Universität Augsburg
Fakultät für Angewandte Informatik

Praktikum Autonomes Fahren

Group 1

Chair of Computer engineering with a focus on mechatronics

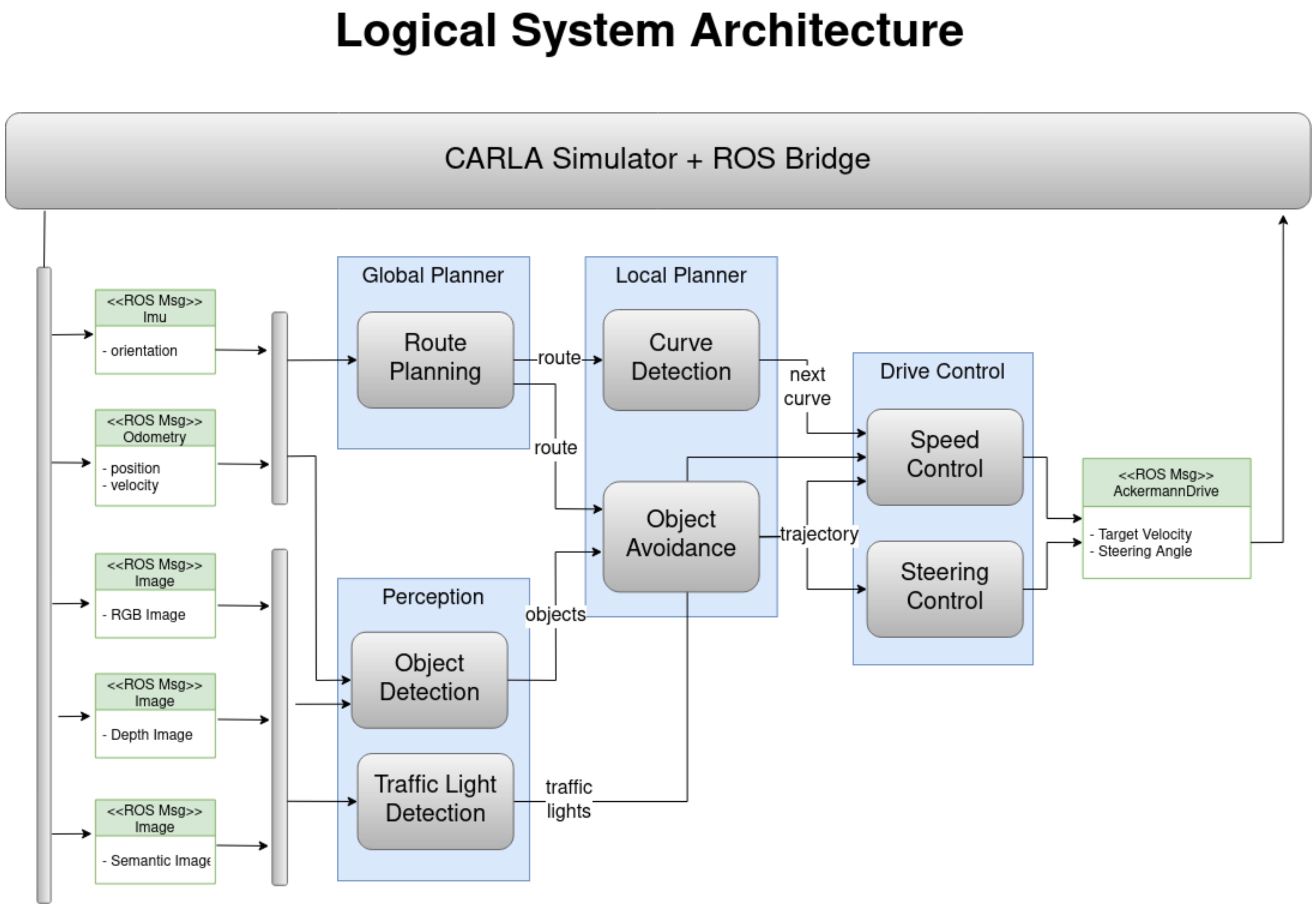
31. March 2022, Augsburg

Agenda

01

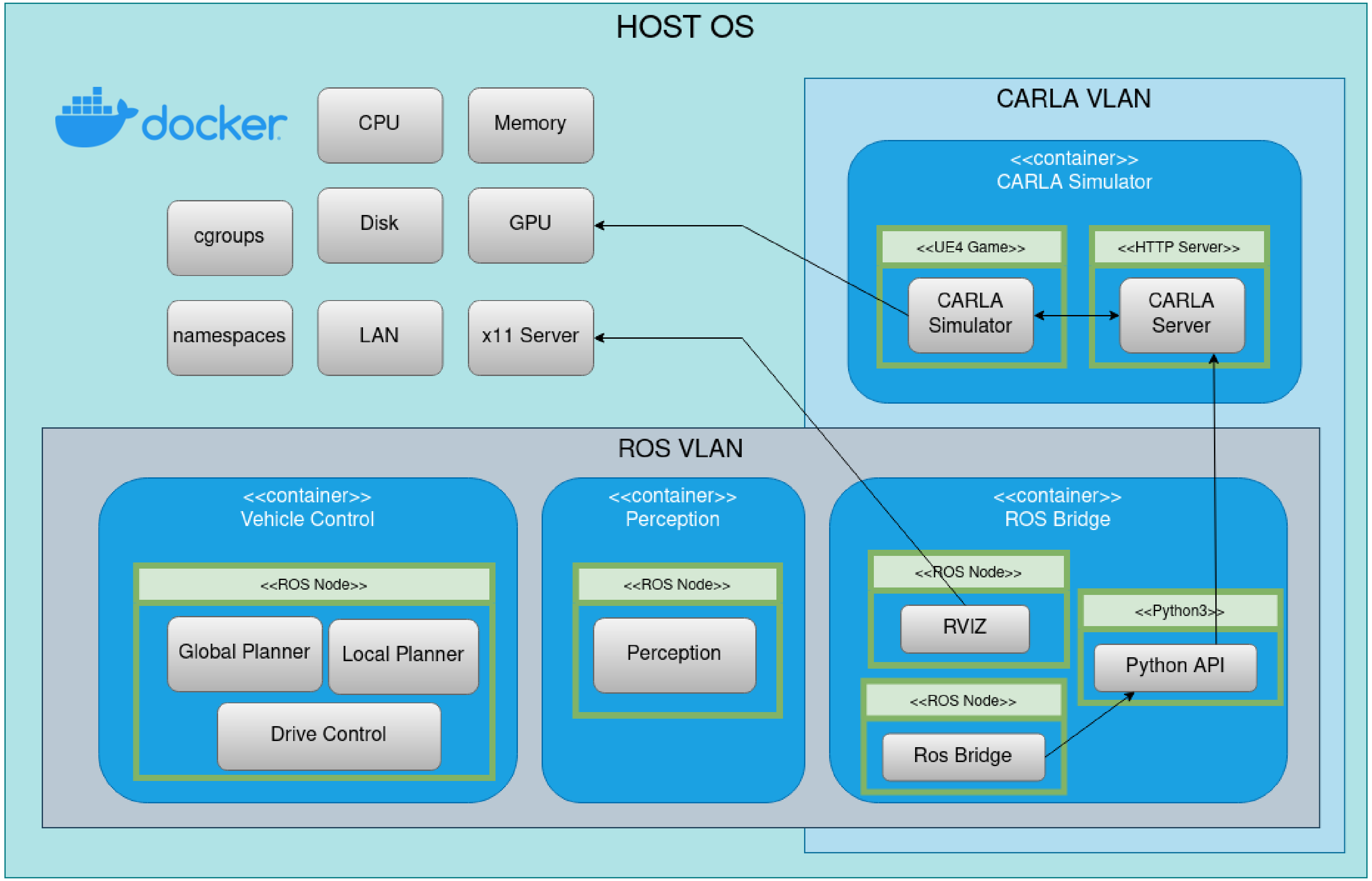
OVERALL ARCHITECTURE

SYSTEM ARCHITECTURE



SYSTEM ARCHITECTURE

Technical System Architecture



SYSTEM ARCHITECTURE

Benefits of Infrastructure as Code in PAF

Collaboration:

- Simple System Setup
  - access to GPU via NVIDIA Docker
  - fully-automated build / launch procedure
- GitHub CI/CD pipelines

Portability / Scalability:

- Local Development
- Remote Performance Testing
- Live-Training on GPU Cluster (A3C RL)

SYSTEM ARCHITECTURE

Architecture Adjustments / Differences

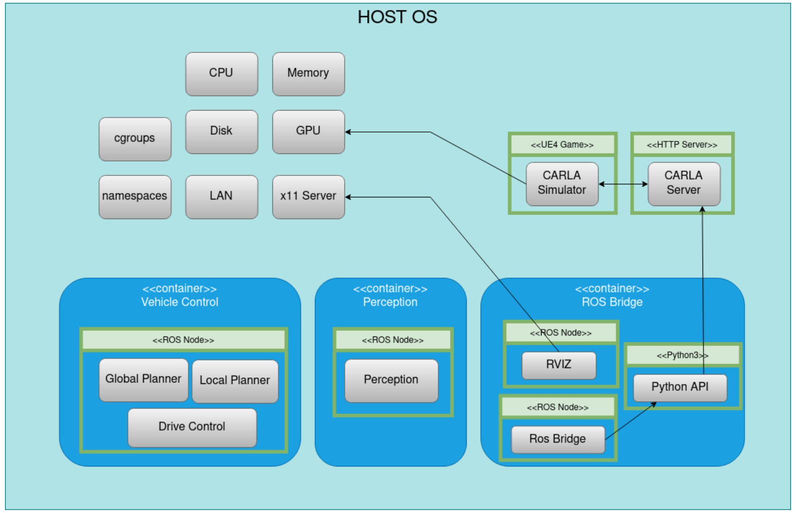
Fuse Global + Local Planning:

- Facilitate route re-planning
- Integrate XODR metadata

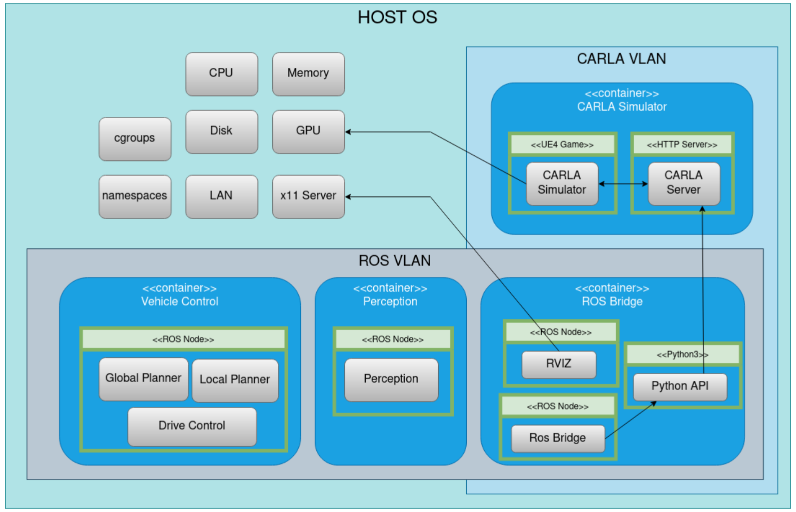
Run Simulator Bare-Metal

- Huge performance gains
- Vehicle Control still abstracted via Docker

Technical System Architecture



Technical System Architecture



02

GLOBAL PLANNING

GLOBAL PLANNING

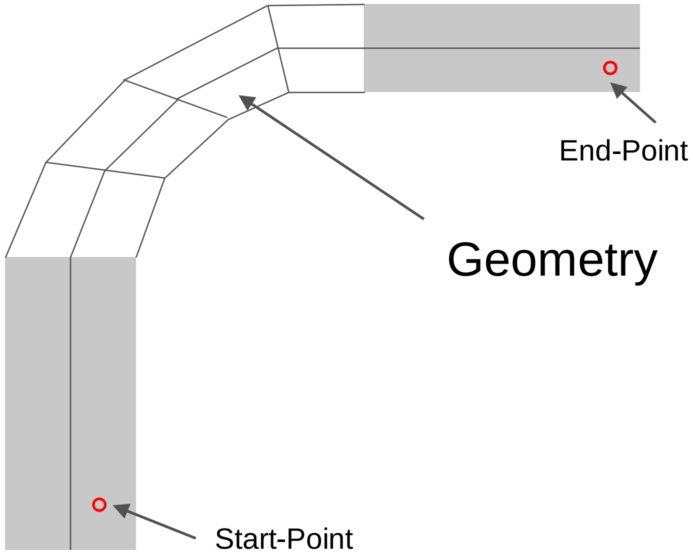
General functionality

- Reading from XODR-Maps
- Generate Path
- Generate Route waypoints
- Read Metadata of XODR
- Return of Annotated waypoints

GLOBAL PLANNING

Generate Route waypoints


```xml
<road name="Road 0" length="4.3410000000000004e+1" id="0" junction="-1">
    <link>
        <predecessor elementType="junction" elementId="498"/>
        <successor elementType="junction" elementId="576"/>
    </link>
    <type s="0.0000000000000000e+0" type="town">
        <speed max="50" unit="mph"/>
    </type>
    <planView>
        <geometry s="0.0000000000000000e+0" x="1.4069525952891570e+2" y="1.9571904130922604e+2
            <line/>
        </geometry>
    </planView>
    <elevationProfile>
        <elevation s="0.0000000000000000e+0" a="0.0000000000000000e+0" b="0.0000000000000000e+
    </elevationProfile>
    <lateralProfile>
        <superelevation s="0.0000000000000000e+0" a="0.0000000000000000e+0" b="0.000000000000
    </lateralProfile>
    <lanes>
        <laneOffset s="0.0000000000000000e+0" a="-3.5000000000000000e+0" b="0.000000000000000
        <laneSection s="0.0000000000000000e+0">
            <left>
                <lane id="6" type="sidewalk" level="false">
                    <width sOffset="0.0000000000000000e+0" a="2.0000000000000000e+0" b="0.0000
                    <roadMark sOffset="0.0000000000000000e+0" type="none" material="standard"
```

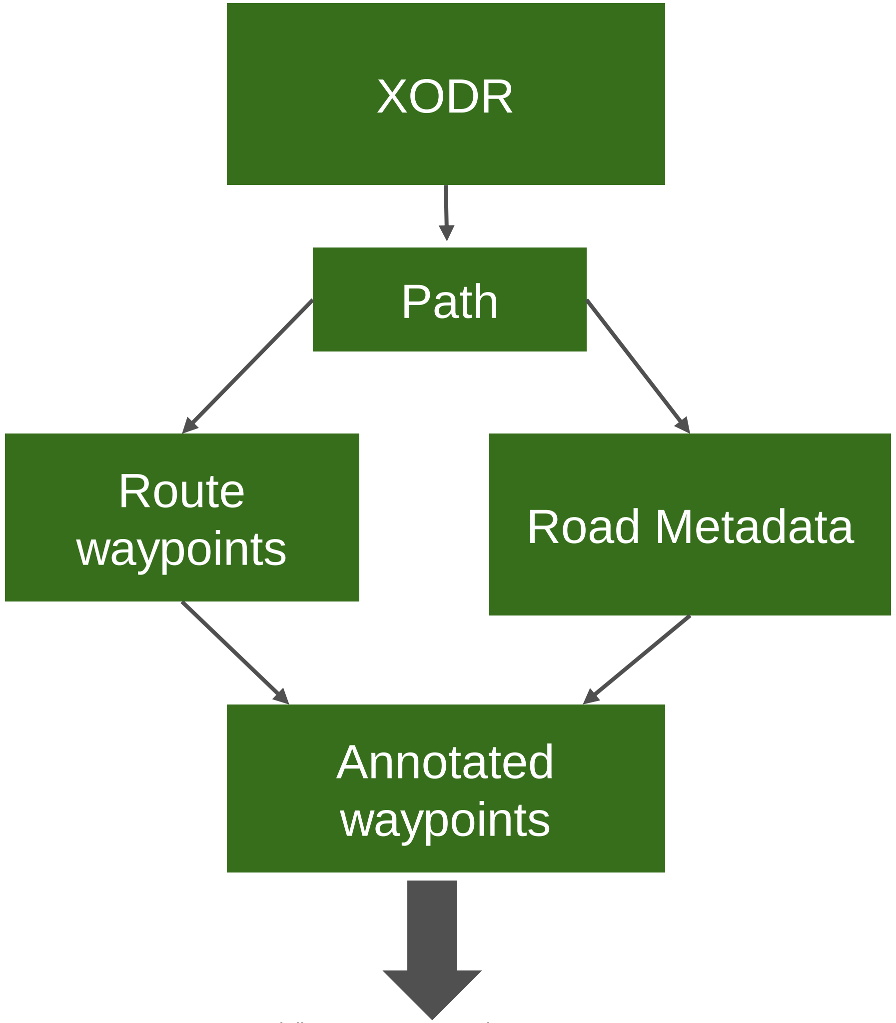



Step by Step:

- Read from XODR and transform it to a logical structure
- Create Matrix with all roads/ lane and their weights
- Find Start and End Point
- Plan the path with dijstra algorithm
- Apend start and end posotion of each Geometry in the path
- Interpolate Route

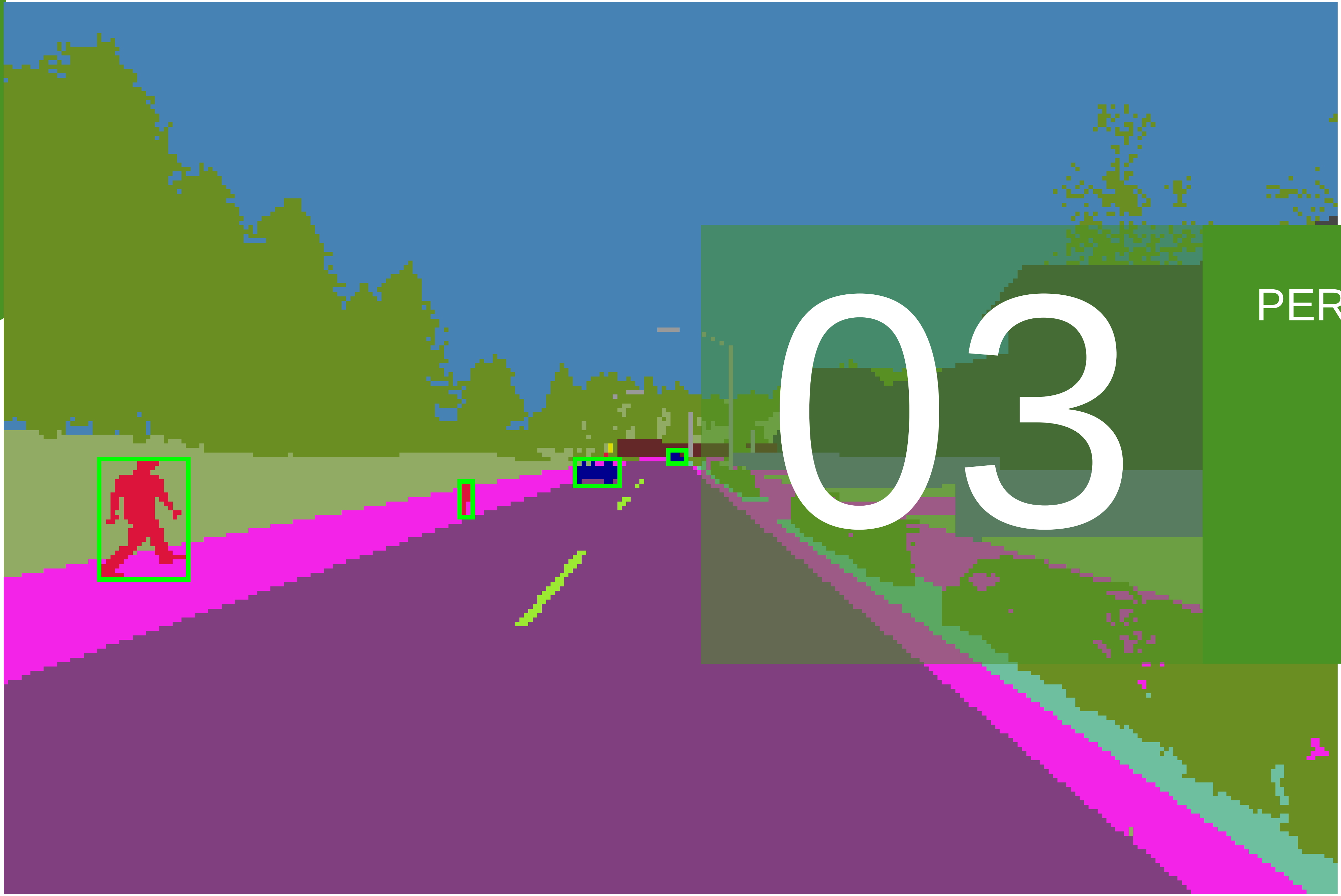
GLOBAL PLANNING

Road Metadata and Return Annotated waypoints



Step by Step:

- Find all Metadata that are along the path
- Connect Metadata with the waypoints
- Annotated waypoints:
  - Position
  - Road ID
  - Planed lane
  - Possible drivable lanes
  - Detected Legal Speed limit
  - Distance to the next Traffic Light
  - End of lane (distance to next Junction)
  - Stop Sign distance
- Theoreticaly we can send more route information if needed



03

PERCEPTION

PERCEPTION

Structure

- Input
  - Semantic-Camera -> semantic image
  - RGB-Camera -> RGB image
  - Depth-Camera -> Depth image

- Detection
  - Traffic Light Detection
  - Object Detection

- Output
  - TrafficLightInfo
  - ObjectInfo

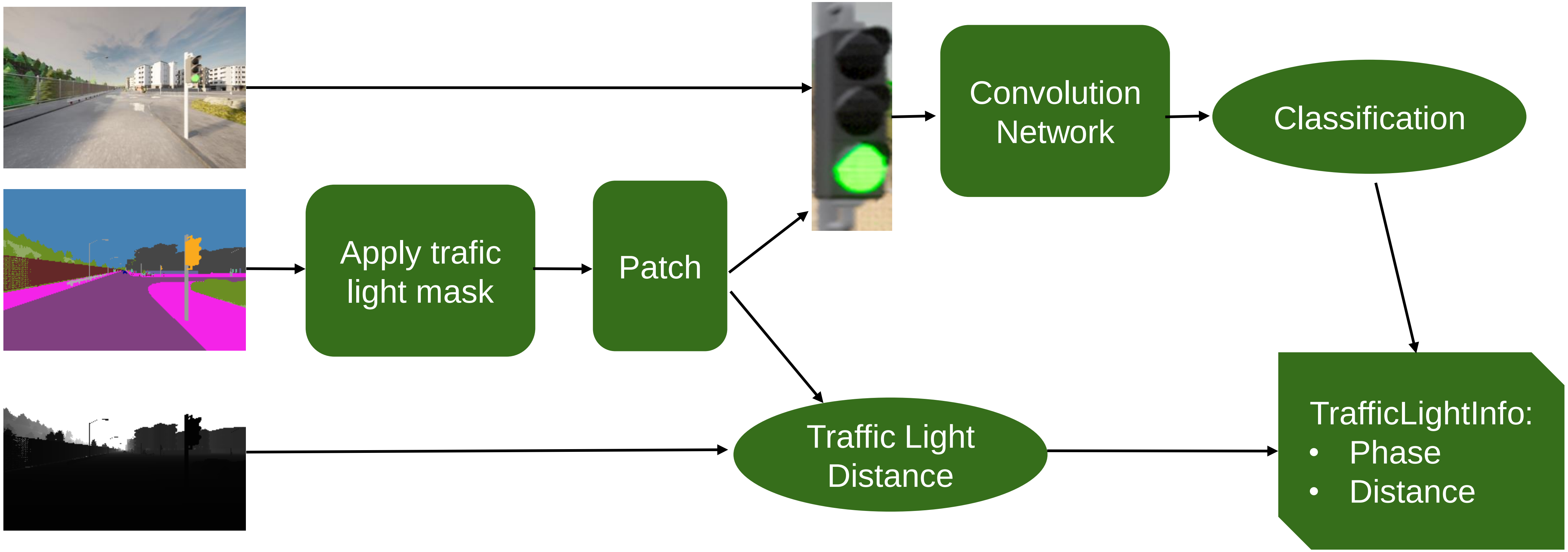
PERCEPTION

Traffic Light Detection

- Use of an artificial neural network --> Convolution Network
  - Input shape [Batch, 32, 32, 3]
  - Output shape [Batch, 4] (Classes: Backside, Green, Red, Yellow)
  - Conv2d with padding same, ReLU activation and four filters
  - Accuracy ~ 99%

PERCEPTION

Traffic Light Detection



PERCEPTION

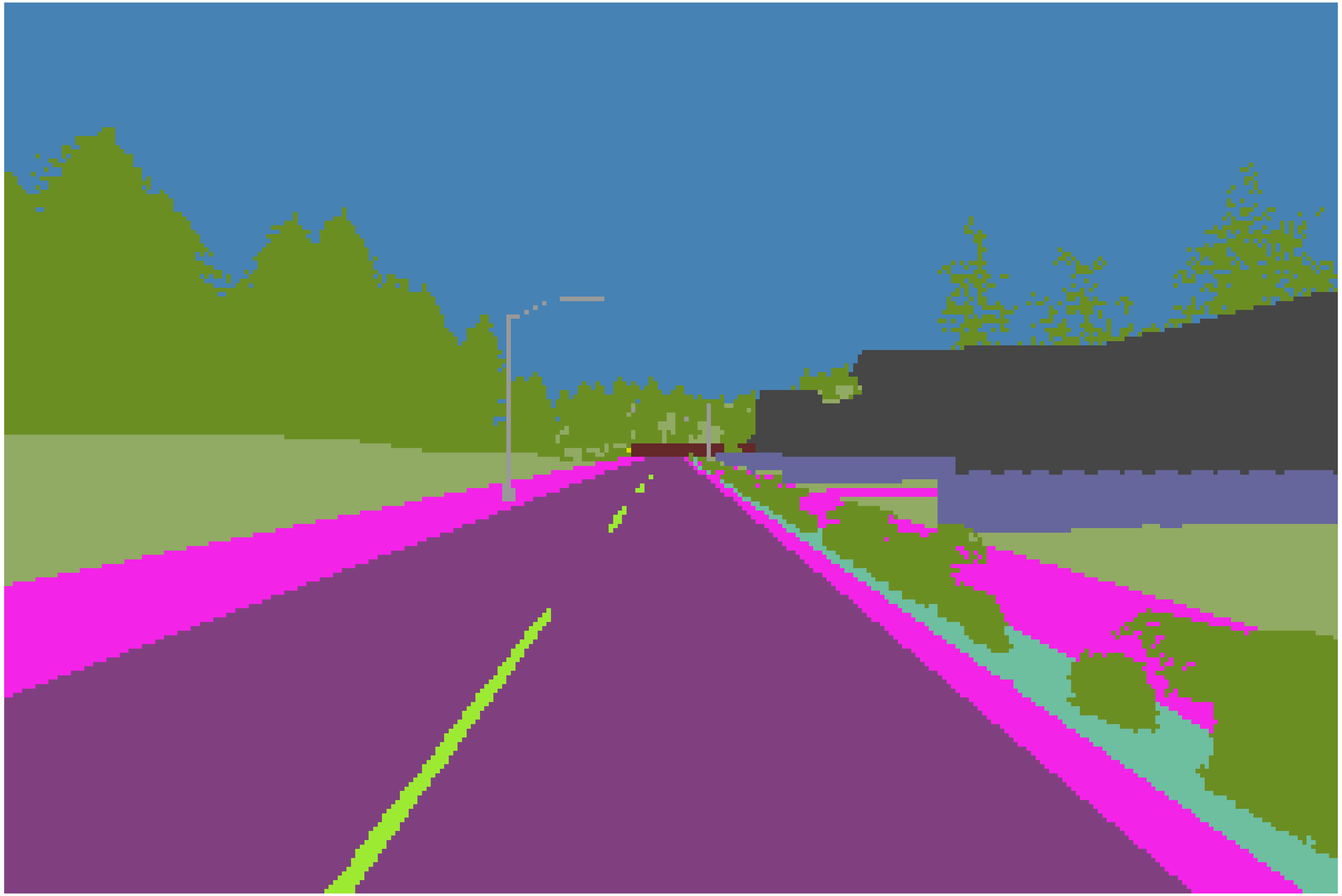
Object Detection

- Object detection on the semantic image with a mask for vehicles and pedestrians
  - ➢ Contours
- Clustering on the depth image separate for vehicles and pedestrians
  - Detect outliers in the contour
  - Detect a invalid contour
  - Use jenks natural breaks algorithm to cluster the contour
  - ➢ Cluster centers with class annotation
- Convert cluster centers to local point cloud
- Track the objects with an euclidean object tracker
- Output:

ObjectInfo:
- Identifier
- Object class
- Relative position

PERCEPTION

Object Detection



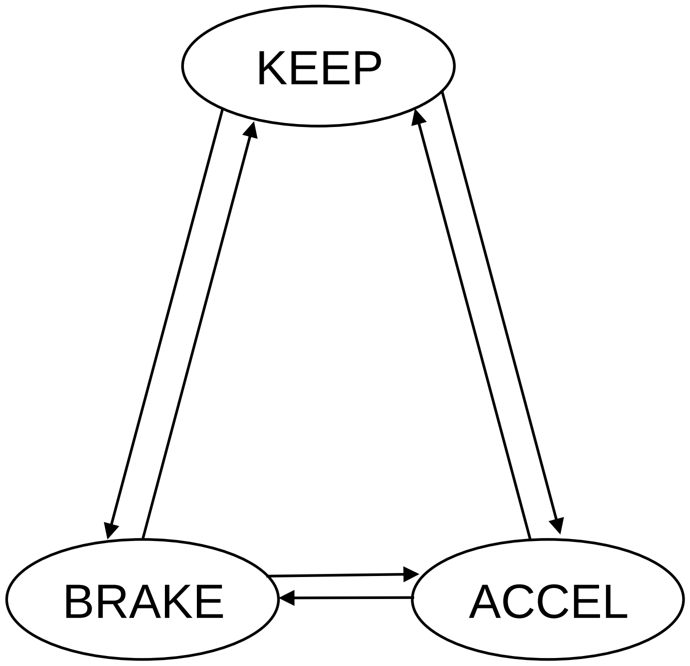
04

VEHICLE CONTROL

VEHICLE CONTROL

Decision Making Component

Speed State Machine



Speed Observation (Sensor Fusion)

SpeedObservation (SO)
is_trajectory_free: Boolean traffic_light_phase: Color dist_next_traffic_light: Meter dist_next_obstacle: Meter dist_next_curve: Meter dist_next_stop: Meter obstacle_speed: m/s curve_target_speed: m/s detected_speed_limit: km/h

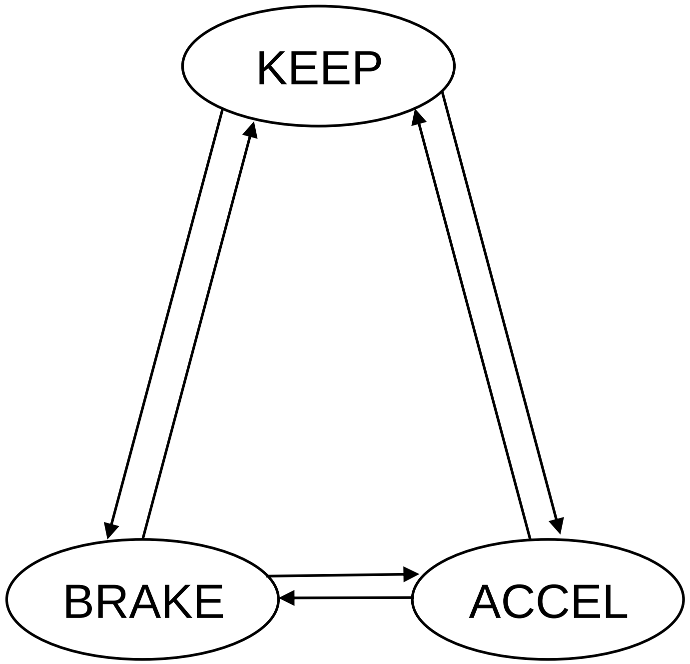
Transition Function

Input \ State	KEEP	BRAKE	ACCEL
SO_1		KEEP	KEEP
SO_2	BRAKE		BRAKE
SO_3	ACCEL	ACCEL	

VEHICLE CONTROL

Speed State Machine

Speed State Machine



Speed Observation (Sensor Fusion)

SpeedObservation (SO)

is_trajectory_free: false
traffic_light_phase: Green
dist_next_traffic_light: 100m
dist_next_obstacle: 15m
dist_next_curve: 60m
dist_next_stop: 50m
obstacle_speed: 0m/s
curve_target_speed: 10m/s
speed_limit: 50km/h

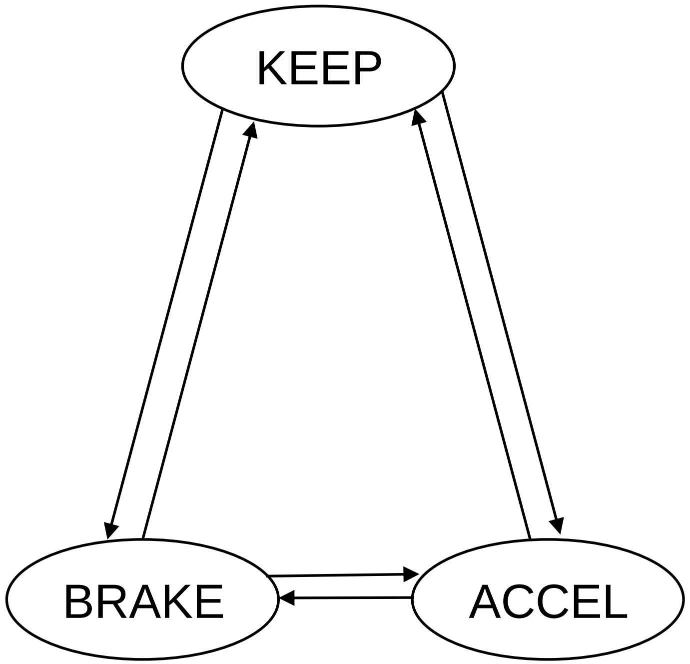
Transition Function

Input \ State	KEEP	BRAKE	ACCEL
SO_1		KEEP	KEEP
SO_2	BRAKE		BRAKE
SO_3	ACCEL	ACCEL	

VEHICLE CONTROL

Speed State Machine

Speed State Machine



Speed Observation (Sensor Fusion)

SpeedObservation (SO)

is_trajectory_free: false
traffic_light_phase: Green
dist_next_traffic_light: 100m
dist_next_obstacle: 15m
dist_next_curve: 60m
dist_next_stop: 50m
obstacle_speed: 0m/s
curve_target_speed: 10m/s
speed_limit: 50km/h

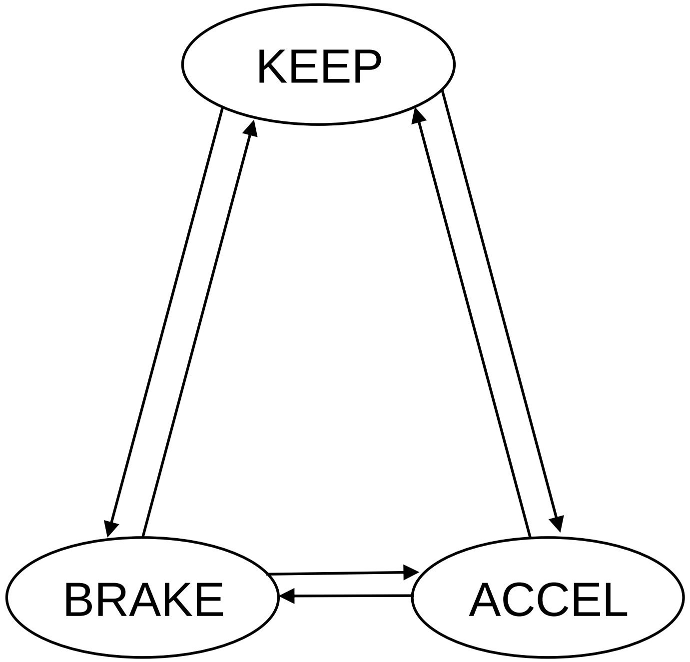
Transition Function

Input \ State	KEEP	BRAKE	ACCEL
SO_1		KEEP	KEEP
SO_2	BRAKE		BRAKE
SO_3	ACCEL	ACCEL	

VEHICLE CONTROL

Speed State Machine

Speed State Machine



Speed Observation (Sensor Fusion)

SpeedObservation (SO)

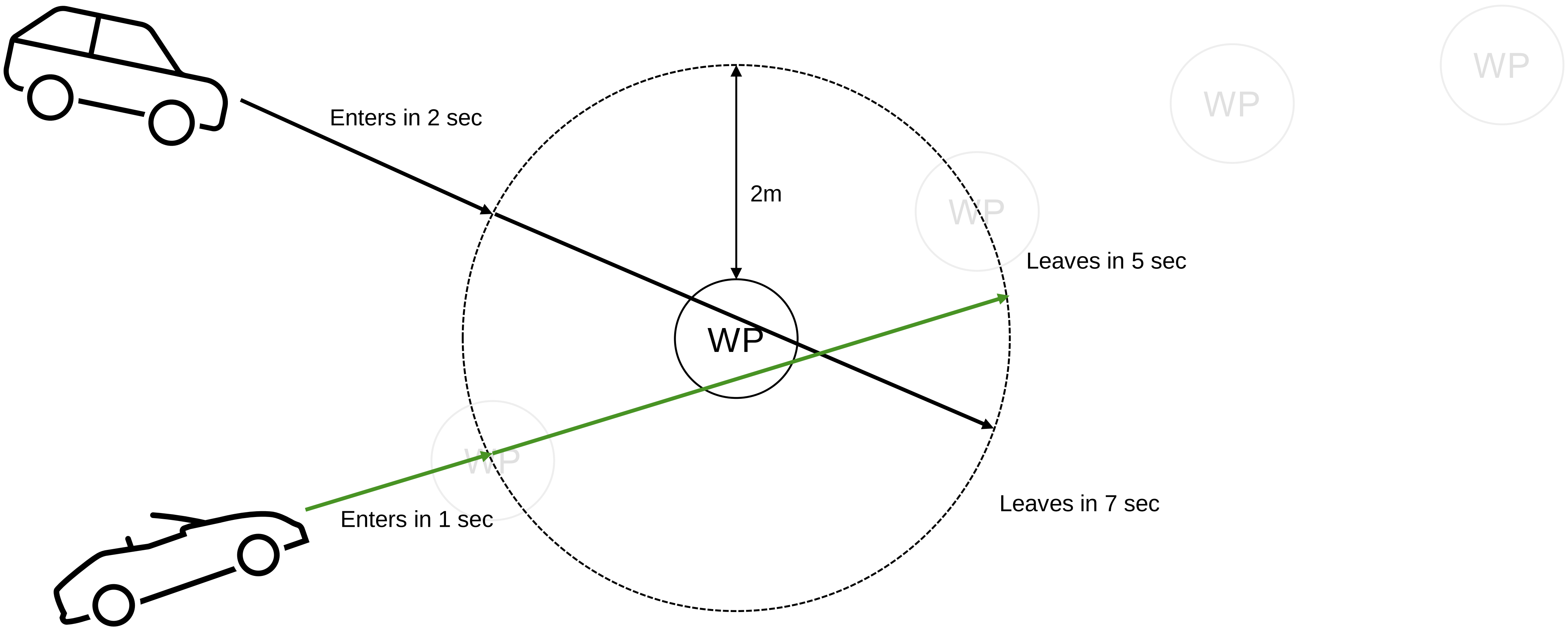
is_trajectory_free: false
traffic_light_phase: Green
dist_next_traffic_light: 100m
dist_next_obstacle: 15m
dist_next_curve: 60m
dist_next_stop: 50m
obstacle_speed: 0m/s
curve_target_speed: 10m/s
speed_limit: 50km/h

Transition Function

Input \ State	KEEP	BRAKE	ACCEL
SO_1		KEEP	KEEP
SO_2	BRAKE		BRAKE
SO_3	ACCEL	ACCEL	

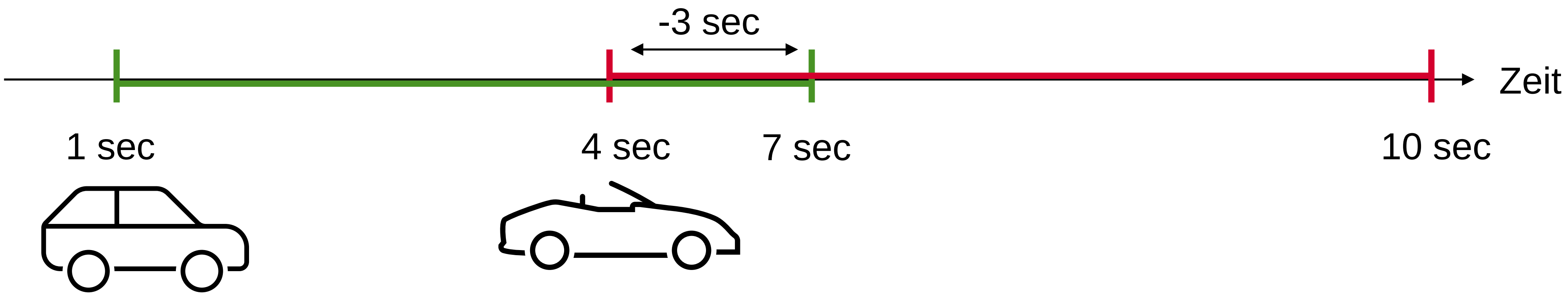
VEHICLE CONTROL

Zone Clearance Time



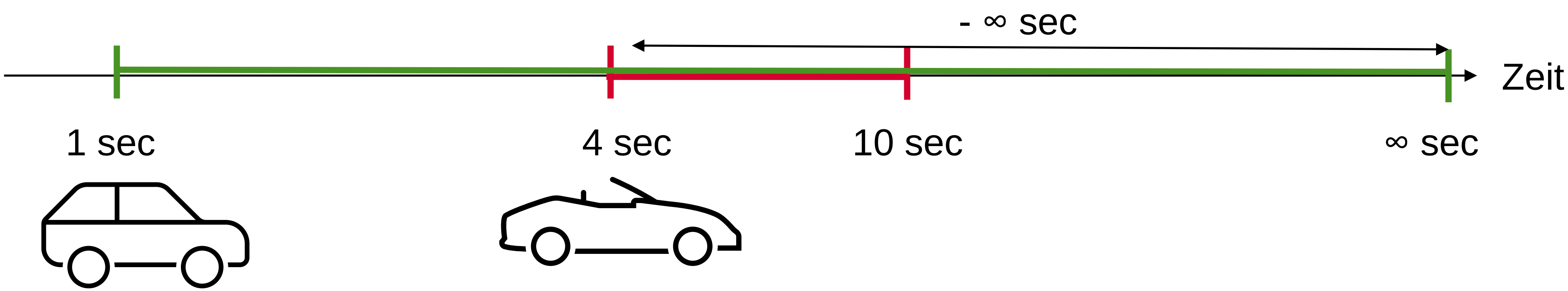
VEHICLE CONTROL

Zone Clearance Time



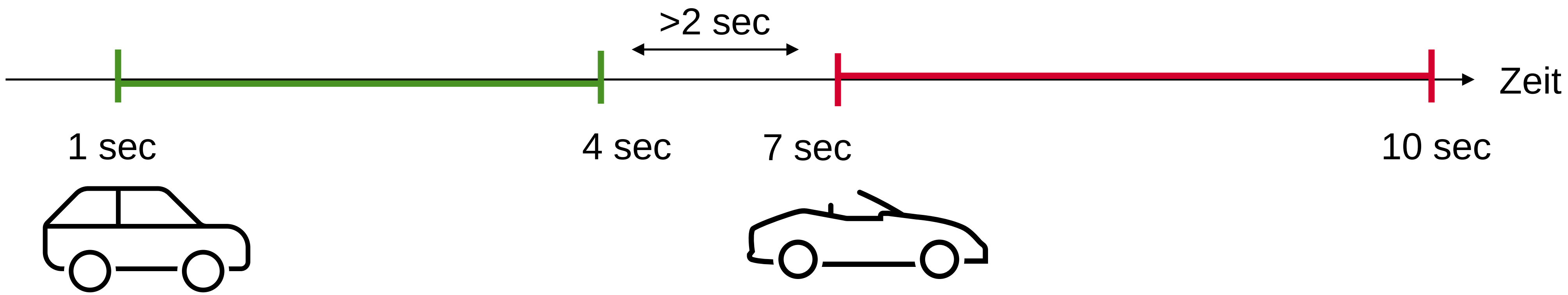
VEHICLE CONTROL

Zone Clearance Time



VEHICLE CONTROL

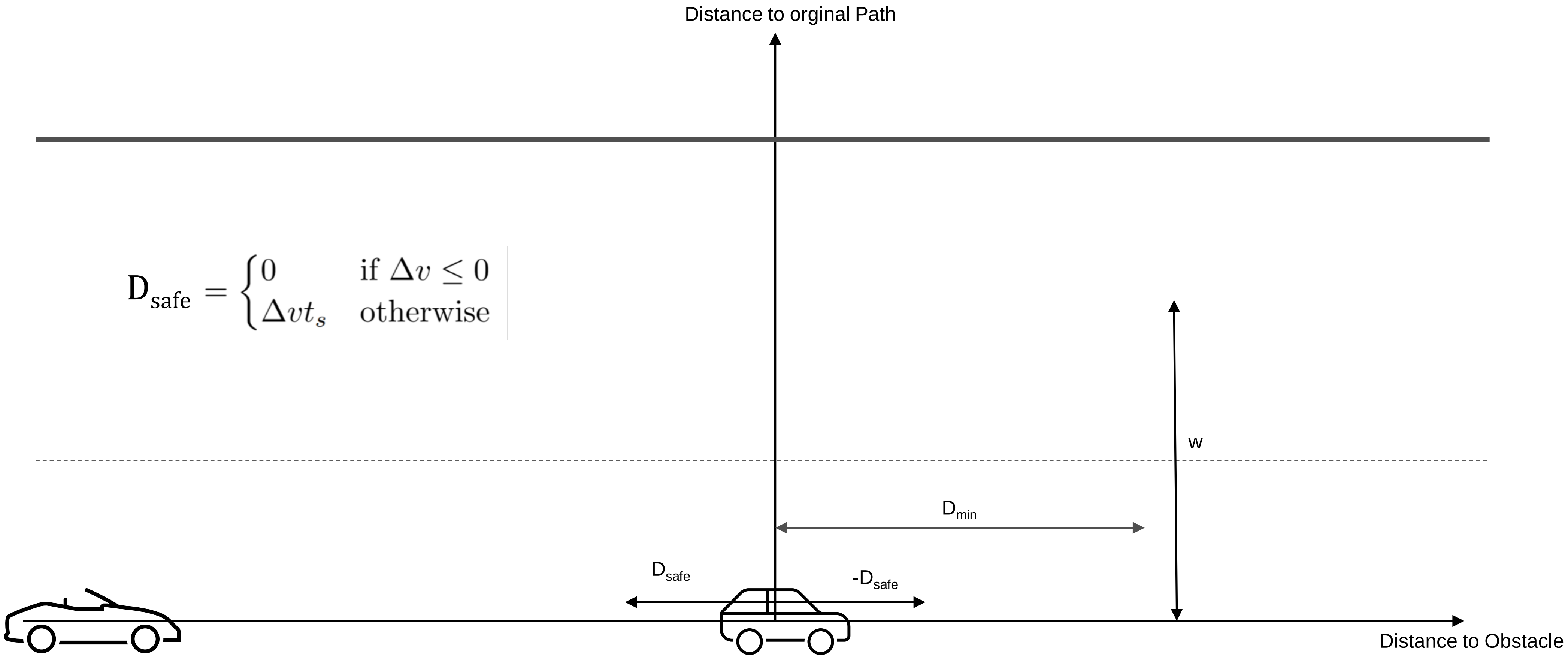
Zone Clearance Time



VEHICLE CONTROL

Overtaking

$$D_{\text{safe}} = \begin{cases} 0 & \text{if } \Delta v \leq 0 \\ \Delta v t_s & \text{otherwise} \end{cases}$$

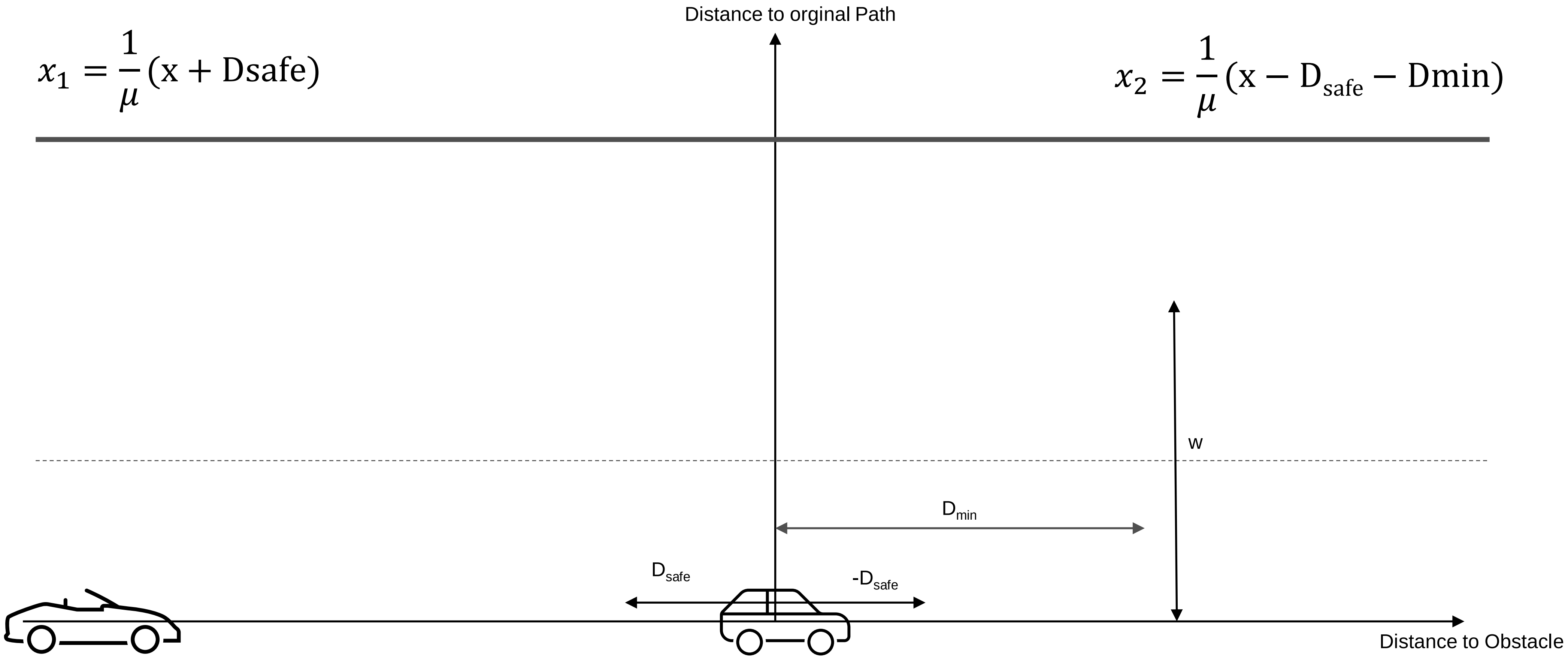


VEHICLE CONTROL

Overtaking

$$x_1 = \frac{1}{\mu}(\mathrm{x} + \mathrm{Dsafe})$$

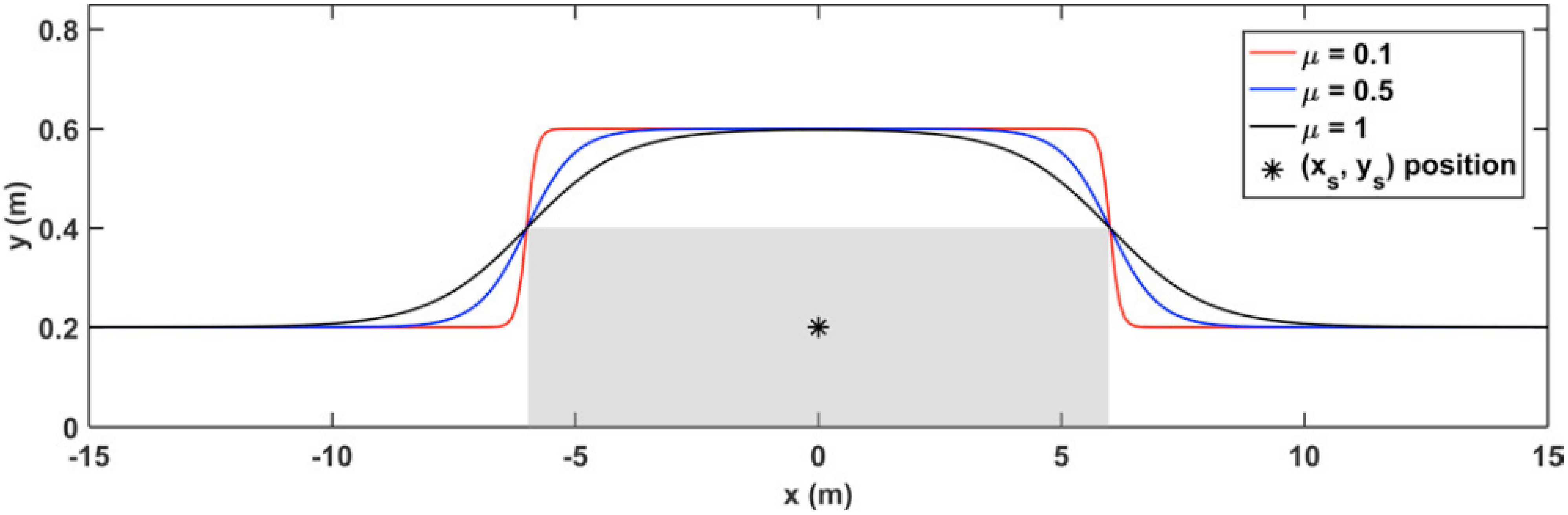
$$x_2 = \frac{1}{\mu}(\mathrm{x} - \mathrm{D_{safe}} - \mathrm{Dmin})$$



VEHICLE CONTROL

Overtaking

$$y_1 = \frac{w}{1 + e^{-x_1}} \qquad y_2 = \frac{w}{1 + e^{-x_2}} \qquad y = y_1 + y_2 + yl \qquad y_l = 0$$

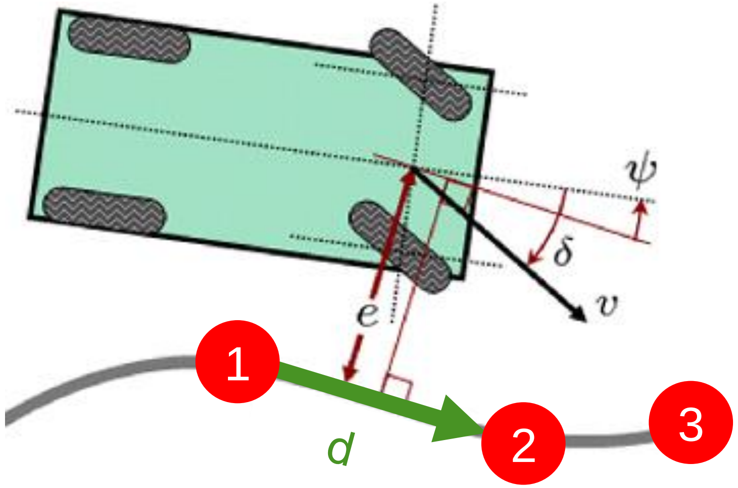


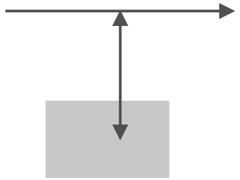
$$D_{\text{safe}} = 6, w = 0.4, yl = 0.2$$

VEHICLE CONTROL

Stanley Controller - Approach

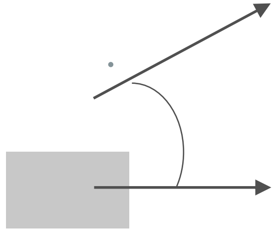
e = distance trajectory vehicle
v = velocity vehicle
d = distance of interpolation
(adopted to vehicle velocity)





crosstrack error

$$\delta(t) = \tan^{-1}\left(\frac{ke(t)}{v_f(t)}\right)$$



heading error:

$$\psi(t)$$

\+ steering angle

VEHICLE CONTROL

Stanley Controller - Challenges

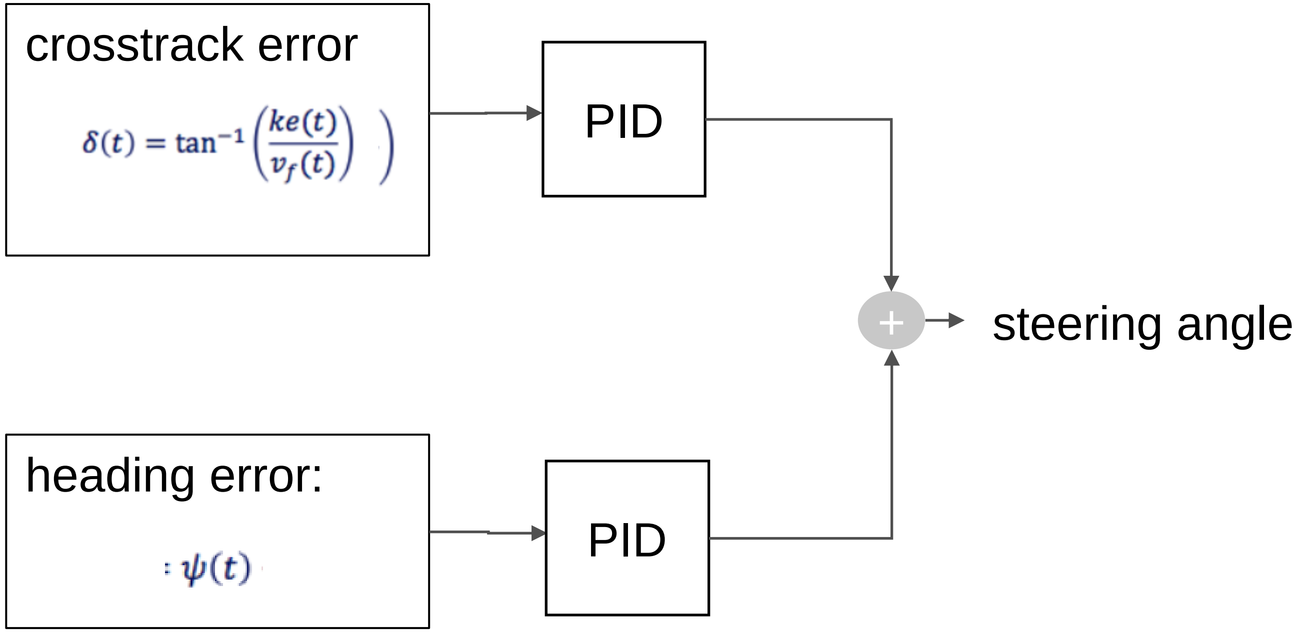
1. Low refresh rate of the controller and the GPS data
    1. refresh rate controller loop at ~40Hz
    2. irregular odometry (GPS) messages at ~20Hz

$\Rightarrow$ at 130 km/h: refresh steering angle: every ~1 meter of travelled distance
new GPS data: every ~2 meter of travelled distance

high requirements on the controller

VEHICLE CONTROL

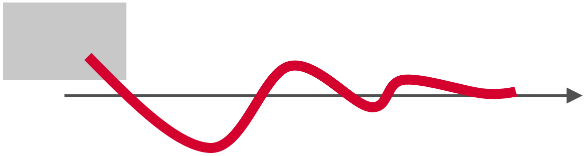
Stanley Controller – Approach - PID



Manual adjustment of the parameters

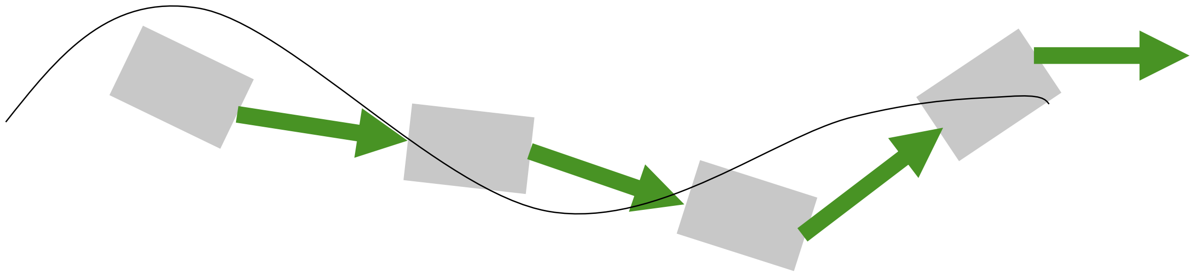
1. high I-Value prevent oscillating
2. high D-Value fast adoption of steering angle

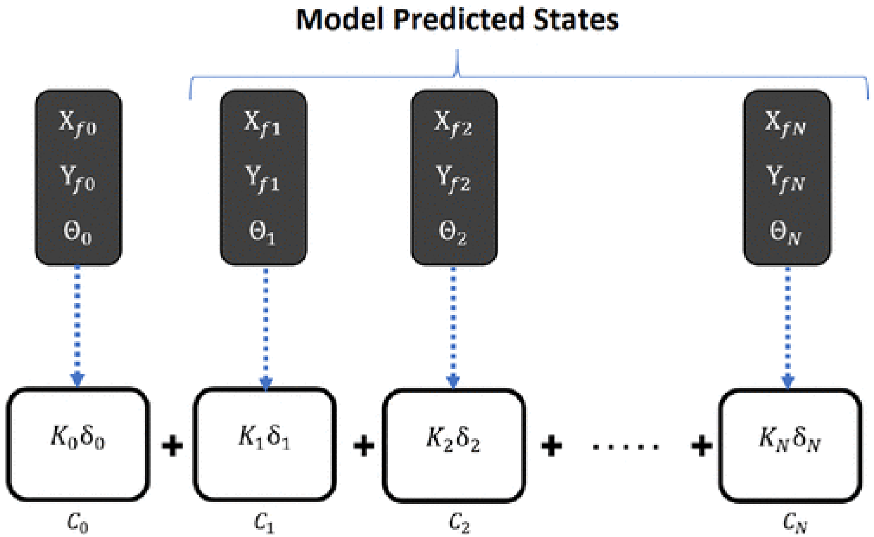
reset of the I-proportion at low speed



VEHICLE CONTROL

Predictive Stanley Controller

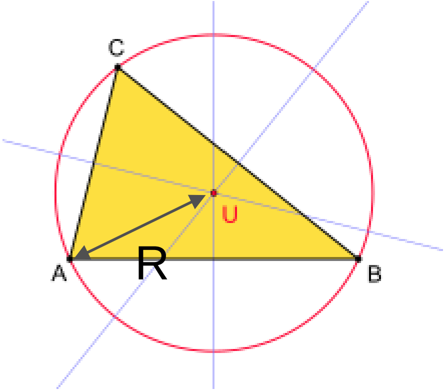




VEHICLE CONTROL

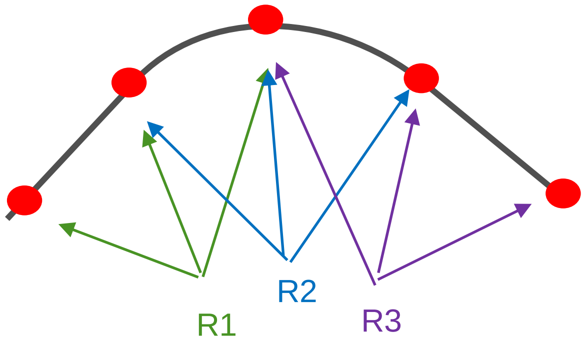
Curve detection + speed control

Calculating the radii of the trajectory by calculating the perimeter of a triangle

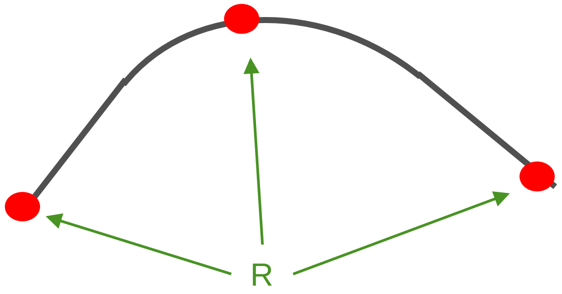


$$R = \frac{abc}{4A}.$$

Detect curve bounds



Calculating the maximal curve speed



$$v = \sqrt{\mu_{Haft} \cdot r \cdot g}$$

VEHICLE CONTROL

Traffic light's handling / American TL

To reduce classification time and effort

- Fixed size for traffic lights
- Fixed zones

05

SUMMARY AND OUTLOOK

SUMMARY AND OUTLOOK

Summary

- Vehicle contains basic components
  - Path Planning, Image Processing, Vehicle Control
- Lessons Learned
  - Implementing everything from scratch is very time consuming
    - More research is important
  - Pair-Programming enhances communication
  - Working across all domains grants everybody basic understanding of autonomous driving cars
  - Testing code is very important as early as possible (Test-Driven-Development)
  - Reduce dependencies and implement key features early

SUMMARY AND OUTLOOK

Outlook

- Online-Training with distributed reinforcement learning (A3C)
- Replace determinism with more propabilistic approaches
- Implement more maneuvers / traffic rules
- Enhance perception layer to get 360 degree vision

GEFÖRDERT VOM
Bundesministerium
für Bildung
und Forschung

Universität Augsburg
Fakultät für Angewandte
Informatik

Vielen Dank für Ihre Aufmerksamkeit

Joshua Berghoff – joshua.berghoff@student.uni-augsburg.de

Axel Böll - axel.boell@student.uni-augsburg.de

Pavlo Mospan - pavlo.mospan@student.uni-augsburg.de

Johannes Stoljar – johannes.stoljar@student.uni-augsburg.de

Daniel Sturm - daniel.Sturm@student.uni-augsburg.de

Marco Tröster - @student.uni-augsburg.de

Universität Augsburg

www.uni-augsburg.de