Robotics in Practise – The Truck as a Mobile Robot

Dr.ir. Rudolf Huisman

Senior Control Engineer

DAF Trucks N.V. – Safety & Driver Controls

(Parttime) Assistant Professor

Eindhoven University of Technology – Robotics

MRC 2023-Q4

Learning objectives this guest lecture

1. List the **pros and cons** of **automated driving** for **trucks**
2. Explain what **building blocks** are to be developed
3. Give concrete **examples** of actually used **sensors** and **algorithms**
4. Explain how **MRC LO’s** relate to the **development** of **autonomous trucks**

Automated driving of trucks

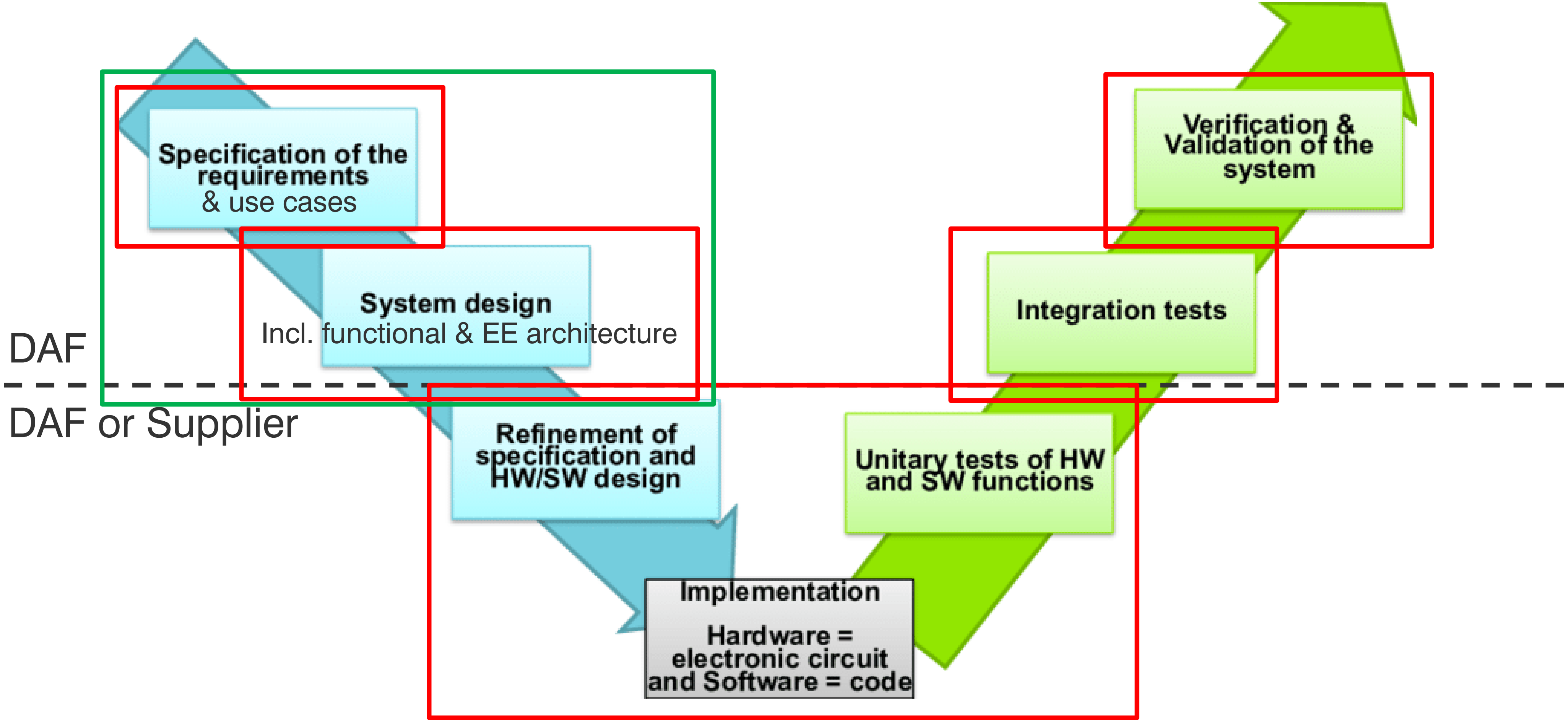
Pros

- Efficiency improvement (24/7)
- Driver shortage
- Cost reduction (driver 1/3 total costs)
- Safety improvement potential (90% accidents human cause)

Cons

- Safety (potential not proved)
- Development costs (AD is not easy)
- Security (hacking)
- Public opinion / driver unions

Development process



[ISO 26262 V-cycle Development Process. | Download Scientific Diagram (researchgate.net)]

Use case example

ID	UC01 (step 2)
Name	Drop off
Description	Driver gets out at terminal gate and truck switches to autonomous operation
Initial condition	Truck arrives at the terminal (step 1)
Trigger	Driver activates Yard Automation (YA) via switch
Sequence	1. Driver stops vehicle at drop off area and activates park brake 2. Driver activates YA via switch 3. YA searches for Control Tower wireless network 4. YA requests driver to accept connection with network 5. When connected: YA takes over vehicle control and informs driver 6. Driver leaves truck
Final condition	YA is enabled, vehicle and Control Tower are connected, driver outside truck

[Tran2020, ANITA project]

Requirements example

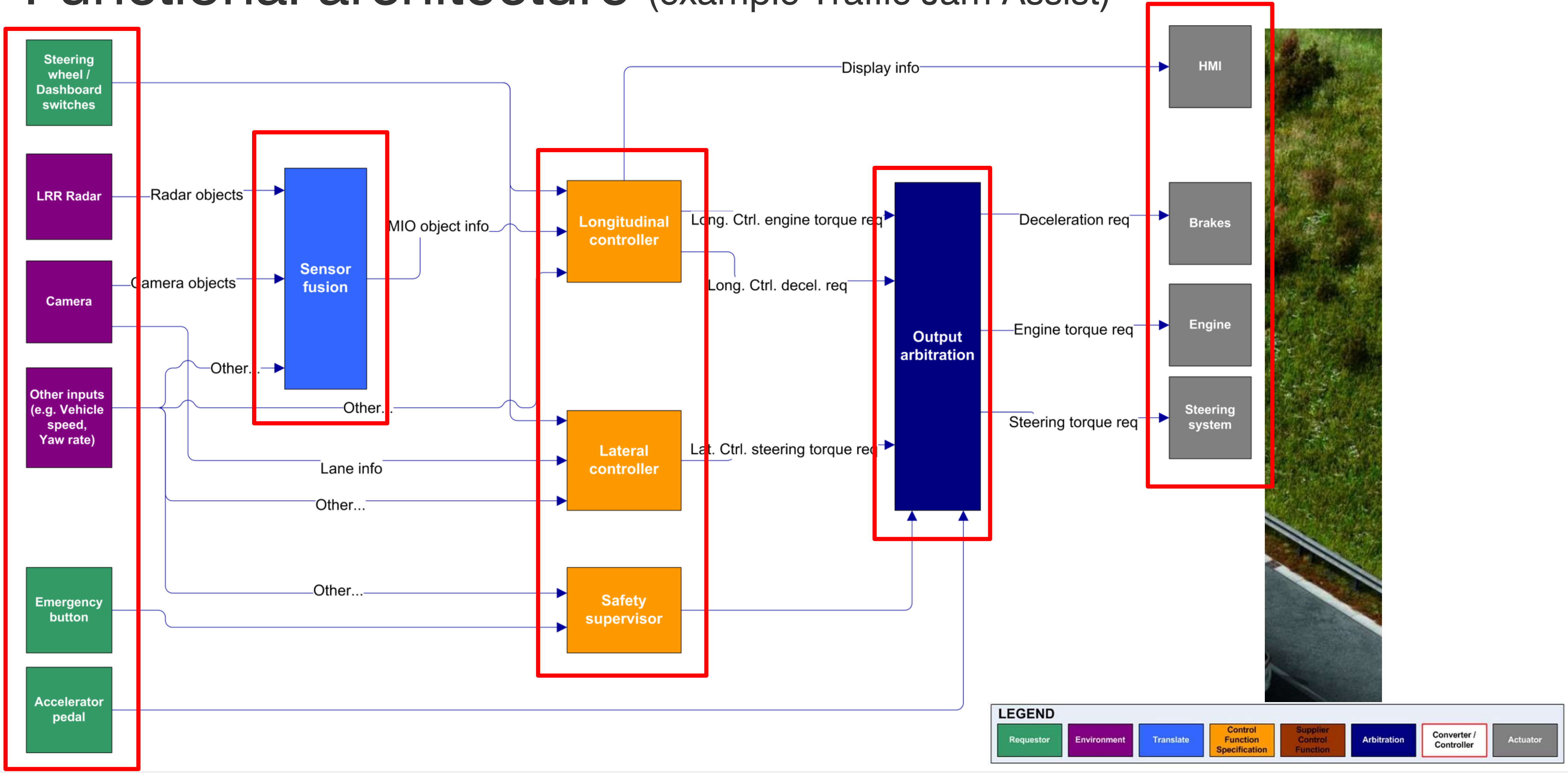
REQ01-001	Description:	Driver shall be able to activate YA e.g. via a switch if the following conditions are fulfilled: ▪ Vehicle is standing still ▪ Park brake is applied
	Name:	**Driver activation**
	Source:	DAF, UC01
	Rationale:	Driver shall be able to decide when YA can take over the driving task. The vehicle shall be in a safe state to transfer the control from driver to YA.

Functional requirement

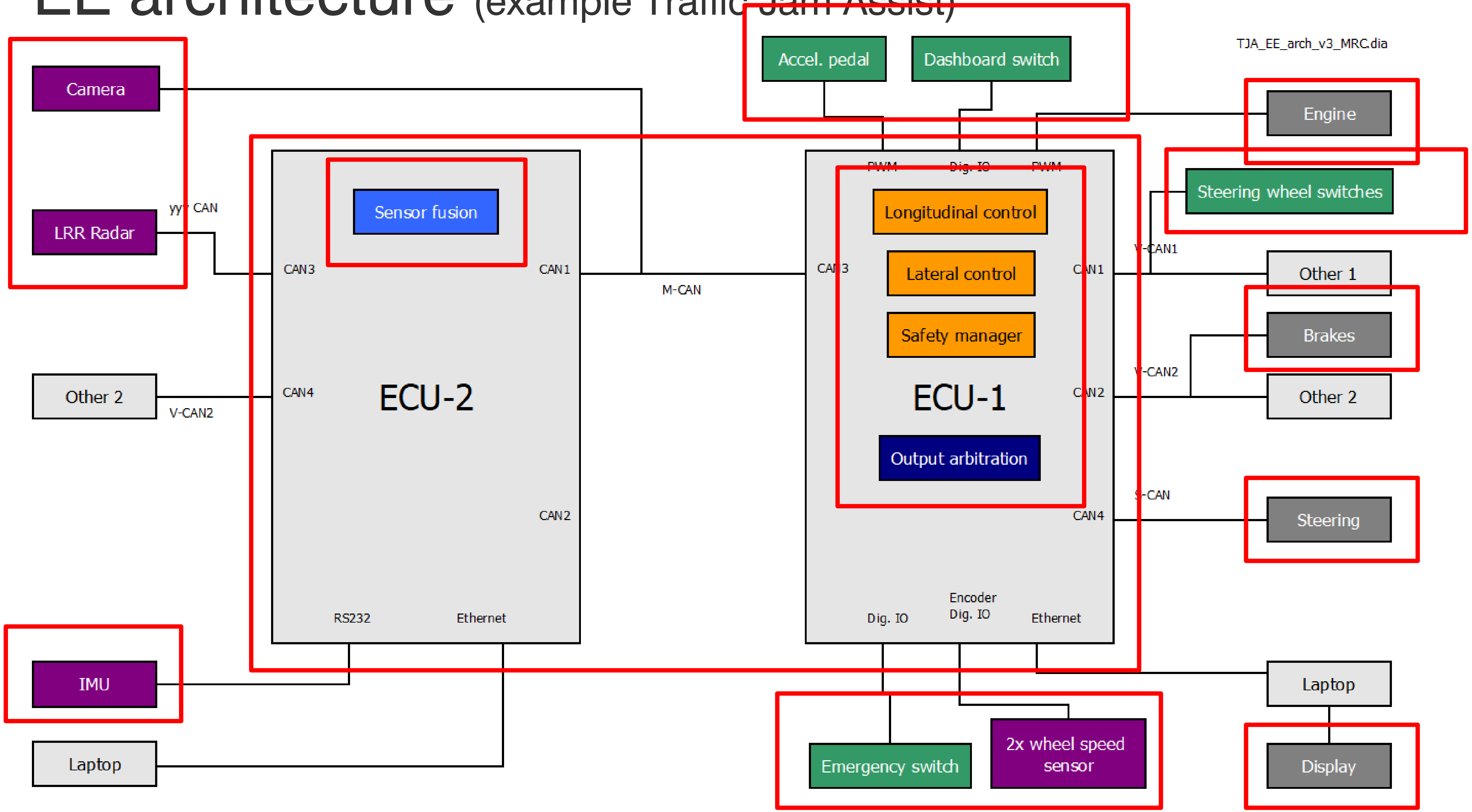
REQ00-001	Description:	YA shall be available in the vehicle speed range from -5 km/h up to 30 km/h.
	Name:	**Vehicle speed range**
	Source:	DAF
	Rationale:	YA shall include forward and backward driving. Maximum speed limited for safety reasons.

Non-Functional requirement
(e.g. performance)

Functional architecture (example Traffic Jam Assist)

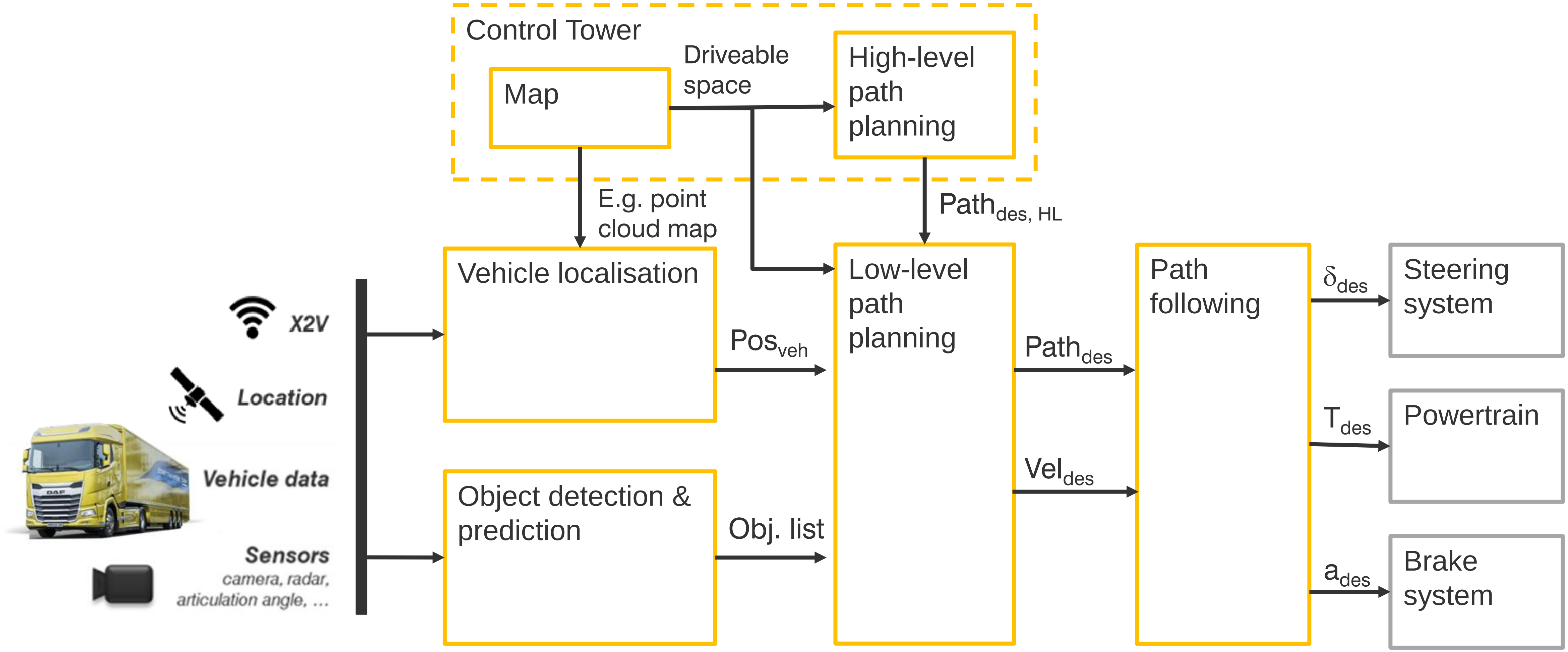


EE architecture (example Traffic Jam Assist)

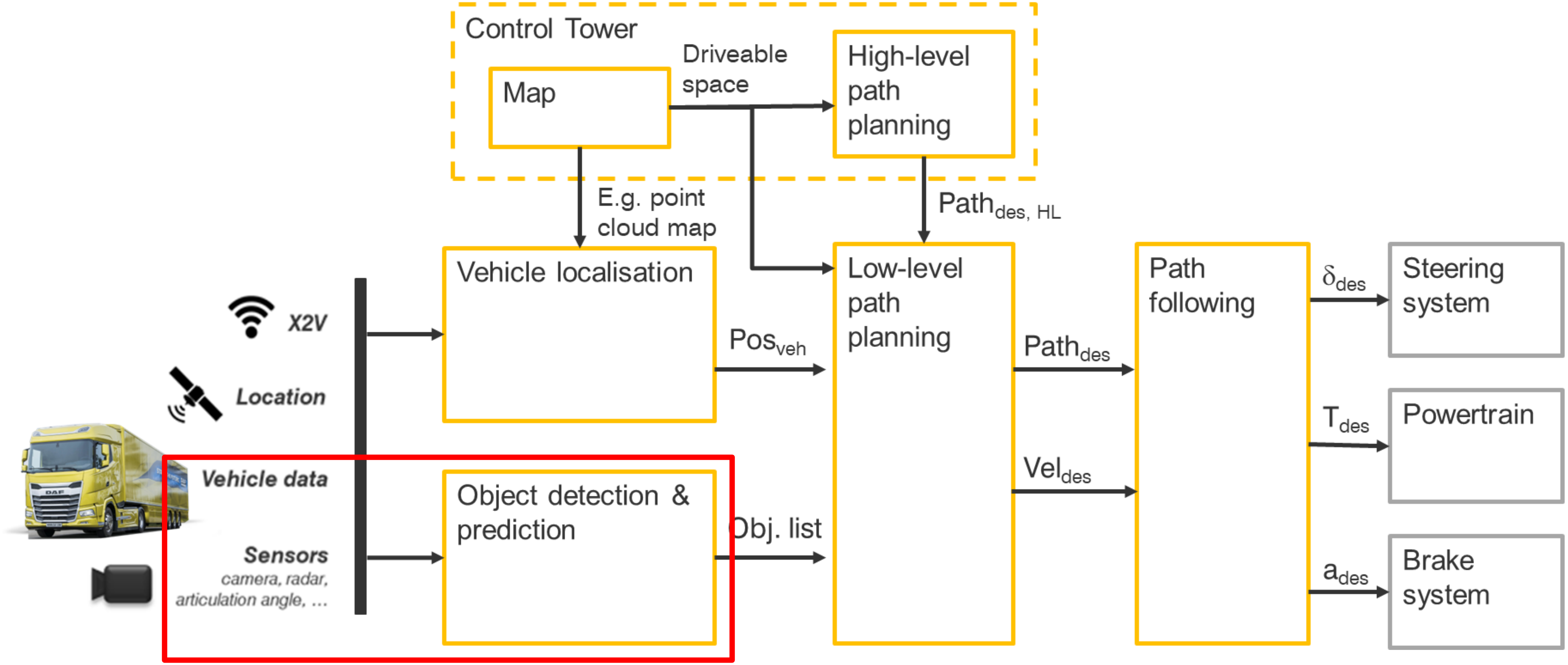



DAF

Building blocks for automated driving



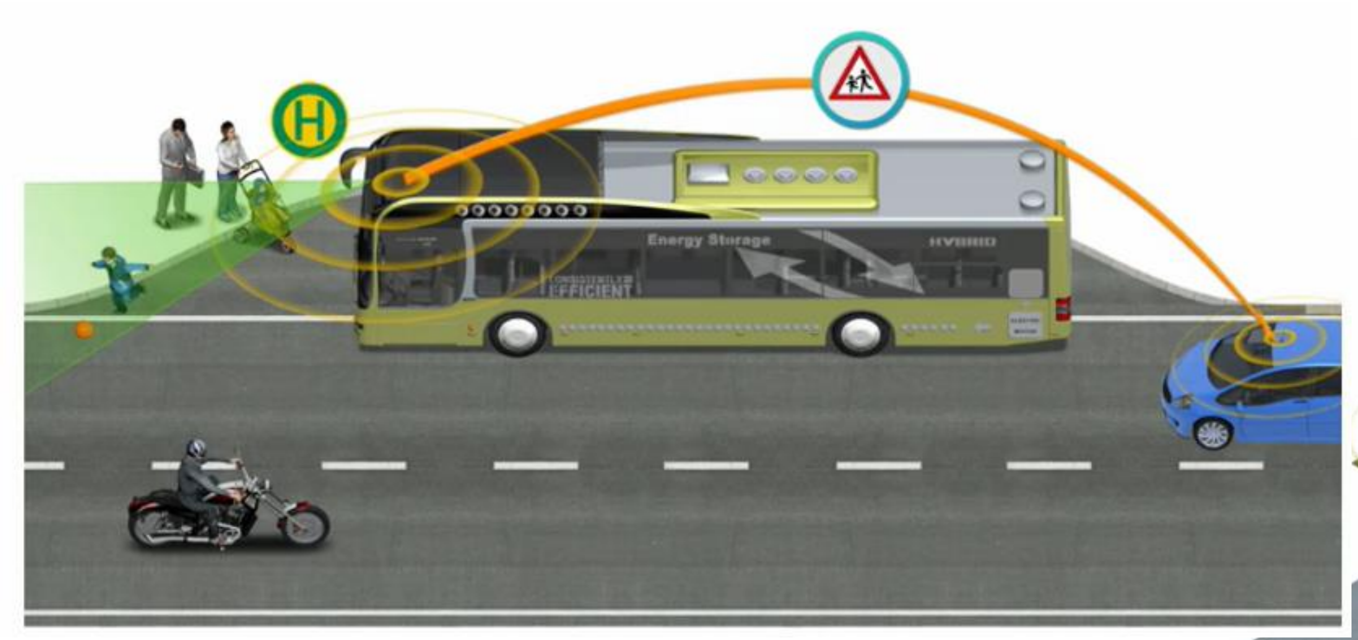
Building blocks for automated driving

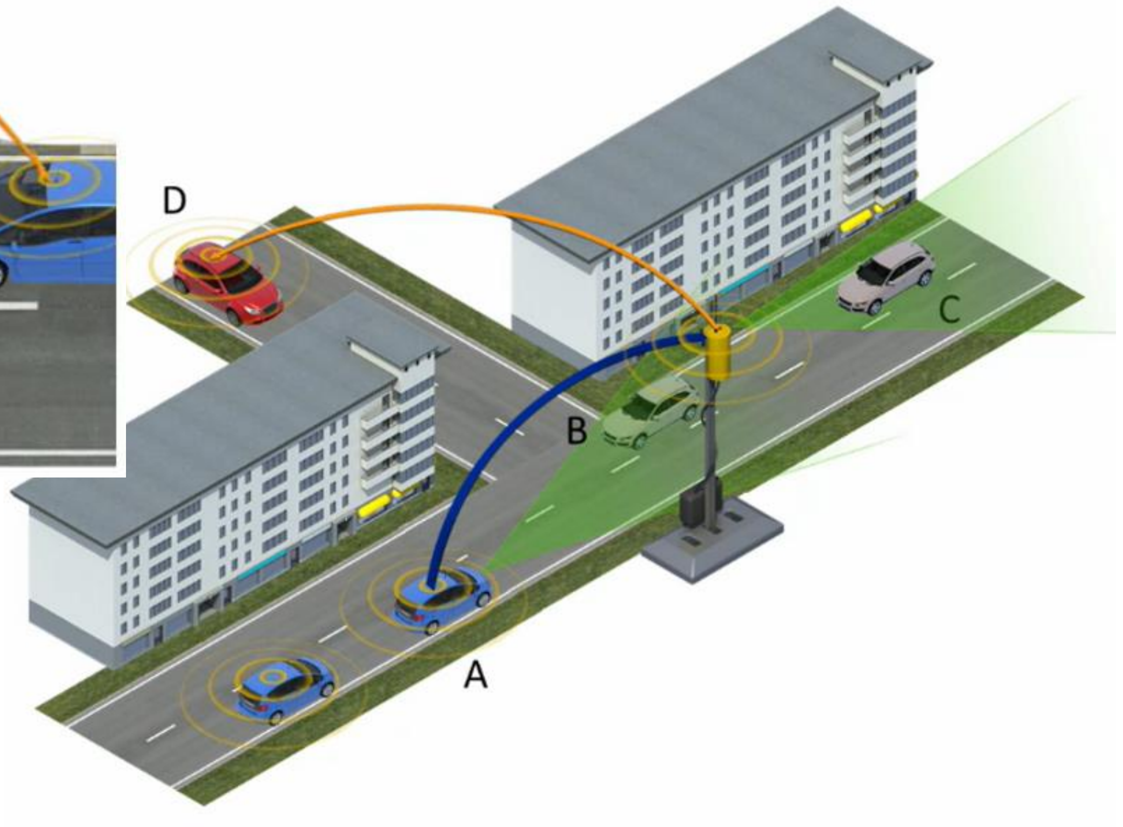


Object detection (using on-board sensors)

Type	Example	Pros	Cons
Camera	 [ZF]	- Good feature detection (e.g. lanes, VRU's) - Accurate in lateral direction	- Bad robustness for weather conditions
Radar	 [Continental]	- Good robustness for weather conditions - Accurate long. distance and speed measurement	- Limited feature detection - Limited lateral distance and speed measurement
Lidar		- Very accurate long. and lat. distance measurement - Suitable for accurate localization	- High costs - Not so robust as radar and ultrasonic
Ultrasonic	 [Bosch]	- Low costs - Good robustness for weather conditions	- No feature detection - Limited accuracy and range

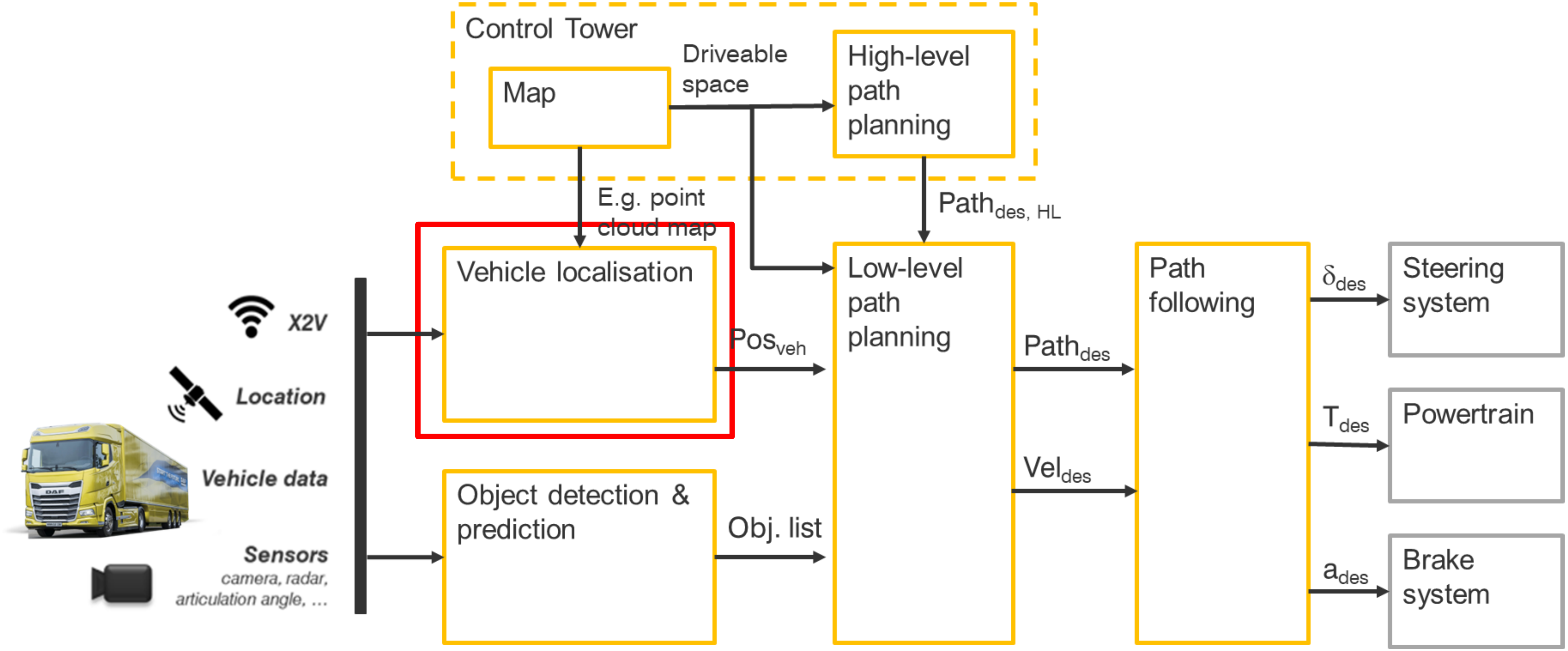
Object detection (using V2X information)



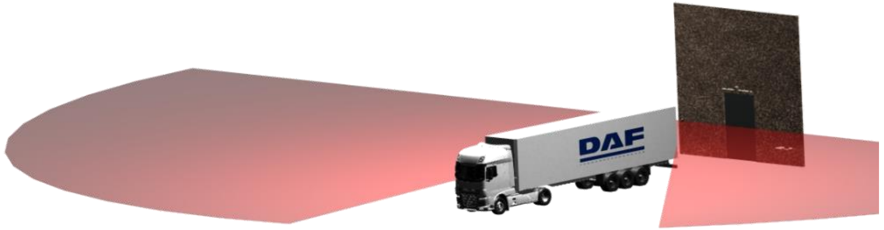


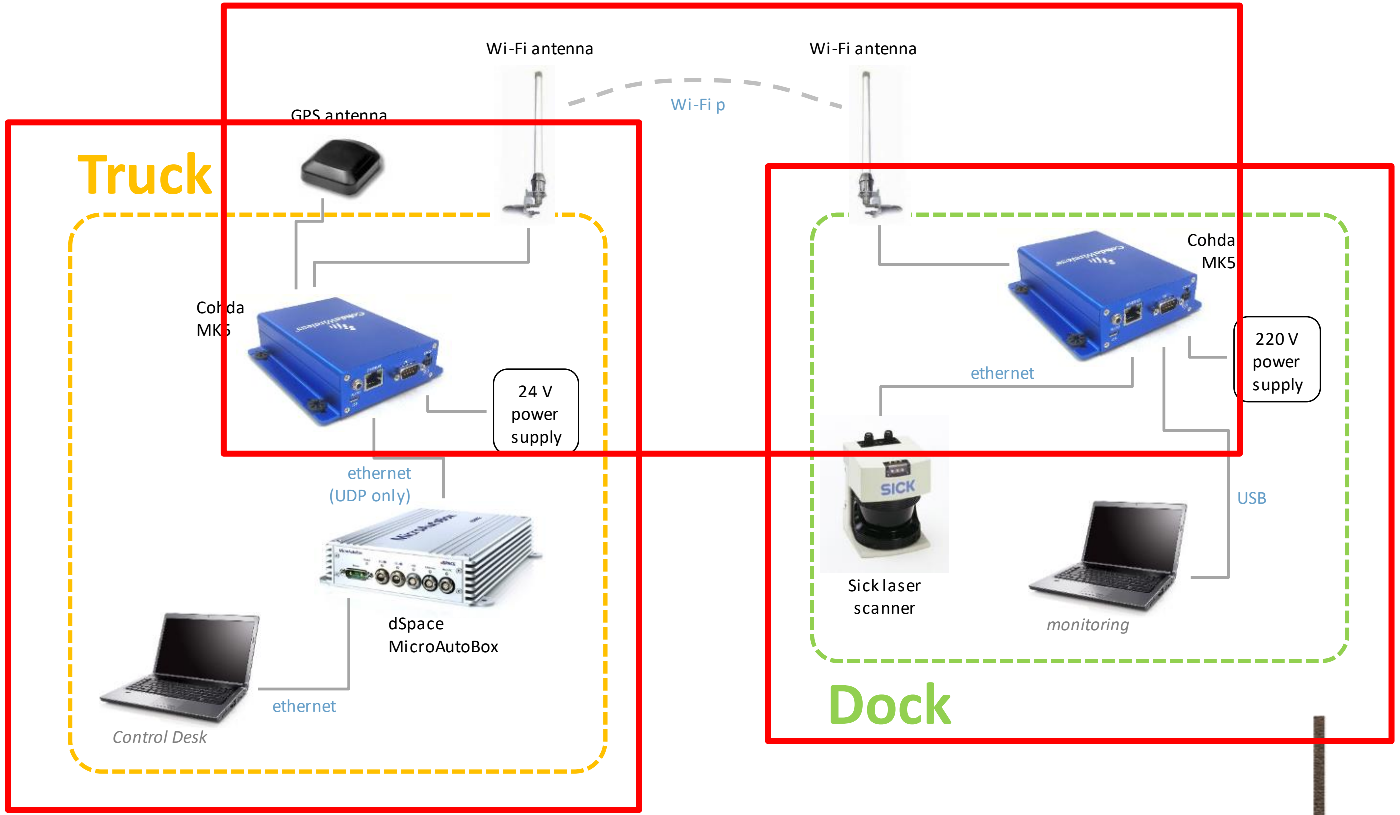
[F. Schiegg. Removing blind spots: Infrastructure-assisted collective perception. VDI ELIV conference 2021]

Building blocks for automated driving

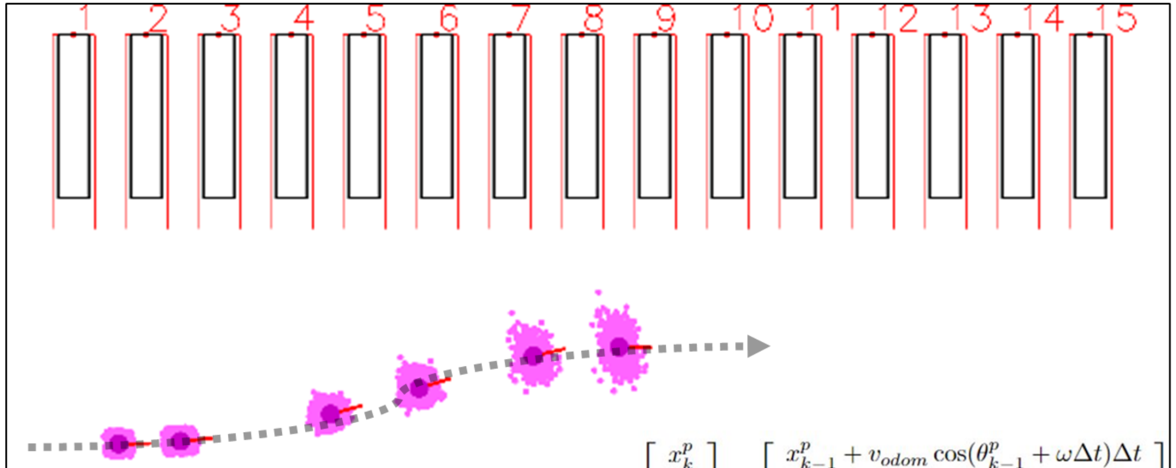


Vehicle localization
(example 1)



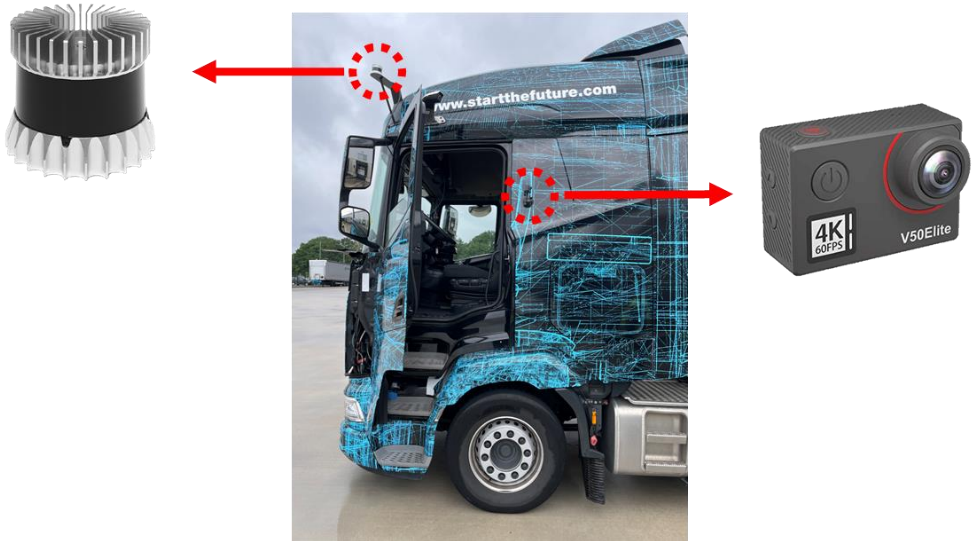


Vehicle localization (example 2)



$$x_k = \begin{bmatrix} x_k^p \\ y_k^p \\ \theta_k^p \end{bmatrix} = \begin{bmatrix} x_{k-1}^p + v_{odom}\cos(\theta_{k-1}^p + \omega\Delta t)\Delta t \\ y_{k-1}^p + v_{odom}\sin(\theta_{k-1}^p + \omega\Delta t)\Delta t \\ \theta_{k-1}^p + \omega\Delta t \end{bmatrix}$$

$$u_k = \begin{bmatrix} v_{odom} \\ \delta_{odom} \end{bmatrix}$$



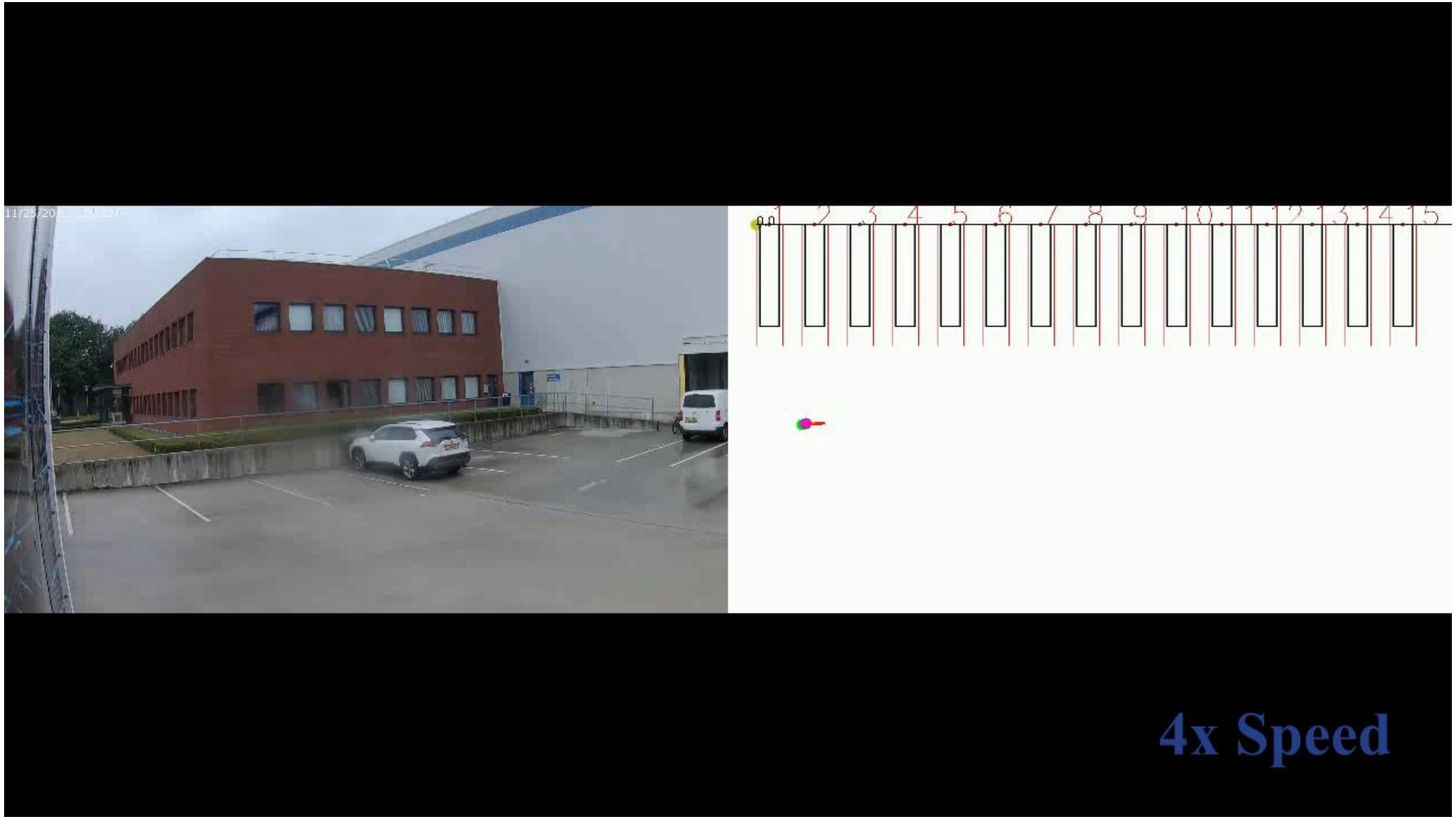
Localization using a particle filter

[Kokkelmans2022, Konings2022]

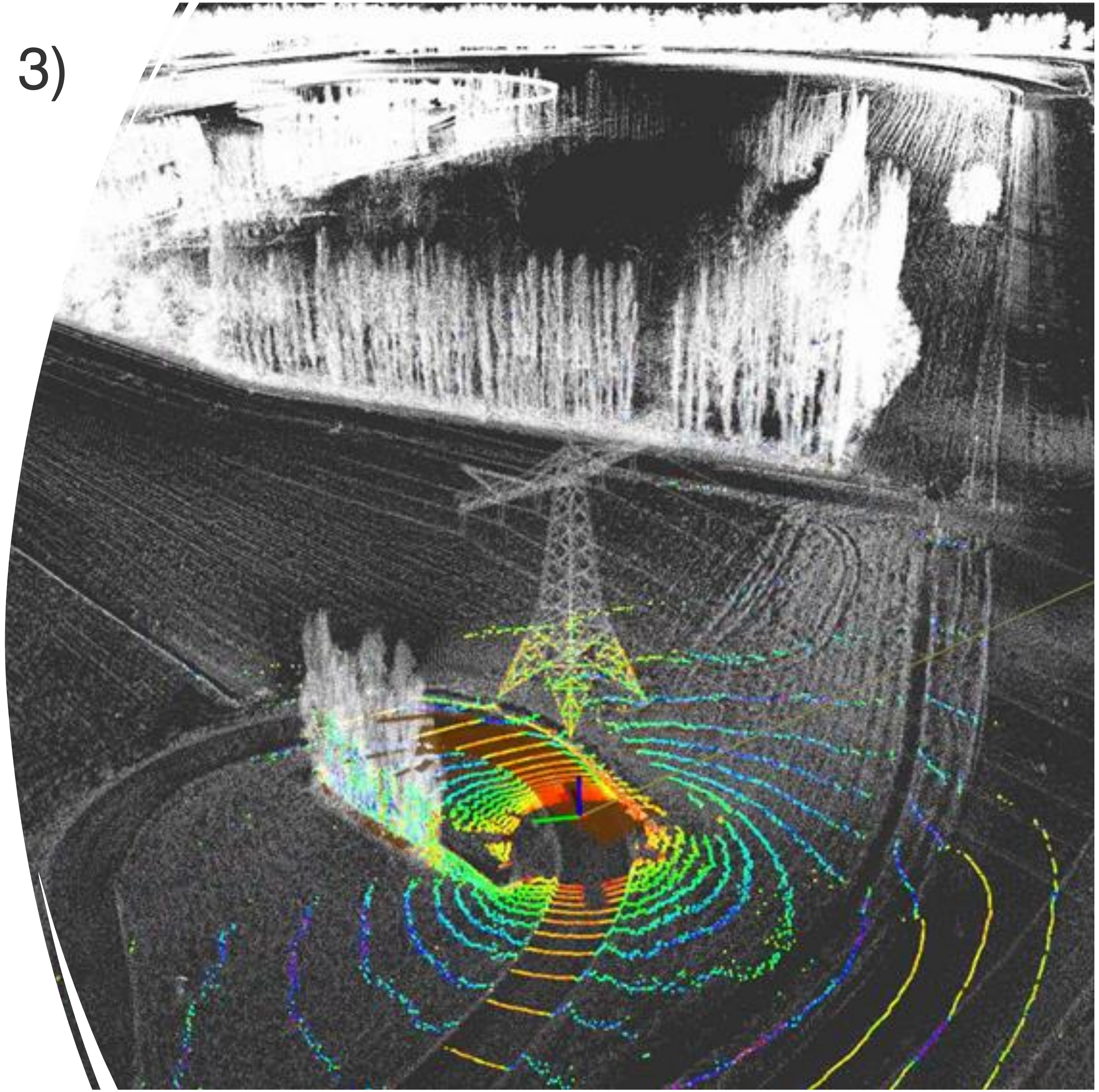
Vehicle localization (example 2)

[Konings2022]

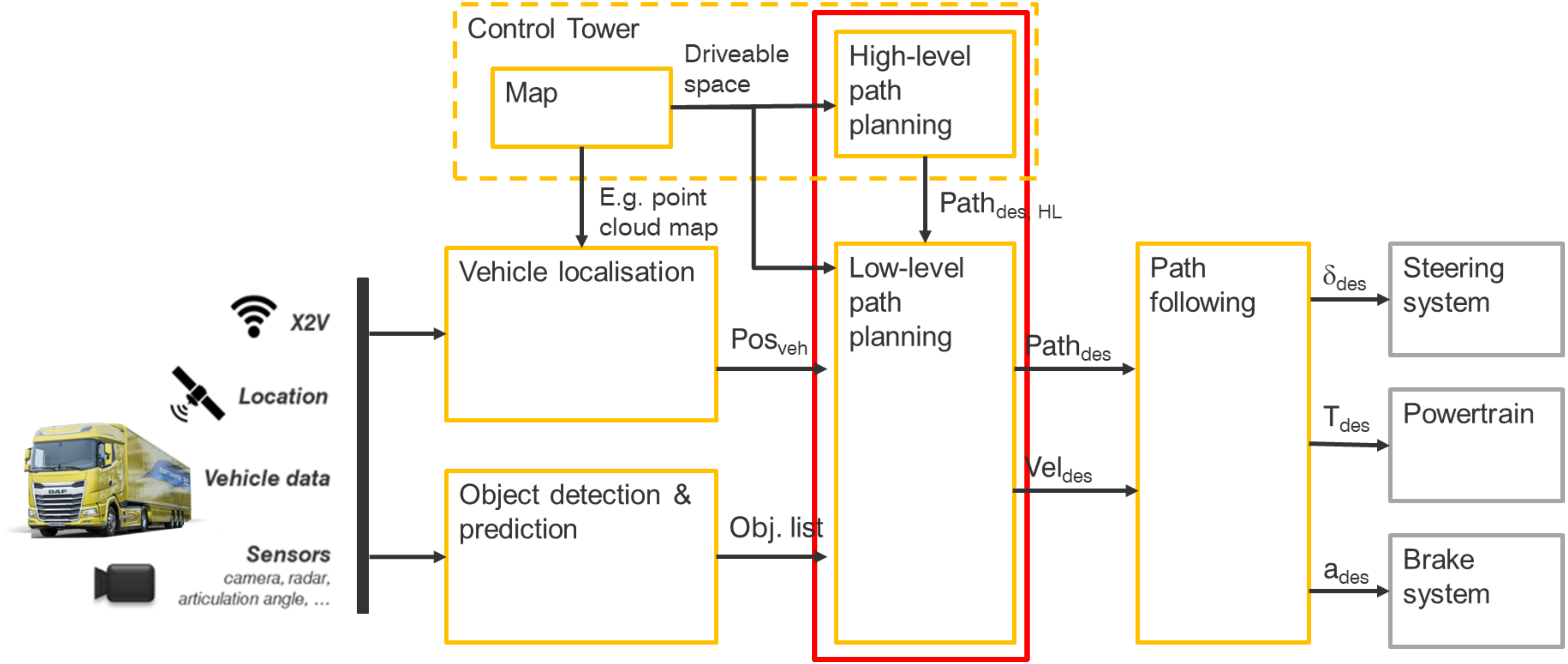
- Green:
  Ground truth
- Yellow:
  Camera based
- Pink:
  Output particle filter



Vehicle localization (example 3)



Building blocks for automated driving



Path planning (example high level planner: Hybrid A*)

- Use the simplified model to find a global path
- Obstacle avoidance at global level
- Decide driving direction

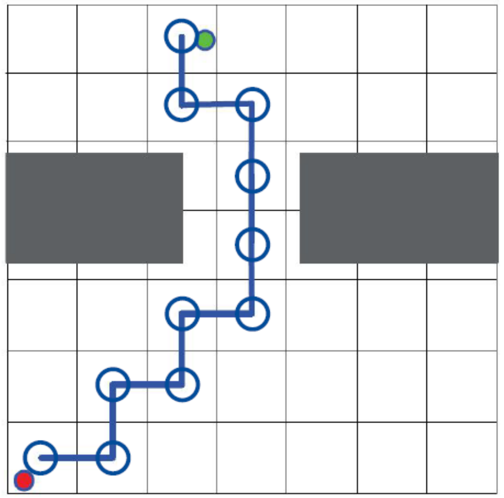


Figure 2.3: Output path of A-star algorithm with start node in red, goal node in green and intermediate nodes indicated with blue circles, obtained from [5].

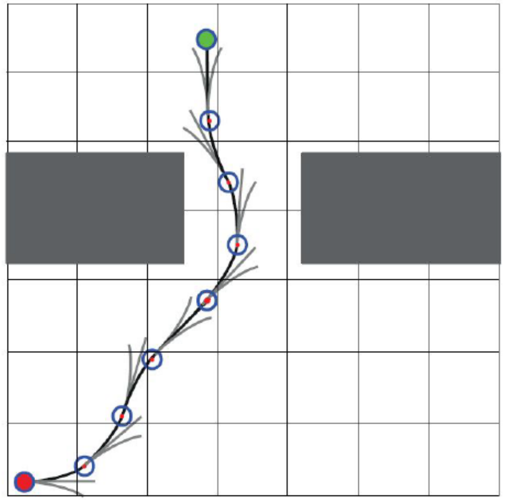
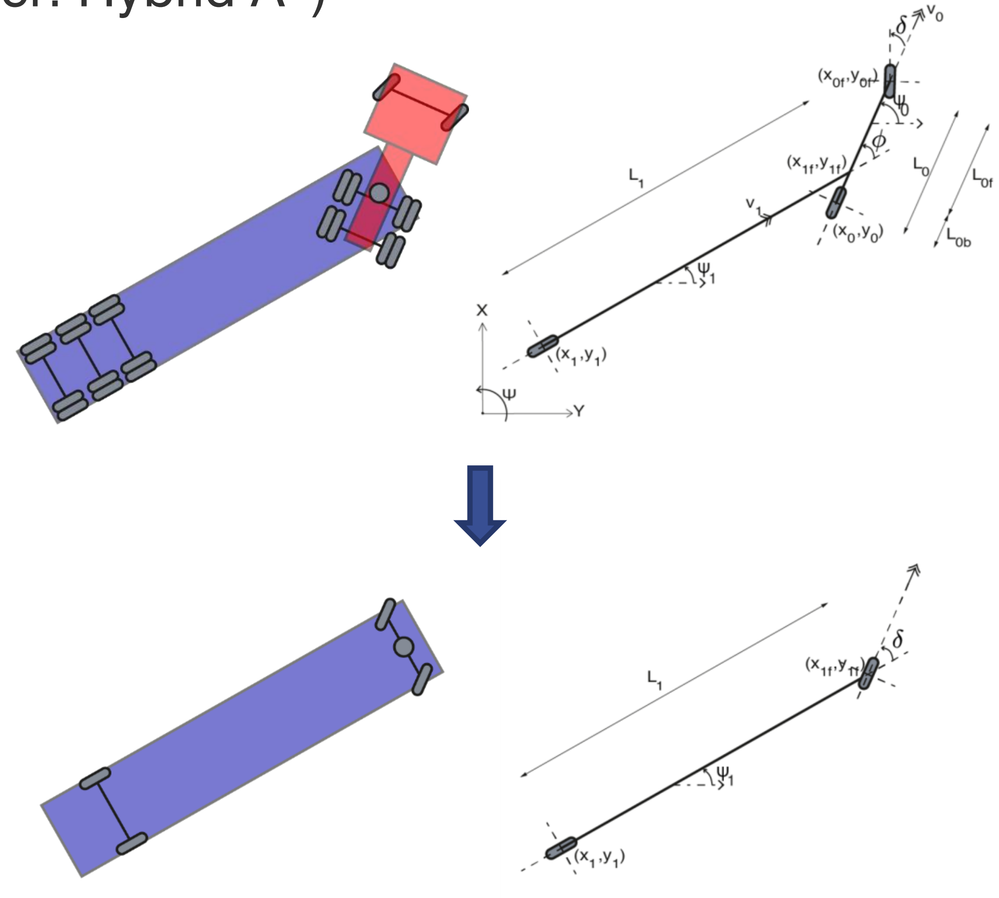


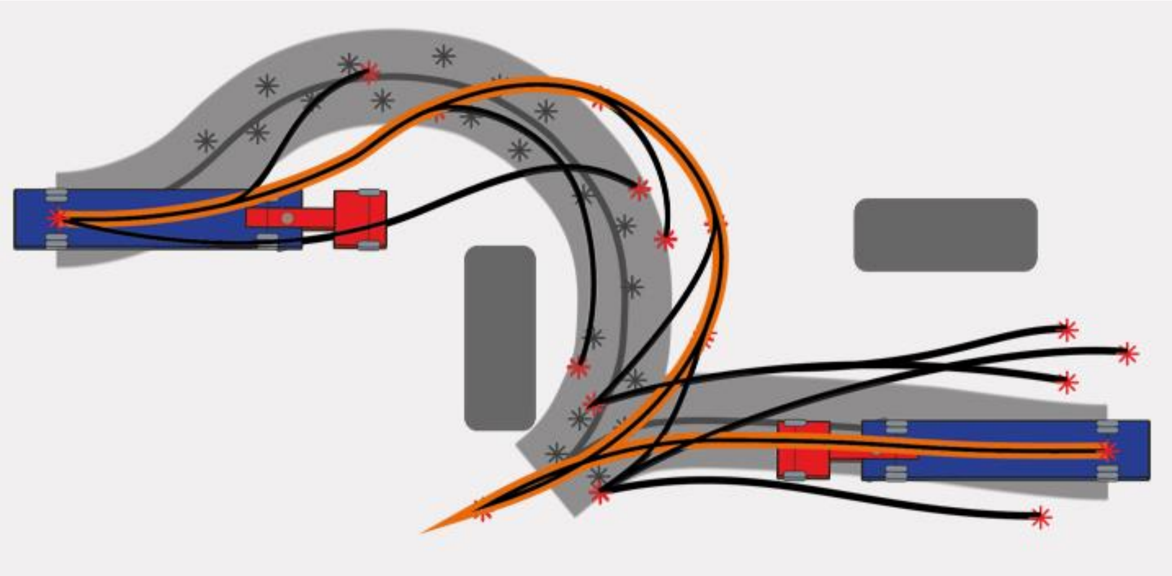
Figure 2.4: Output path of Hybrid A-star algorithm with start node in red, goal node in green and intermediate nodes indicated with blue circles, obtained from [5].

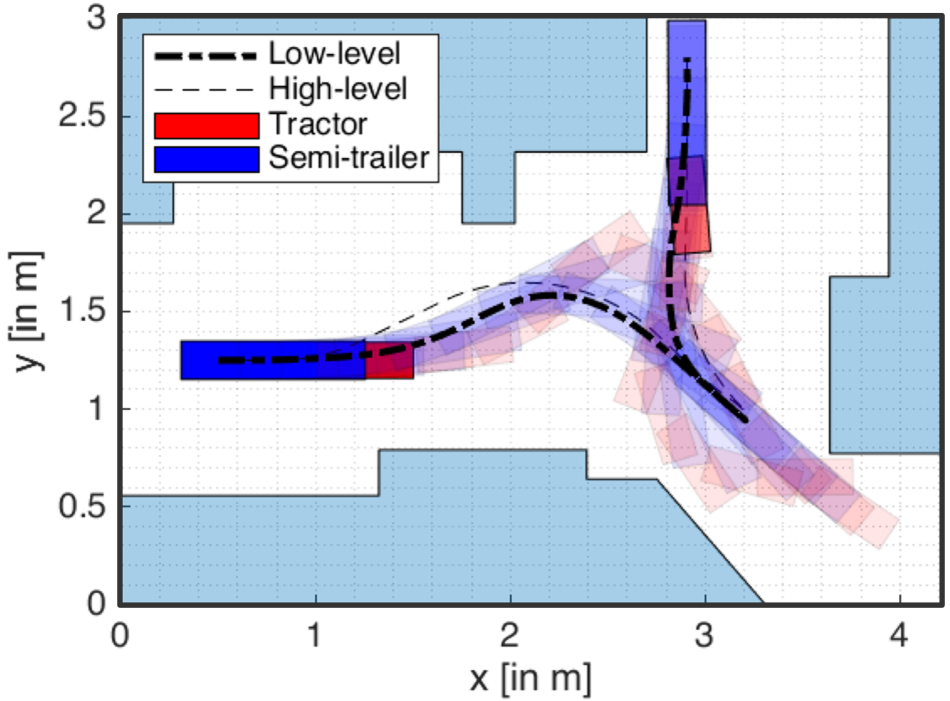


[Nair2019, Hendrix2020]

Path planning (example low level planner: CL-RRT)

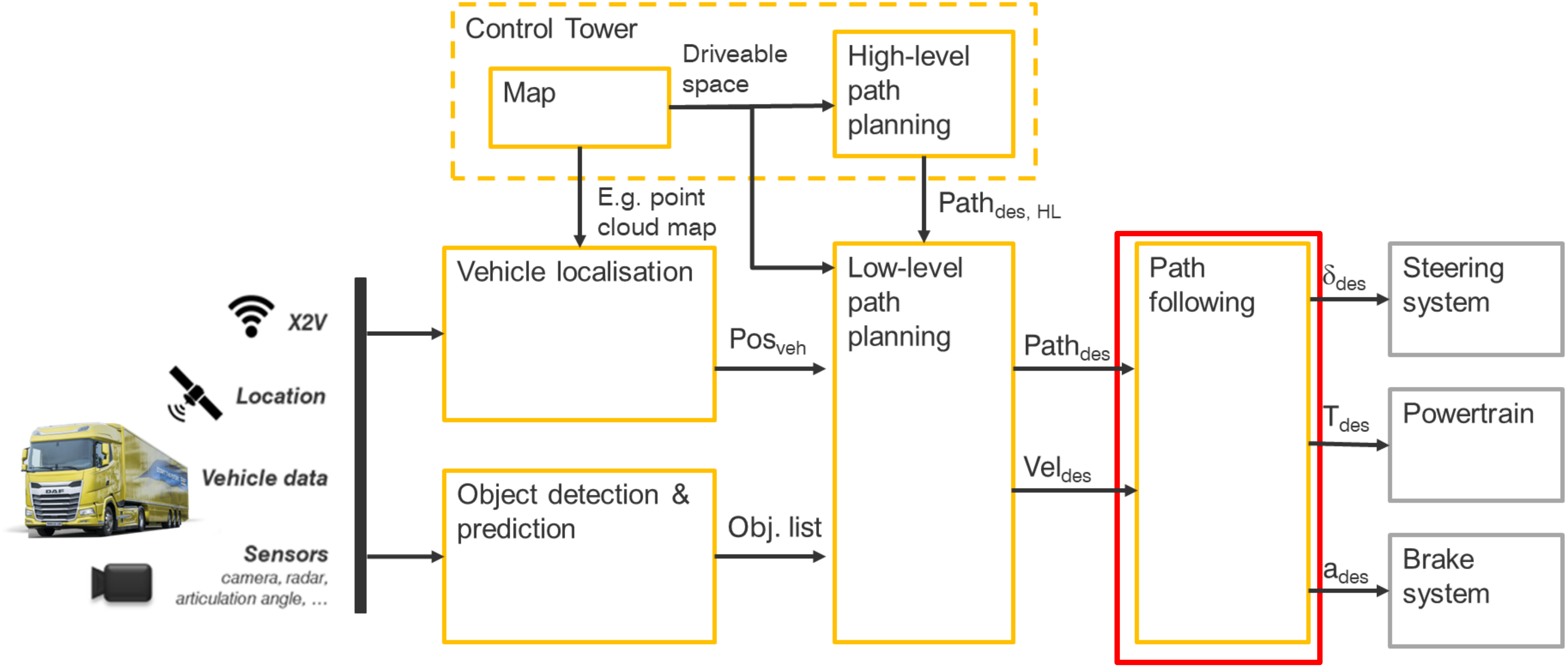
- Use detailed model (incl. path following controller) to plan the final path using high-level path as input
- For forward and backward driving
- Ensure final docking accuracy





[Nair2019, Hendrix2020]

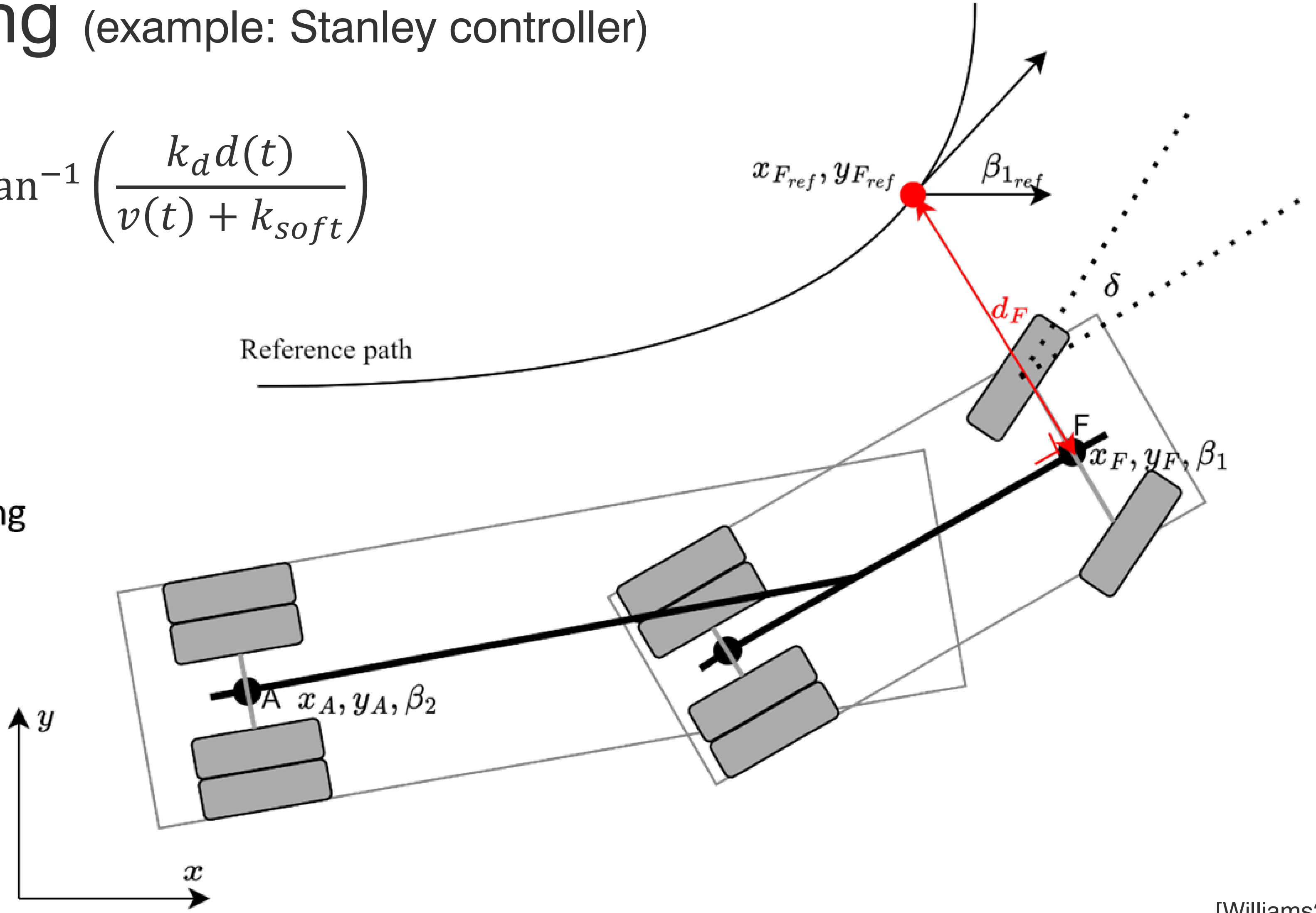
Building blocks for automated driving



Path following (example: Stanley controller)

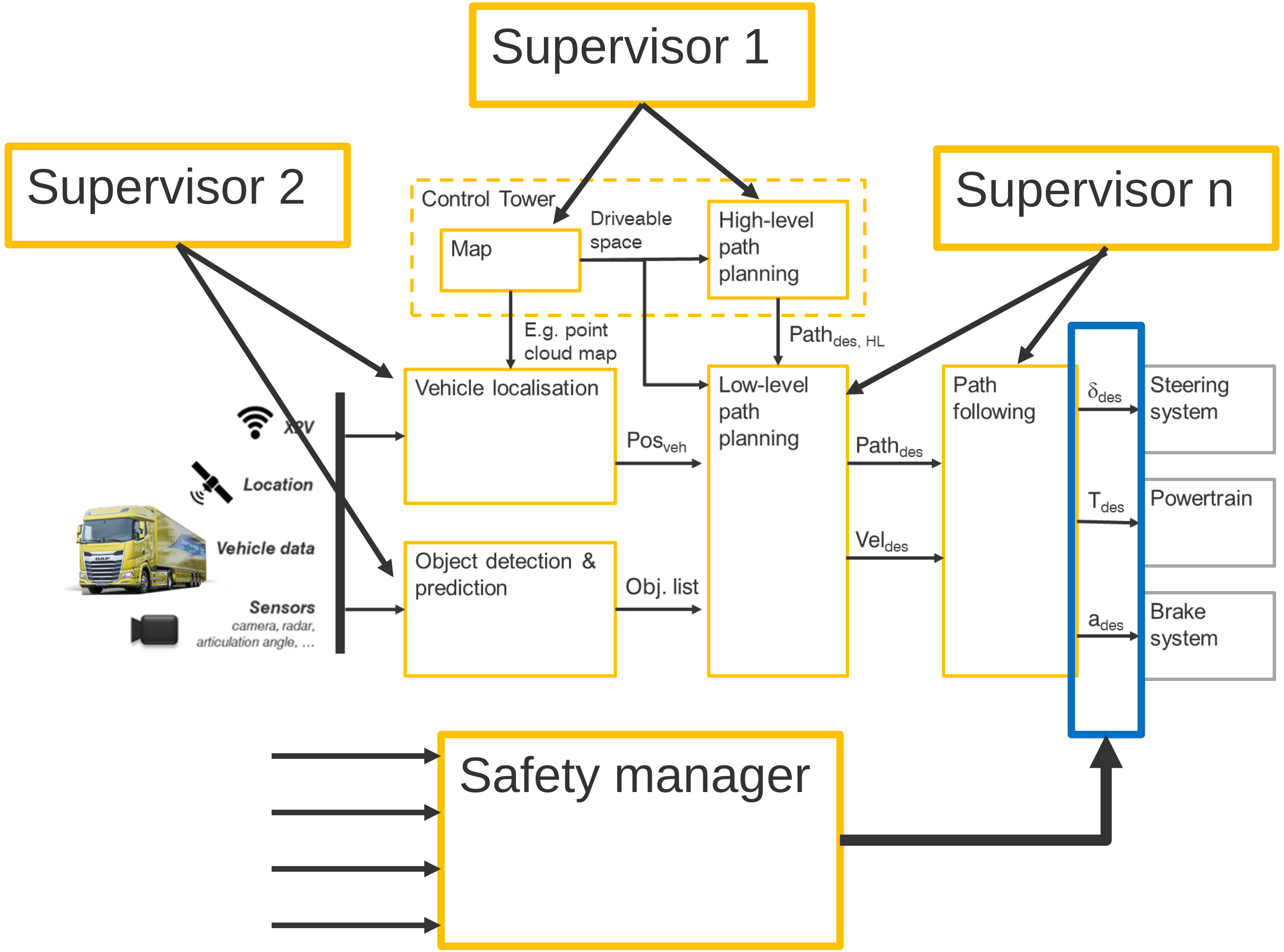
$$\delta(t) = \beta_{error}(t) + \tan^{-1}\left(\frac{k_d d(t)}{v(t) + k_{soft}}\right)$$

Forward driving



[Williams2024]

Supervisor(s) & Safety Manager



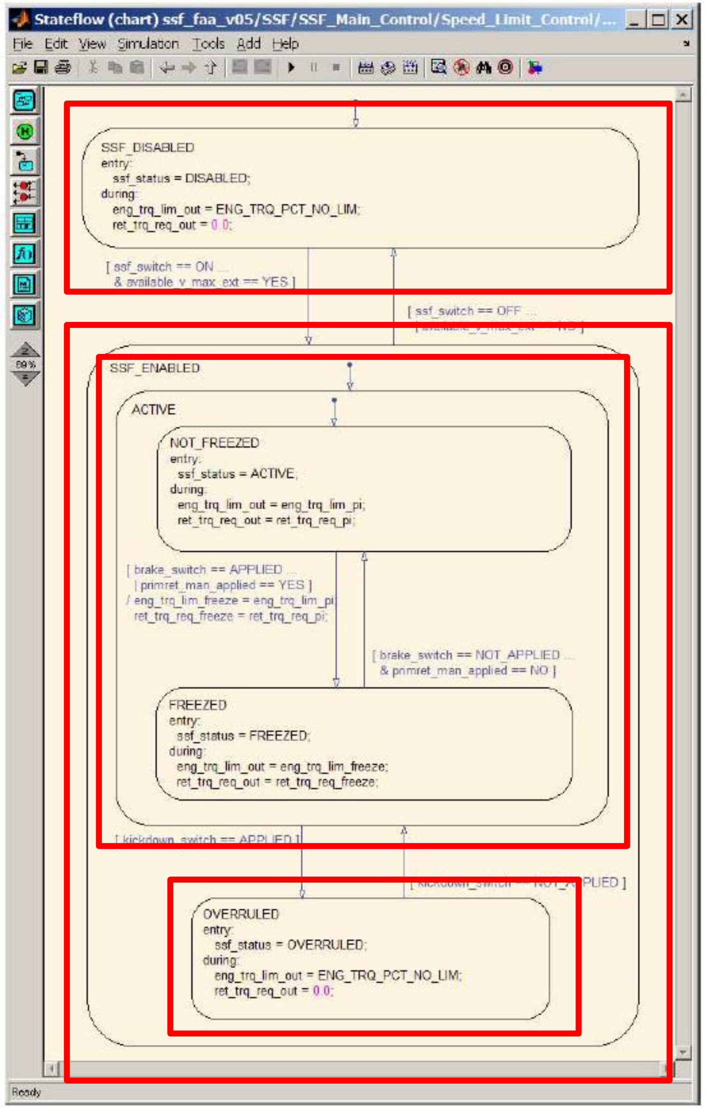
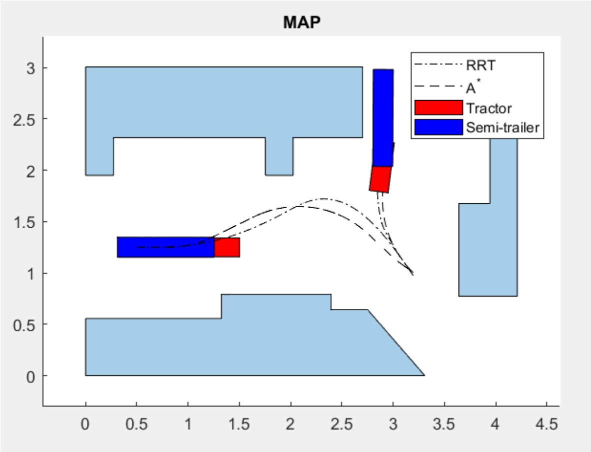


Figure 9. Example function behaviour specification of the SSF (i.e. state chart showing the SSF enable and disable logic).

Verification & Validation



Functional and failure tests

- Model-based simulations
- Software-In-the-Loop
- Hardware-In-the-Loop
- Rapid Controller Prototyping
- Vehicle tests

Exercise building blocks, sensors and algorithms

MRC LO’s versus development autonomous trucks

No.	Learning objectives	Actually developing an autonomous truck
1	Describe the ***challenge*** of ***autonomous mobile robot***	Use cases and requirements
2	Describe and develop a ***global path planning algorithm*** (e.g. A*, RRT)	High-level path planning (Hybrid A*)
3	Describe and develop a ***local path planning algorithm*** (e.g. APF, DWA)	Low-level path planning (CL-RRT)
4	Describe and develop a ***localization algorithm*** (e.g. particle filter)	Vehicle localization (feature detection, particle filter, point cloud matching (NDT))
5	Design an ***architecture*** that ***integrates different algorithms*** to enable a mobile robot to fulfill a given use-case	Functional / EE-Architecture (EE = Electrical-Electronical)
6	***Validate*** your system architecture on a ***physical robot***	Rapid Control Prototyping, Hardware-in-the-Loop, Vehicle Testing
7	Use ***tools*** common in robotics industry	Simulink, dSPACE, Python, ROS…

Concluding suggestions

- The “what”:
  - Use cases
  - Requirements
- The “how”:
  - First high-level architecture
  - Then detailed design
- Start simple
- Integrate & test regularly

  (try to avoid “big-bang”)