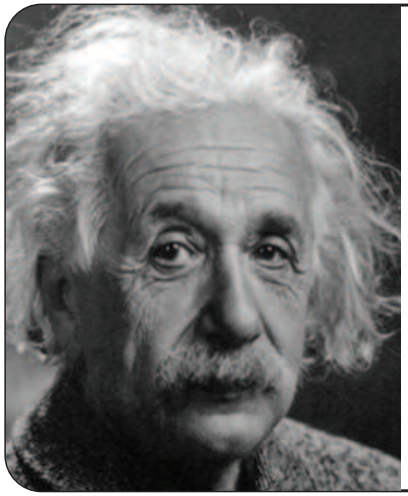


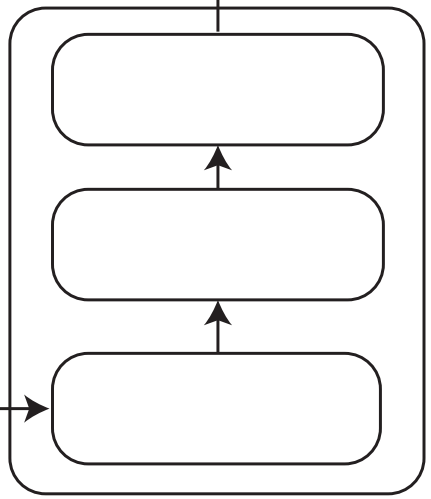
Safety Net



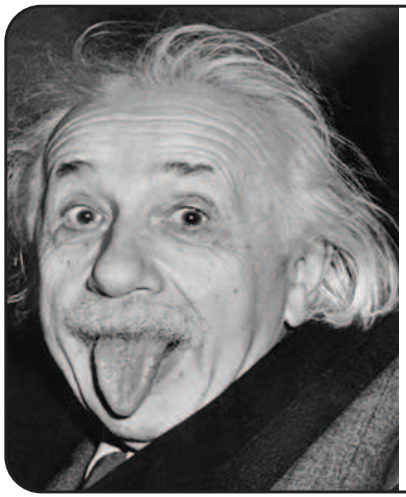
Playful development of artificial intelligence:

- Behavior may depend on past experience of vehicle, surprising behavior possible.



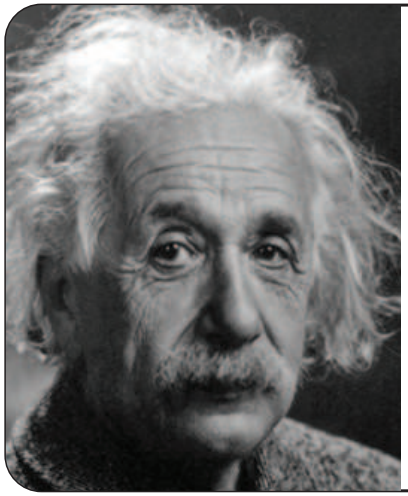


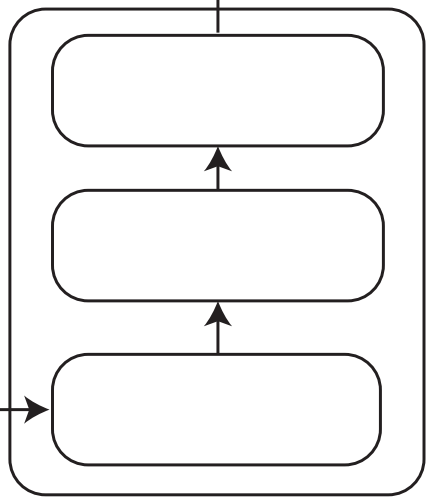
Safety Net



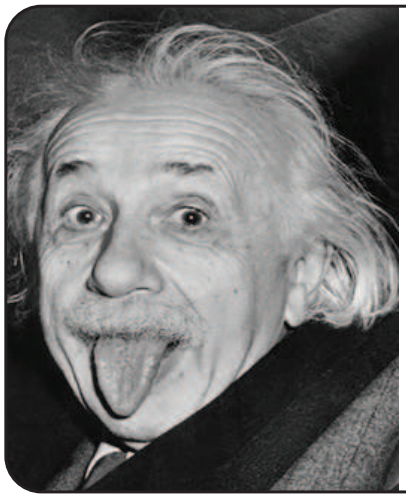
Playful development of artificial intelligence:

- Behavior may depend on past experience of vehicle, surprising behavior possible.
- Large and unpredictable software.



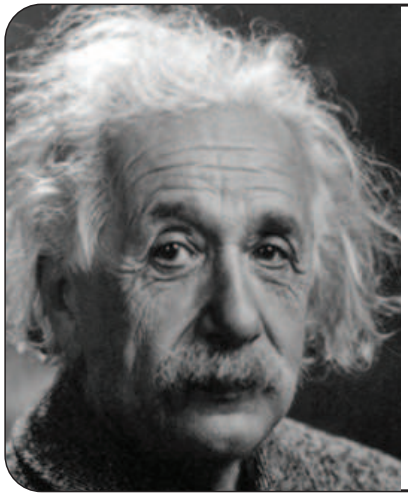


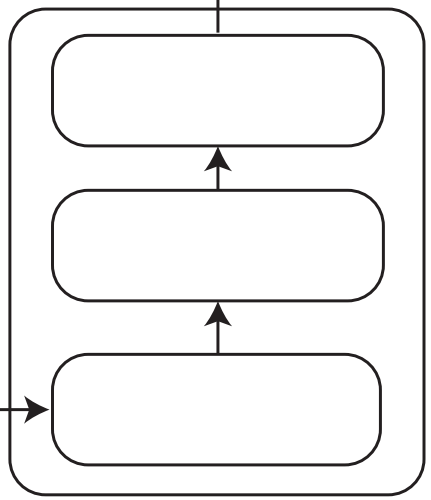
Safety Net



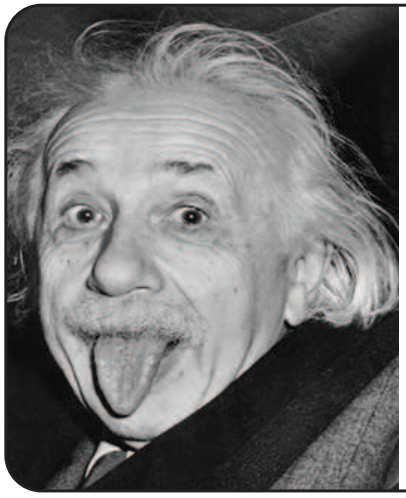
Playful development of artificial intelligence:

- Behavior may depend on past experience of vehicle, surprising behavior possible.
- Large and unpredictable software.
- Behavior depends on past experience.



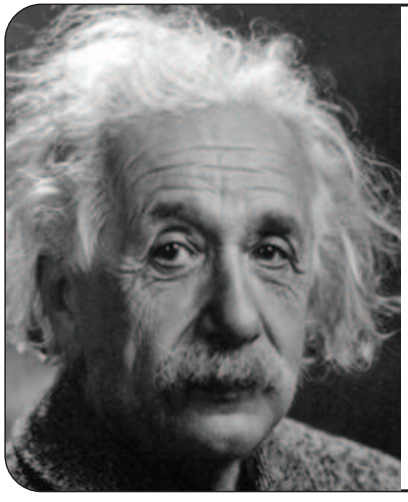


Safety Net

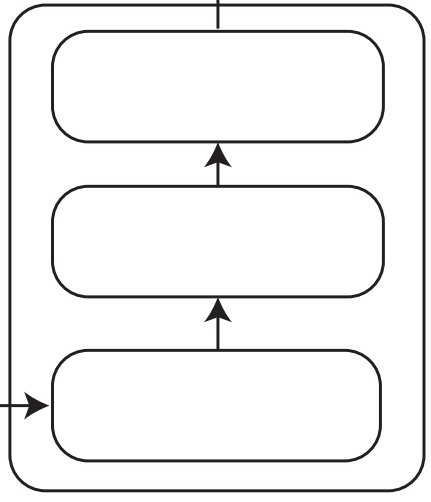


Playful development of artificial intelligence:

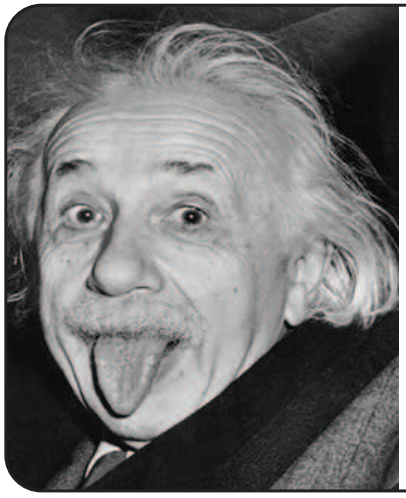
- Behavior may depend on past experience of vehicle, surprising behavior possible.
- Large and unpredictable software.
- Behavior depends on past experience.



Serious development of safe controllers:

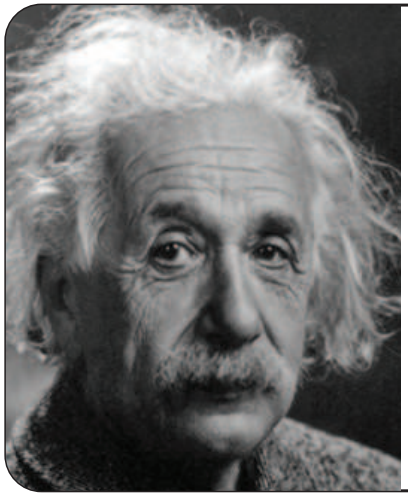


Safety Net



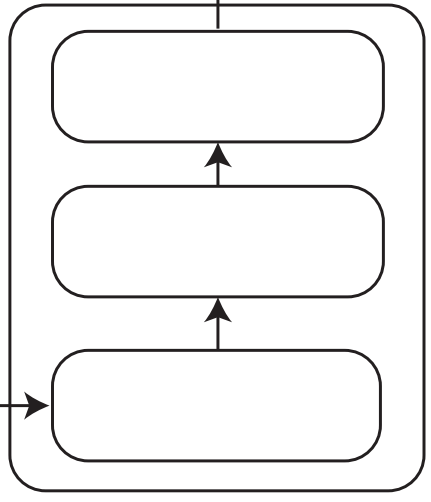
Playful development of artificial intelligence:

- Behavior may depend on past experience of vehicle, surprising behavior possible.
- Large and unpredictable software.
- Behavior depends on past experience.

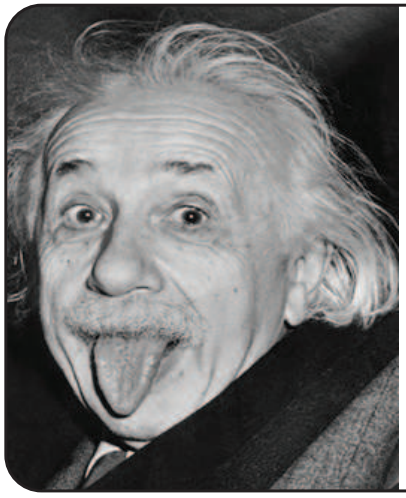


Serious development of safe controllers:

- Only emergency maneuvers.

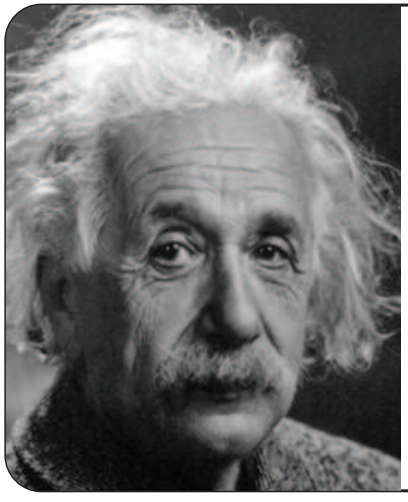


Safety Net



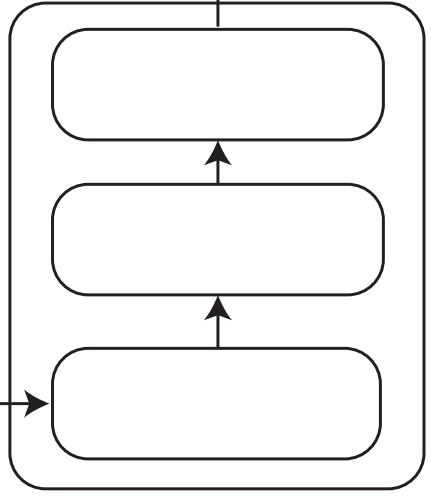
Playful development of artificial intelligence:

- Behavior may depend on past experience of vehicle, surprising behavior possible.
- Large and unpredictable software.
- Behavior depends on past experience.

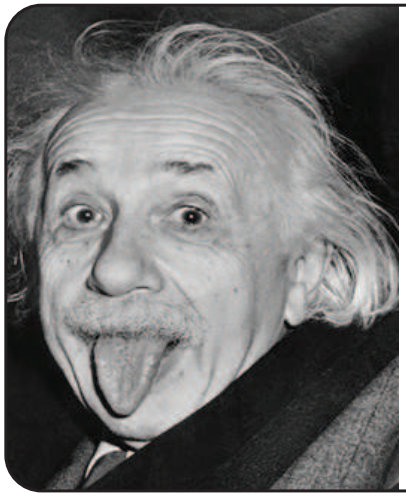


Serious development of safe controllers:

- Only emergency maneuvers.
- Formal methods guarantee correctness.

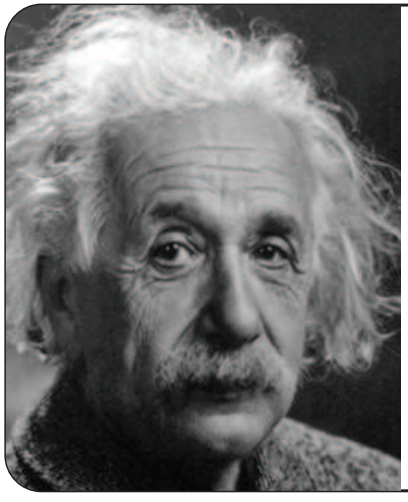


Safety Net



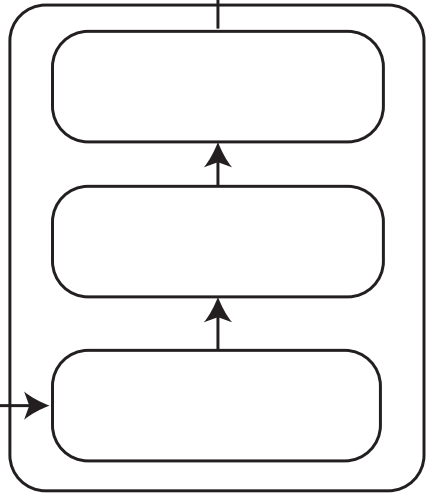
Playful development of artificial intelligence:

- Behavior may depend on past experience of vehicle, surprising behavior possible.
- Large and unpredictable software.
- Behavior depends on past experience.

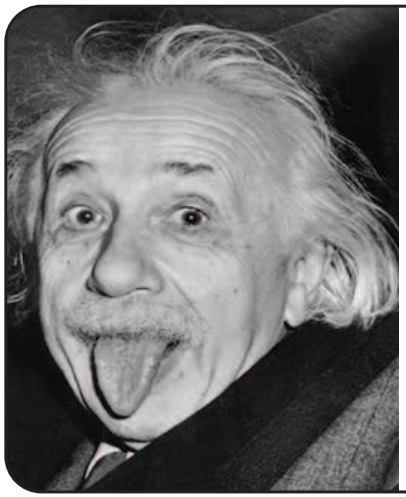


Serious development of safe controllers:

- Only emergency maneuvers.
- Formal methods guarantee correctness.
- Deterministic behavior, no surprising behavior.

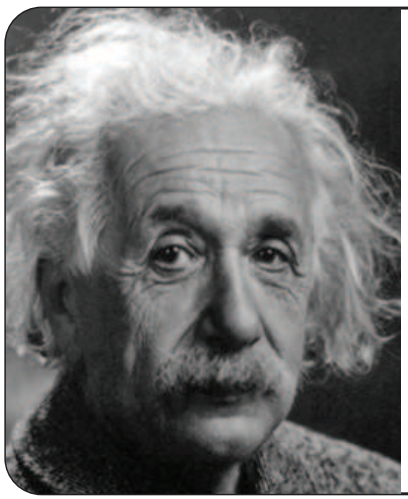


Safety Net



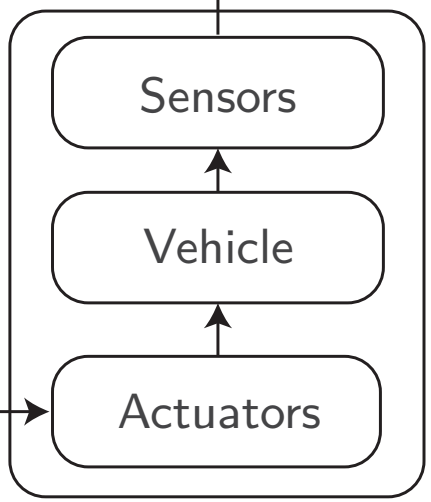
Playful development of artificial intelligence:

- Behavior may depend on past experience of vehicle, surprising behavior possible.
- Large and unpredictable software.
- Behavior depends on past experience.

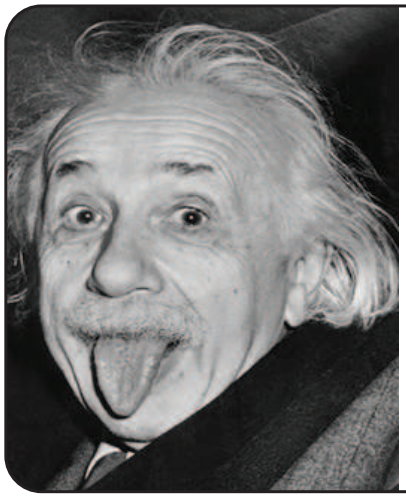


Serious development of safe controllers:

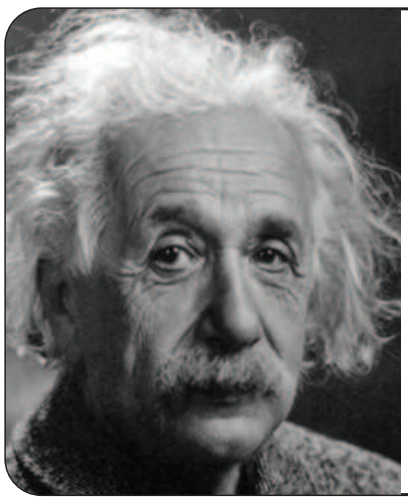
- Only emergency maneuvers.
- Formal methods guarantee correctness.
- Deterministic behavior, no surprising behavior.



Dual Use: Human Assistance

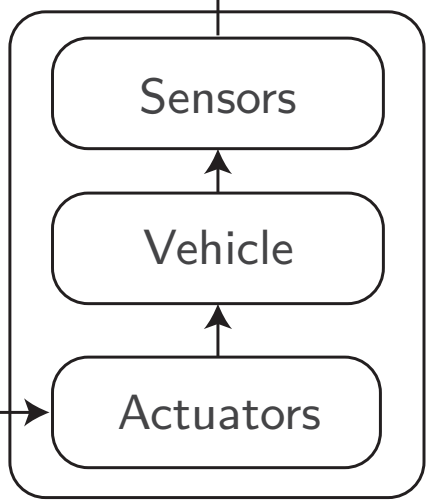


Human operator/driver



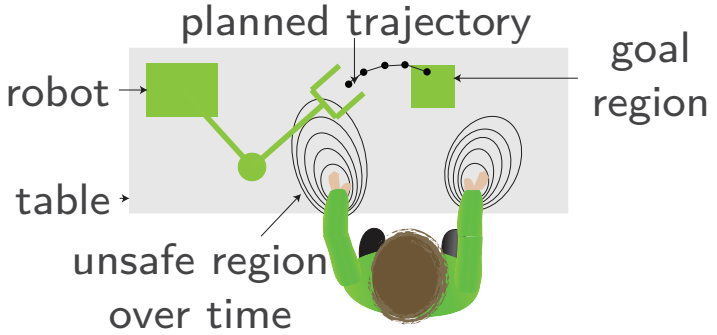
Serious development of safe controllers:

- Only emergency maneuvers.
- Formal methods guarantee correctness.
- Deterministic behavior, no surprising behavior.

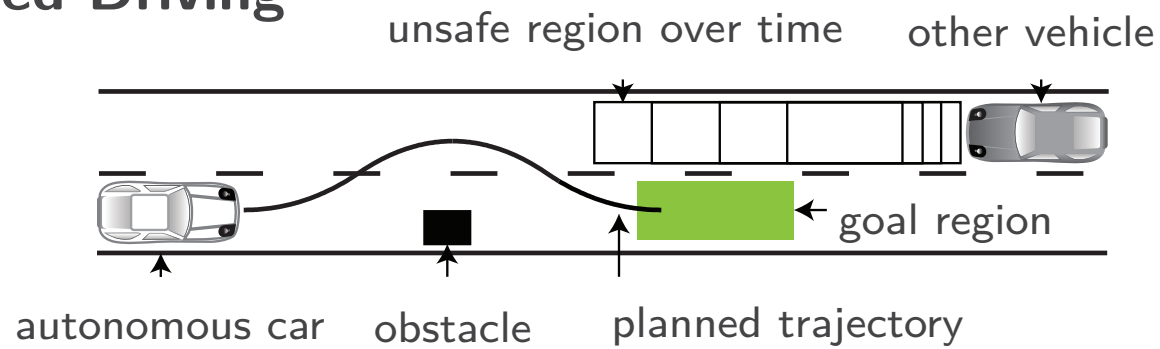


Problem Statement of Our Use Cases

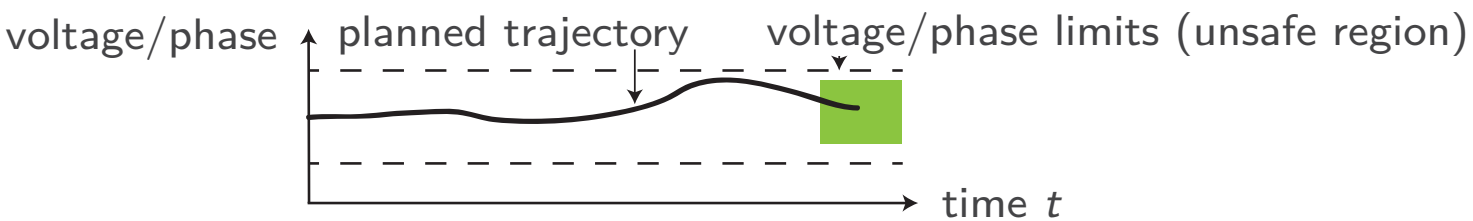
Human-Robot Co-Existence



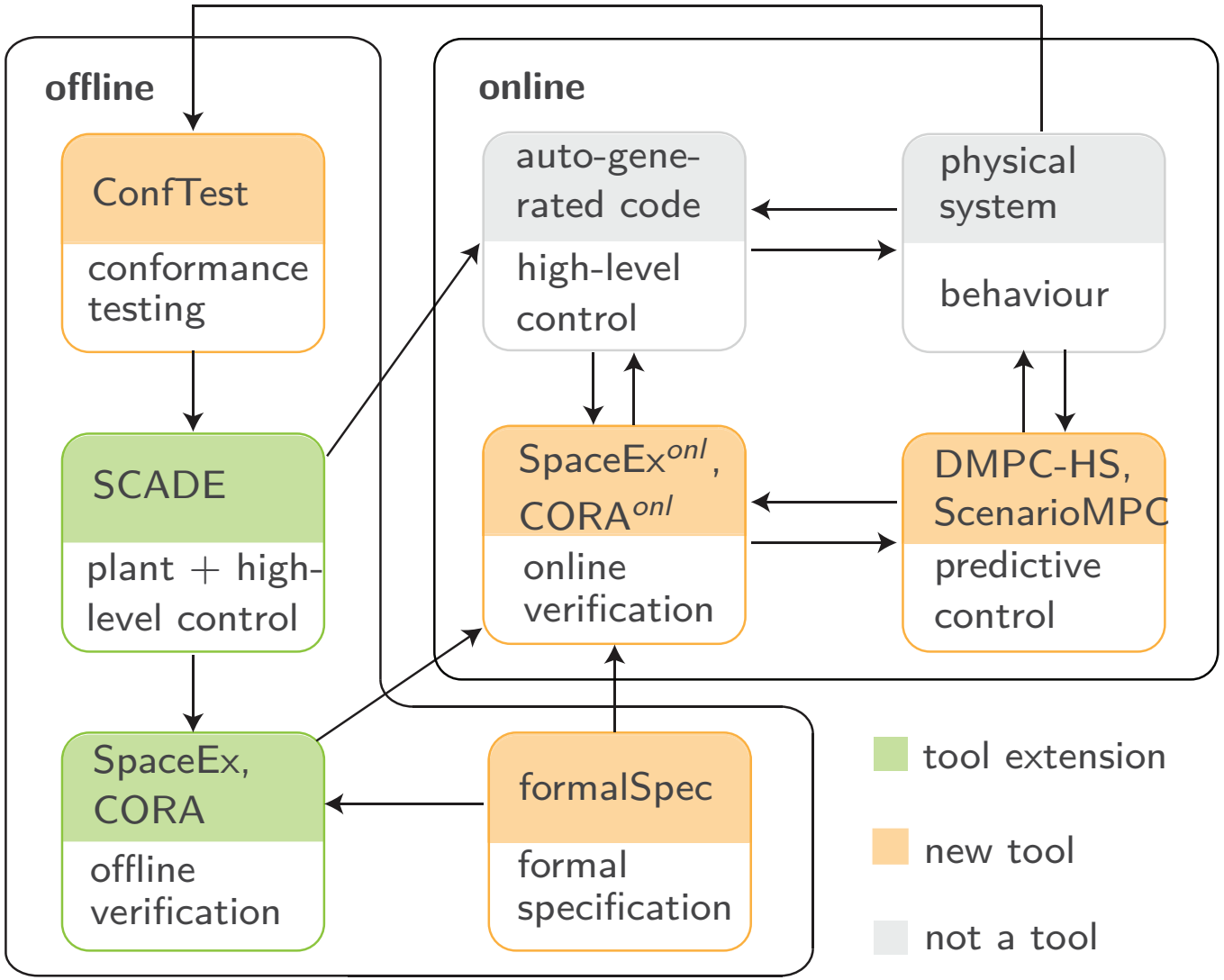
Automated Driving



Smart Grid (similar for wind turbine)

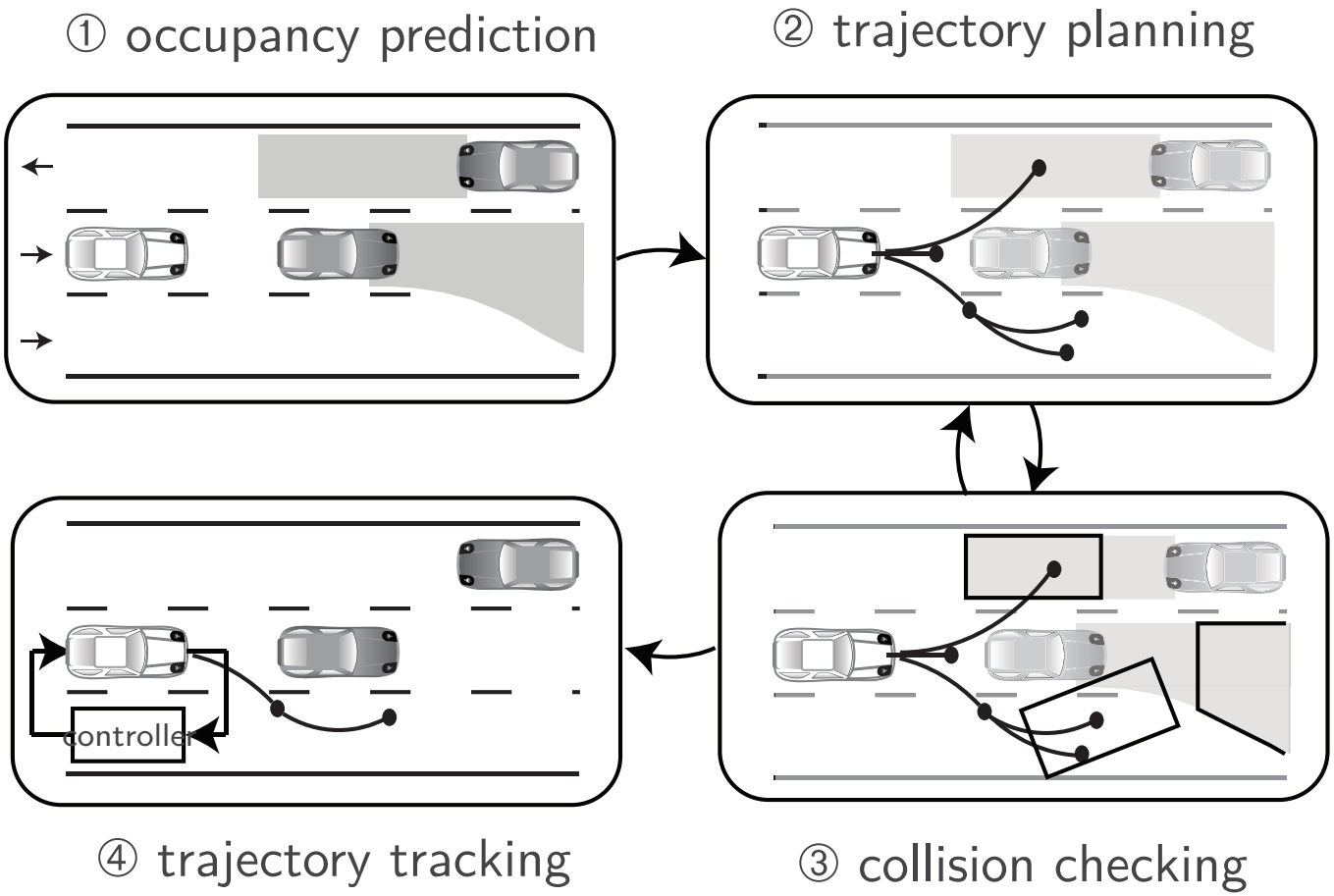


Tool chain of UnCoVerCPS



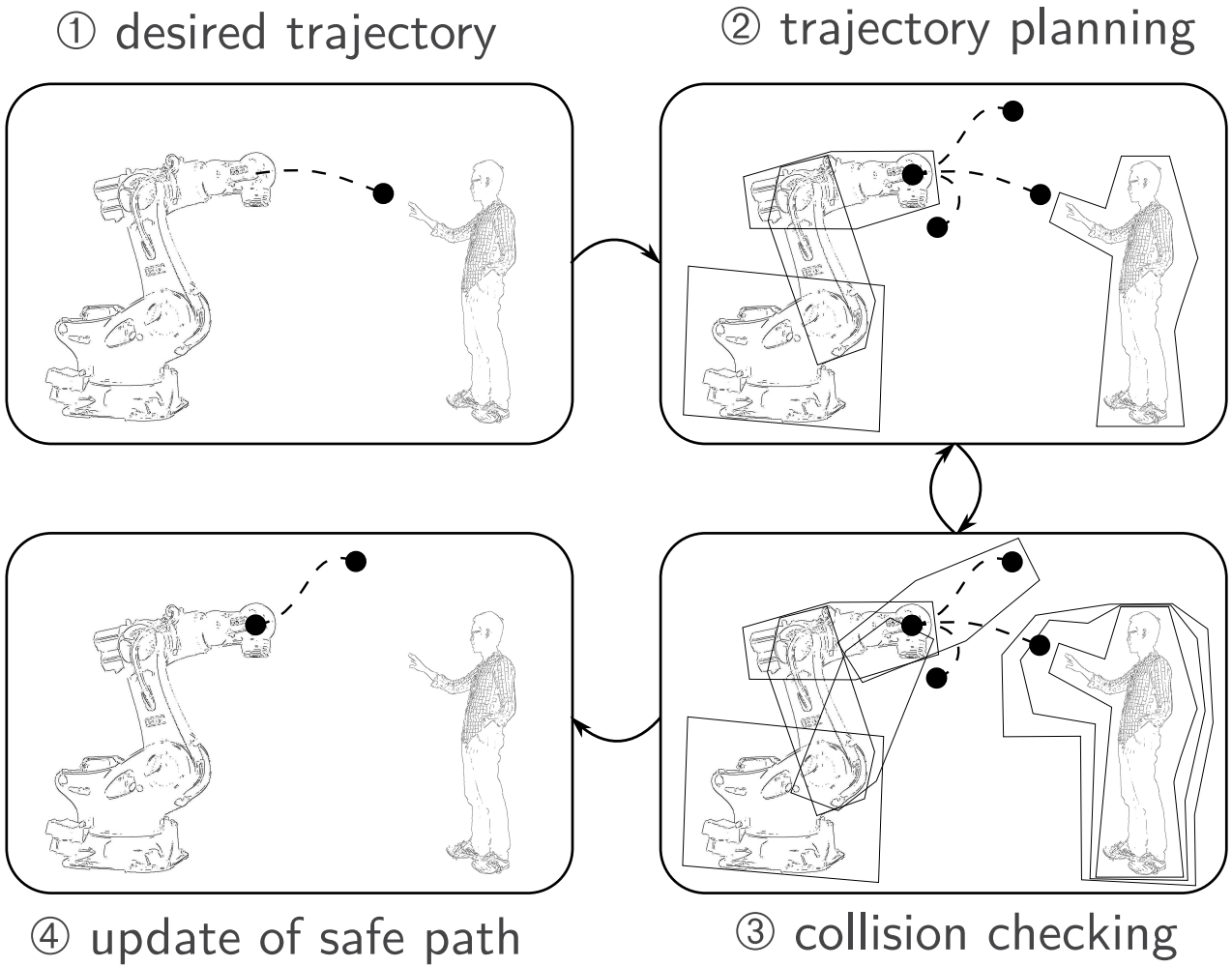
Prediction-planning-verification-control loop

Use case: **automated driving**



Prediction-planning-verification-control loop

Use case: **human-robot co-existence**



This project has received funding from the European Union's Horizon 2020
research and innovation programme under grant agreement No 643921.

UnCoVer CPS

- Safety-critical, autonomous cyber-physical systems require on-the-fly control and verification.

This project has received funding from the European Union's Horizon 2020
research and innovation programme under grant agreement No 643921.

UnCoVer CPS

- Safety-critical, autonomous cyber-physical systems require on-the-fly control and verification.
- Changing environments require to unify control and verification to meet formal specifications.

This project has received funding from the European Union's Horizon 2020
research and innovation programme under grant agreement No 643921.

UnCoVer CPS

- Safety-critical, autonomous cyber-physical systems require on-the-fly control and verification.
- Changing environments require to unify control and verification to meet formal specifications.
- Advances are also beneficial to offline control and verification.

This project has received funding from the European Union's Horizon 2020
research and innovation programme under grant agreement No 643921.

UnCoVer CPS

- Safety-critical, autonomous cyber-physical systems require on-the-fly control and verification.
- Changing environments require to unify control and verification to meet formal specifications.
- Advances are also beneficial to offline control and verification.
- Our approach works across several application domains (de-verticalization).

This project has received funding from the European Union's Horizon 2020
research and innovation programme under grant agreement No 643921.

UnCoVer CPS

- Safety-critical, autonomous cyber-physical systems require on-the-fly control and verification.
- Changing environments require to unify control and verification to meet formal specifications.
- Advances are also beneficial to offline control and verification.
- Our approach works across several application domains (de-verticalization).
- We combine our expertise to establish a unique toolchain for future development of cyber-physical systems.

TUM JKU UNIKASSEL VERSITÄT POLITECNICO MILANO GE BOSCH ESTEREL technologies DLR tecnalia Inspiring Business RUROBOTS Cognitive Science at Work