Evaluating Location Compliance Approaches for Automated Road Vehicles

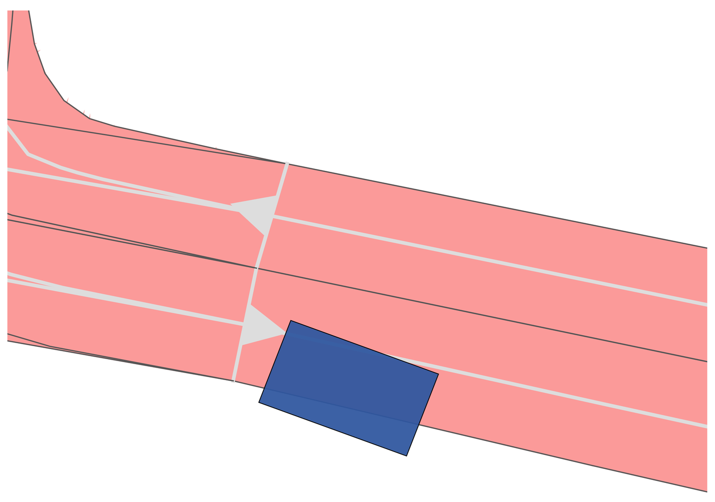
Alexander Zhu, Stefanie Manzinger, and Matthias Althoff
Technische Universität München

Problem Statement

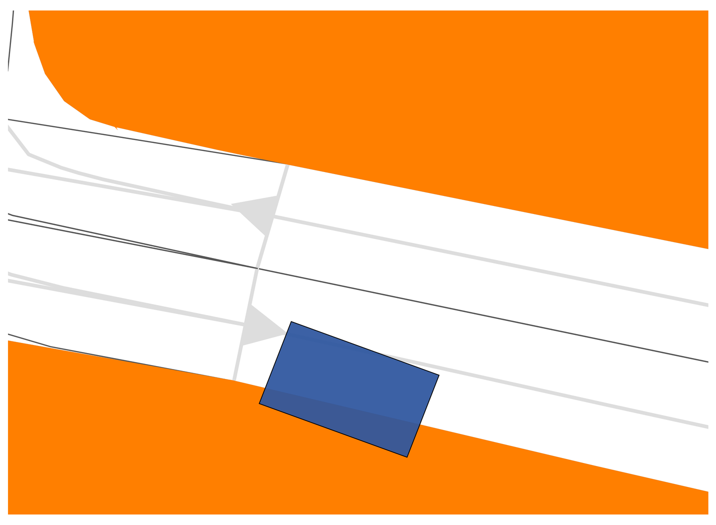
Problem Statement: Automated road vehicles should be prevented from leaving the boundaries of the road. We refer to **location compliance** as an allowed translational and rotational positioning of the vehicle on a two dimensional road map composed of lanelets.

Definition of Location Compliance: A vehicle is location compliant iff its *occupancy* $\mathcal{E}$ is in the *allowed region* $\mathcal{A}$:

$$\mathcal{E} \subseteq \mathcal{A} \Leftrightarrow \mathcal{E} \cap \mathcal{A}^C = \emptyset.$$



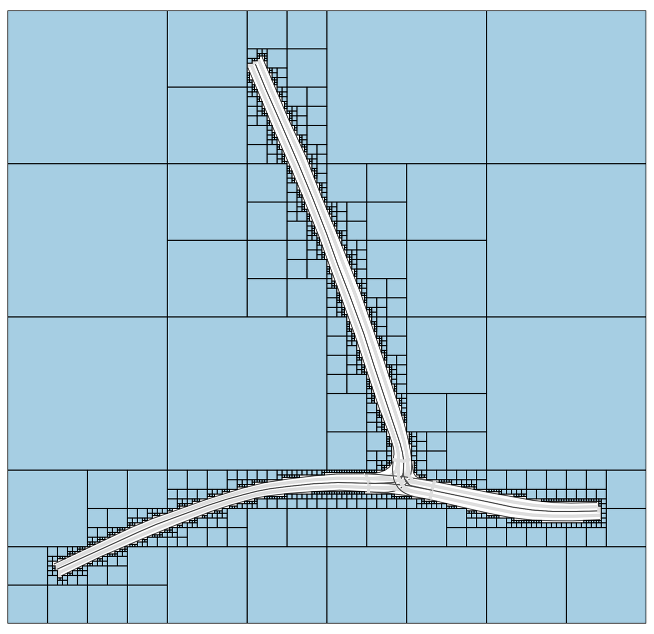
Enclosure checking: Is the blue vehicle enclosed in the red allowed region? ($\mathcal{E} \subseteq \mathcal{A}$?)



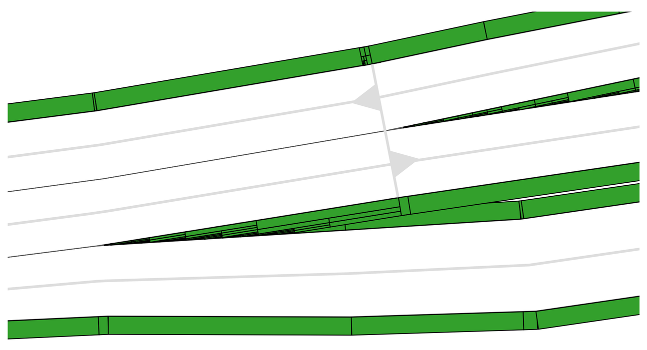
Boundary collision approach: Does the blue vehicle collide with the orange forbidden region? ($\mathcal{E} \cap \mathcal{A}^C = \emptyset$?)

Boundary Collision

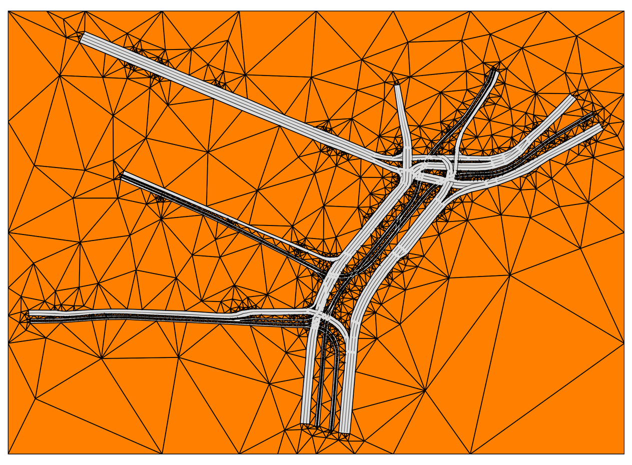
In order to determine location compliance with collision detection, we need to construct the *forbidden region* $\mathcal{A}^C$.
We examine the following approaches for obtaining $\mathcal{A}^C$:



Quadtree: The map is divided in a recursive quadtree algorithm. This approach is simple, but there are inaccuracies around the road border, due to the use of axis-aligned rectangles.



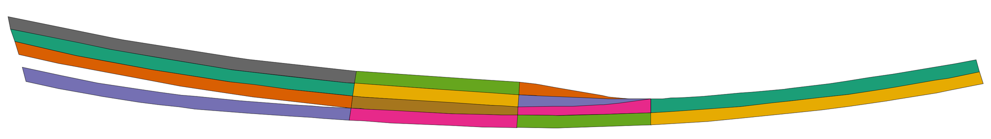
Shell Approach: Oriented rectangles are constructed around the driveable area. There are fewer inaccuracies, as the obstacles are aligned with the curvature of the road.



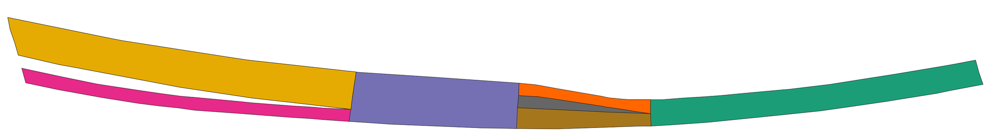
Triangulation: The forbidden area is expressed with a triangle mesh, which is obtained by performing Constrained Delaunay Triangulation (CDT). There are no inherent inaccuracies in this representation.

Enclosure Checking

- **Polygon model:** The allowed area $\mathcal{A}$ is modeled with a set of polygons $\mathcal{L} = \bigcup_i \mathcal{L}_i$.
- For an efficient computation, there should be few polygons in $\mathcal{L}$ and each polygon $\mathcal{L}_i$ should have a low number of points. We examine two choices for polygon representations:



The road map, which is composed of **lanelets**, is expressed as a polygon $\mathcal{L}_i$.



Laterally adjacent lanelets are combined to so-called **lane sections**, which each correspond to a single polygon.

- Location compliance can be checked with polygon difference:

$$\mathcal{E} \subseteq \mathcal{L} \Leftrightarrow \mathcal{E} \setminus \mathcal{L} = \emptyset$$

Main Results

- We propose various approaches to determine location compliance *automatically from maps composed of lanelets.*
- **Precision:** Currently used approaches approximate the road boundary with rectangles. However, our triangulation and enclosure approaches model $\mathcal{A}$ exactly.
- **Efficient computation:** From our measurements, the *triangulation* is the most efficient of the collision methods. The *polygon enclosure with lane sections* is slightly faster compared to it, but further work on comparing them should be done.

Method	GER B471	GER Ffb 2	GER Muc1a	NGSIM US101
Quadtree	1.80	0.75	4.52	1.13
Shell	1.24	0.29	5.38	**0.19**
Triangulation	**0.31**	**0.10**	**0.85**	0.23
Lanelets	0.29	0.15	1.04	0.36
Lane sections	**0.23**	**0.08**	**0.73**	**0.11**

- Computation times in seconds for 10,000 location compliance checks with random vehicle poses.
- At the top are the boundary collision approaches and at the bottom are the enclosure methods.
- We compare the results on three maps from the Commonroad project and one made for the purpose of this study (GER B471).

{alexander.zhu, stefanie.manzinger, althoff}@tum.de

We gratefully acknowledge partial financial support by the Deutsche Forschungsgemeinschaft (German Research Foundation) within the Priority Program SPP 1835 "Cooperative Interacting Automobiles" (grant number: AL 1185/4-1) and the European Commission project interACT under grant number 723395.

commonroad.in.tum.de