



autoware.universe
Dynamic Drivable Area

Maxime CLEMENT

Drivable Area (DA) in Autoware

Overview.

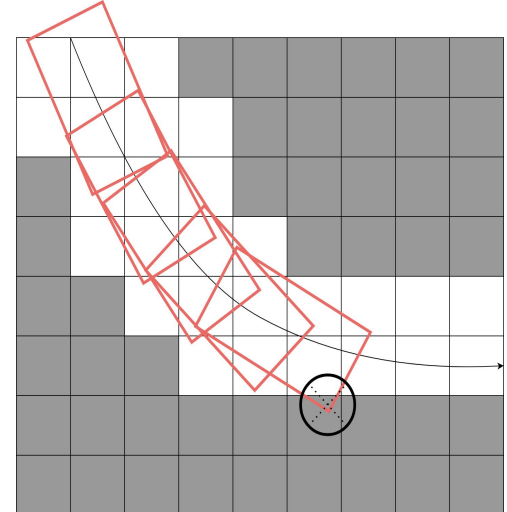
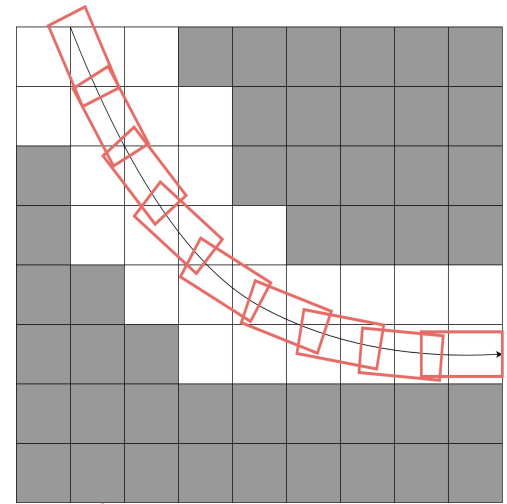
- Generated by the [behavior_path_planner](#).
- Contained in the [Path message](#) as an [OccupancyGrid](#).
- Binary occupancy grid: indicate where the vehicle can drive.

Usage.

- [obstacle_avoidance_planner](#) plans the motion of the vehicle inside the drivable area.
- If the resulting trajectory goes out of the drivable area, a stop point is inserted in the trajectory.

Current issue for the Bus ODD.

- Drivable area can be too narrow → the `obstacle_avoidance_planner` cannot plan a good trajectory.
- Existing solutions are not satisfactory.



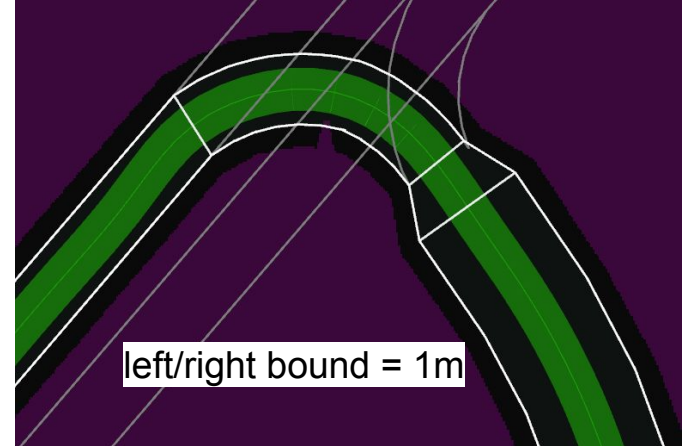
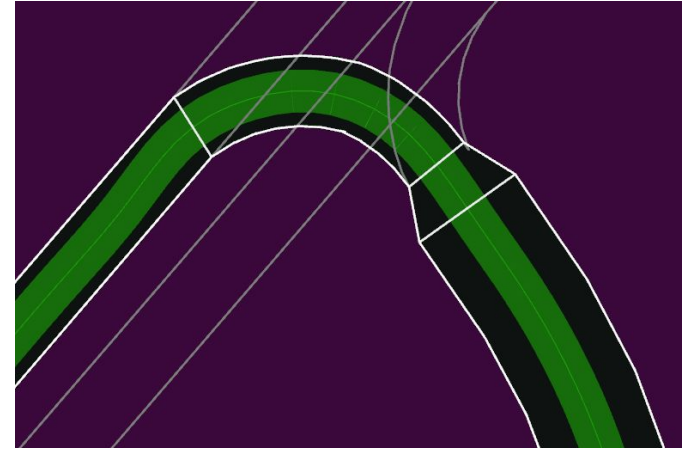
Drivable Area (DA) Generation in Autoware

[behavior_path_planner](#)

- Calculates the drivable area from the lanelet map.
 - In general, left and right linestrings of the route lanelet.
- The space between the left and right linestrings are drivable.
- The space outside of the left and right linestrings are not drivable.

Parameters to expand the drivable area

- left/right bound expansion distance.
- linestring types that cannot be expanded.
 - Currently 'road_border' by default.
 - Will make it a parameter.
- Parameters defined independently for each module.



Current Solutions to the “Narrow DA” issue

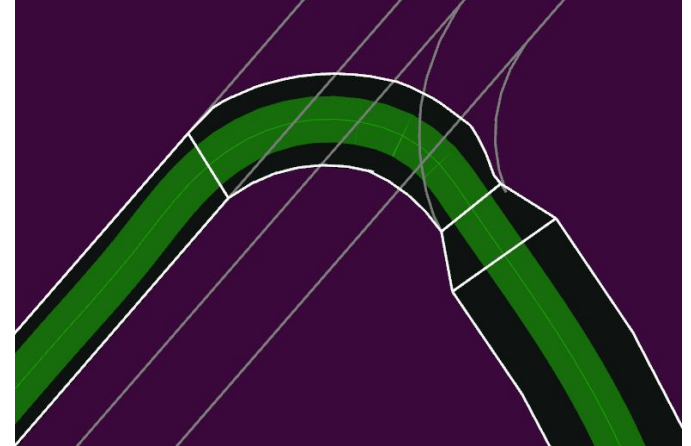
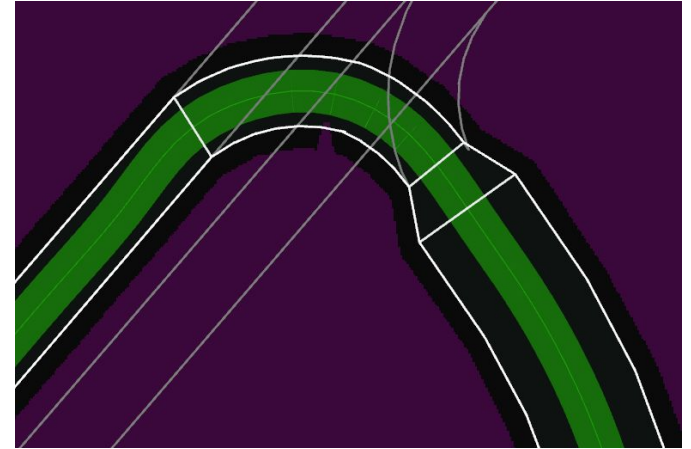
Set left/right bound parameters.

- Set [behavior_path_planner](#) parameters to expand the DA to the left/right by a fixed distance.
- **Good**: simple change of parameter; can be adjusted for each vehicle.
- **Bad**: always on, even when not necessary.

Update lanelet map

- Modify the lanelet map where the DA is too narrow so that it becomes large enough to drive through.
- **Good**: drivable area is only made bigger where necessary.
- **Bad**: breaks the correspondence between the map and the real world; make the map dependent on the vehicle type.

Moreover, these solutions are “static” and may no longer work in the presence of dynamic obstacles, or if a turn is approached with an unexpected angle.



Narrow Drivable Area Issue

Issue: the trajectory footprint does not fit inside the drivable area.
Possible causes.

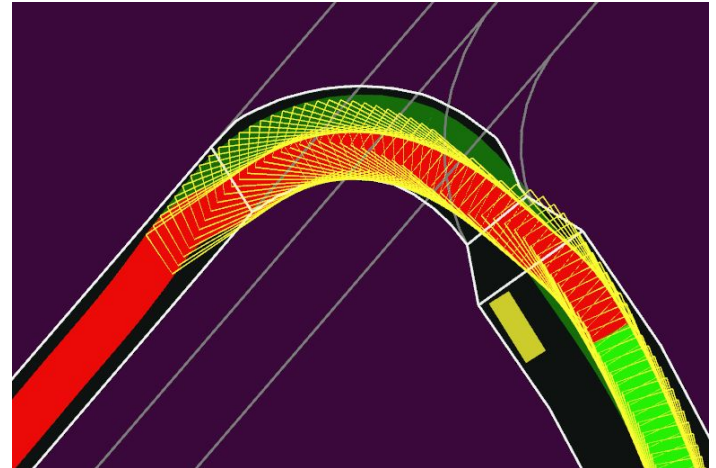
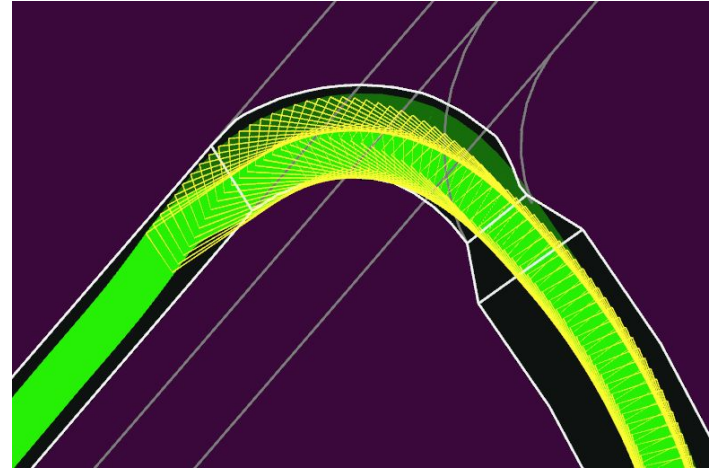
- Vehicle footprint is too large.
- Lane is too narrow.
- Motion planning has to deviate from the centerline.
 - Obstacles.
 - Constraints on steering angle, steering rate.
 - ...

Even by making the drivable area larger, the issue can still occur depending on the motion planner.

Possible design issue.

- Behavior planning generates a reference path and the drivable area.
- Motion planning generates a trajectory around the reference path that **must stay inside the drivable area**.
- → motion planning can be over constrained by the drivable area, with no possible solution to the problem.

The following proposal does not attempt to solve the design issue.

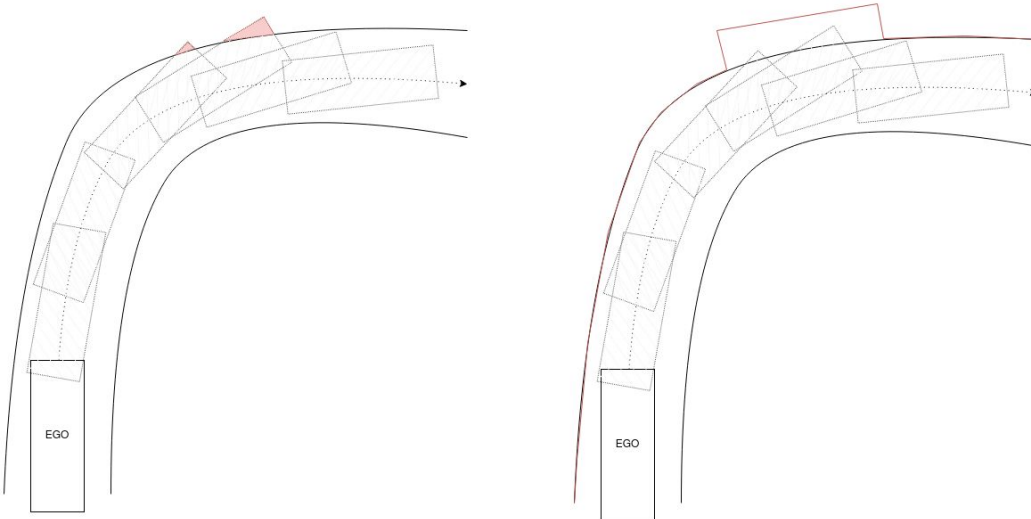


Proposal: Dynamic DA

Main idea: create a footprint polygon from the path, expand the DA around that polygon.

Possible extra features.

- Inflate the polygon for extra DA expansion.
- Not crossing linestrings of specific types (e.g., 'road_border').
- Not crossing predicted paths of dynamic objects.



Pros.

- Only expand DA when necessary.
- Dynamically based on the behavior path (behavior_path_planner).
- Account for the vehicle dimensions.
- Easy to implement (mostly just some geometry operations).

Cons.

- May be computationally expensive.
- May still “fail” in case of complex maneuvers from the motion planner (obstacle_avoidance_planner).

Proposal: Implementation Details

This is the initial plan for implementing the proposal.

Basic functionality.

- Footprint polygon.
 - Calculated from the vehicle dimensions at each path point.
 - [boost::geometry::union](#) to combine the polygons.
- Drivable area.
 - Input coming from the `behavior_path_planner`.
 - [Iterate](#) over the occupancy grid cells inside the footprint polygon.
 - Set the value of each cell to “drivable”.

Extra functionalities.

- Inflated footprint polygon.
 - length & width offsets added to the ego dimensions.
- Not crossing linestrings of specific types.
 - [boost::geometry::difference](#).
- Not crossing the dynamic obstacles' predicted paths.
 - Footprint polygon for the dynamic obstacles.
 - [boost::geometry::difference](#).