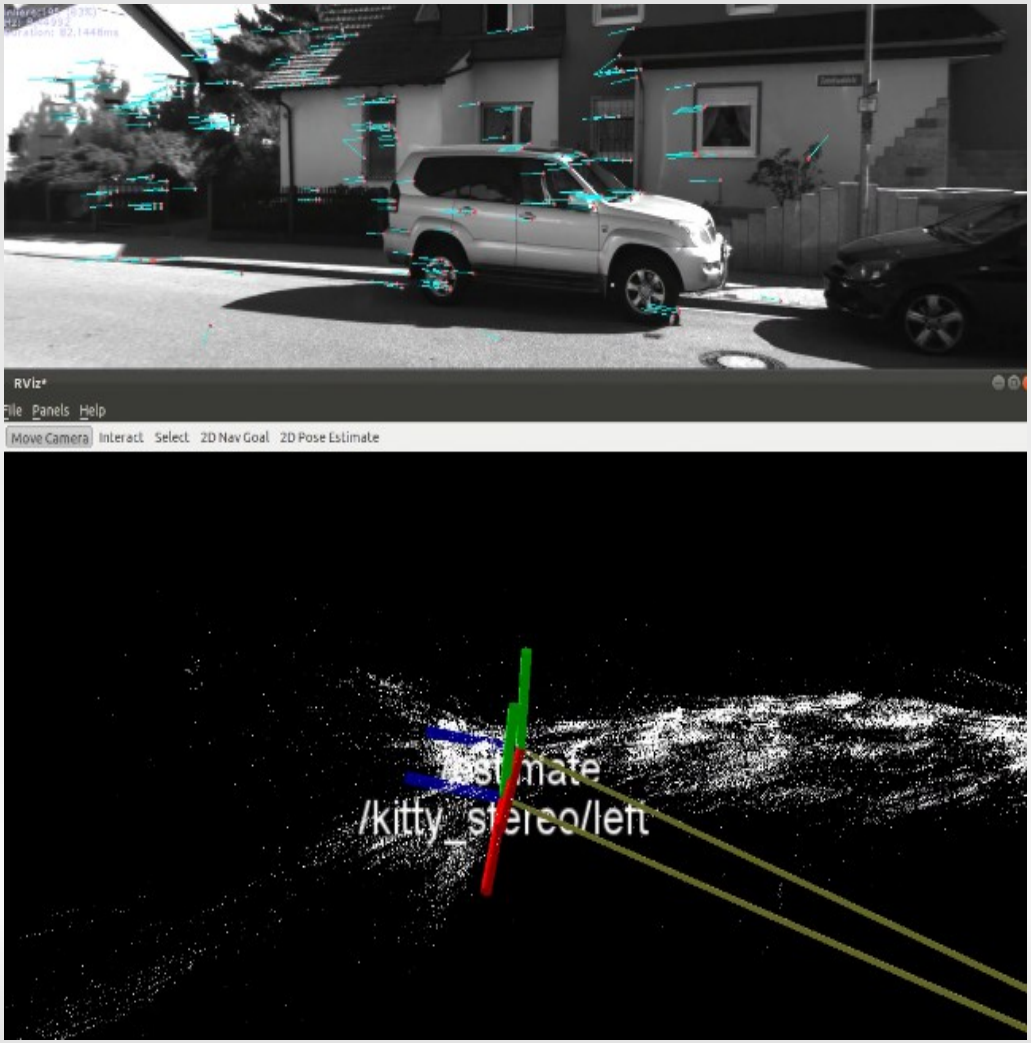
Applied Computer Vision in Robotics

Team Waitinglist:

- Ross Kidson
- Alvaro Gauterin
- Karol Hausman

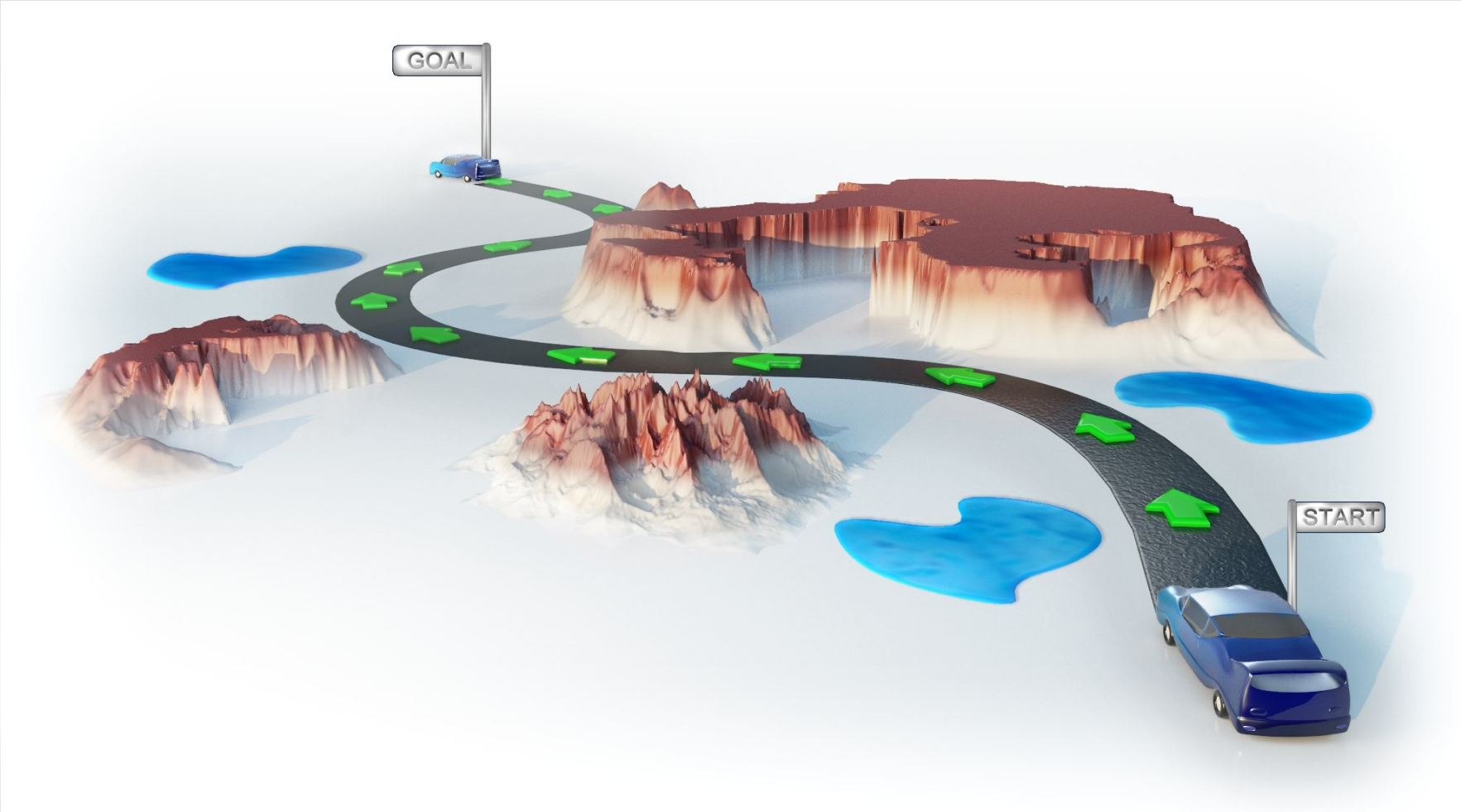


Exercise sheet 3

Exercise sheet 4

Project Motion Planning

- Motion planning in static scenarios



Project Motion Planning

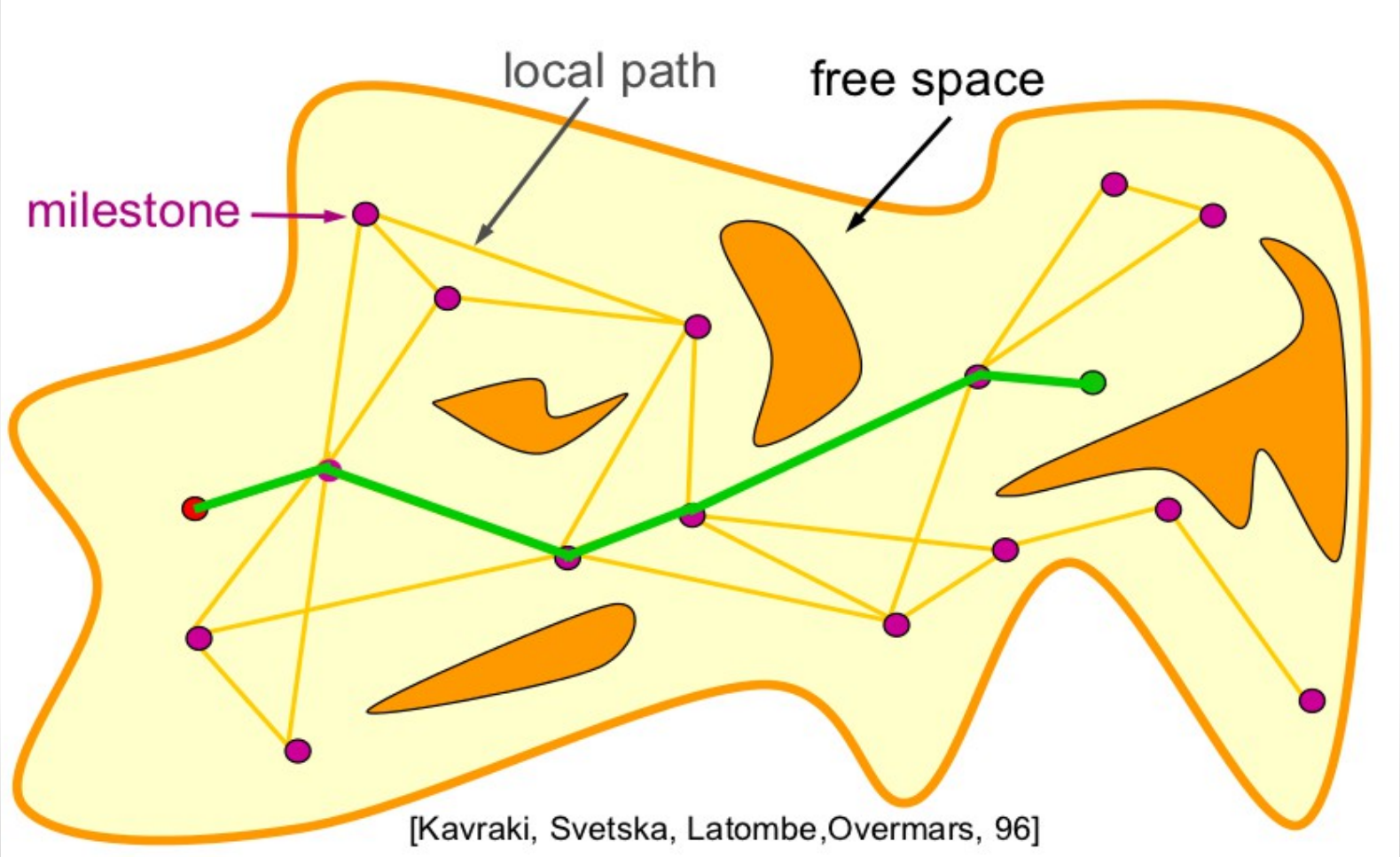
- Motion planning in dynamic scenarios

Static Scenario - Possible solutions

- Different motion planning libraries:
  - OMPL
    - Sampling based planning library
    - No collision detection provided
    - Difficult integration with ROS
  - SBPL
    - Search based planning library
    - No dynamic model of a car

Static Scenario - OMPL Theory

- Probabilistic Motion Planning
  - Rapidly exploring random tree (RRT)



Static Scenario - OMPL Approach

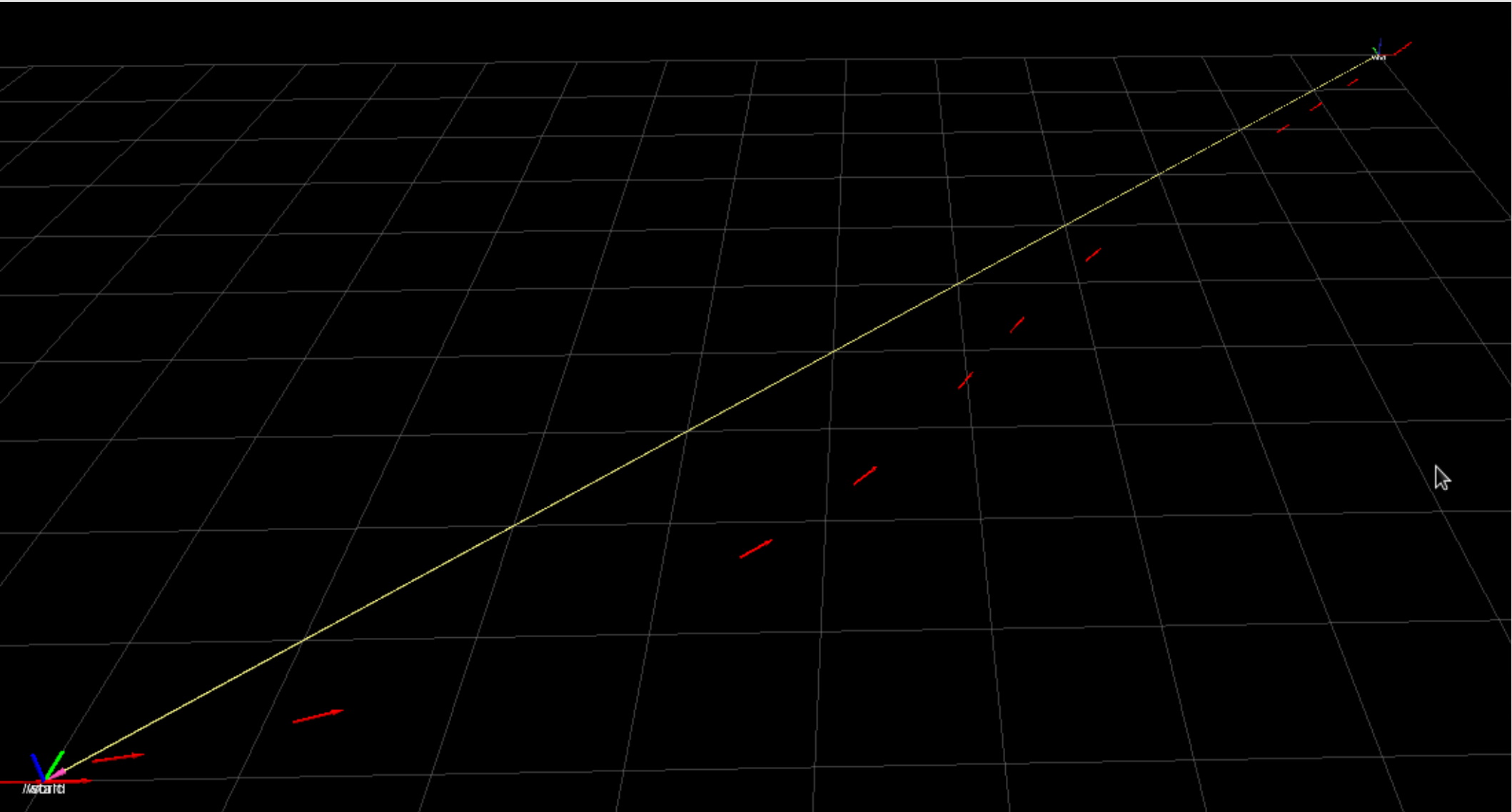
- OMPL → incorporate dynamics of a car
- Implement collision detection

- Dynamics of the second order car:

$$\dot{x} = v\cos\theta$$
$$\dot{y} = v\sin\theta$$
$$\dot{\theta} = \frac{vm}{L}\tan\varphi$$
$$\dot{v} = u_0$$
$$\dot{\varphi} = u_1$$

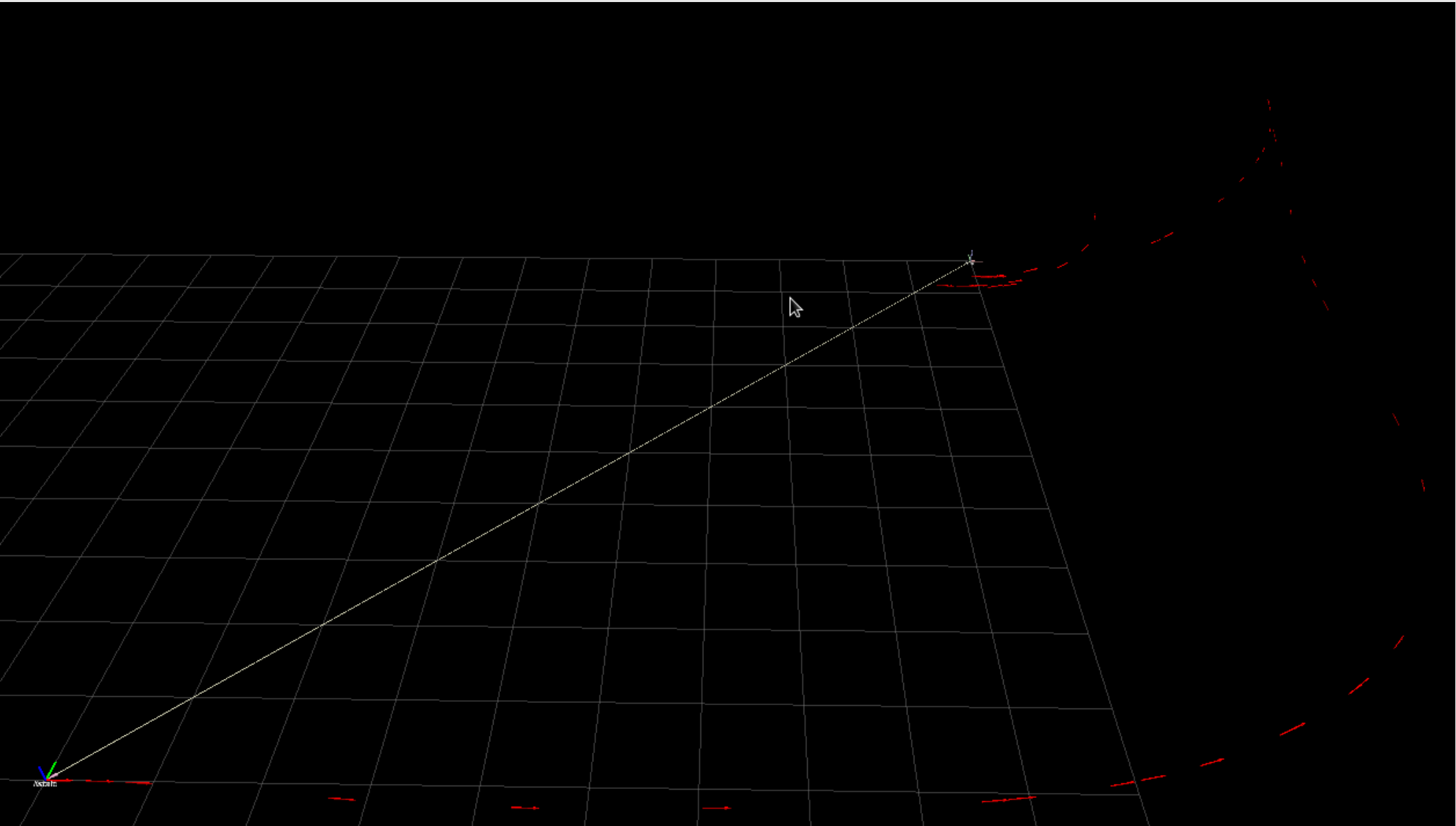
Static Scenario – OMPL Results

- Suboptimal



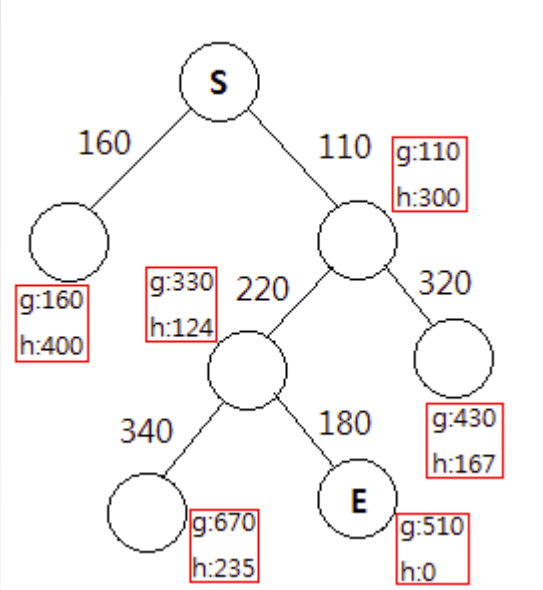
Static Scenario – OMPL Results

- Suboptimal



Static Scenario - SBPL Theory

- SBPL Planner uses ARA* (Anytime Repairing A*) algorithm that was developed by Maxim Likhachev at CMU
- A* algorithm
  - Can be slow
  - Always optimal



Static Scenario - SBPL Theory

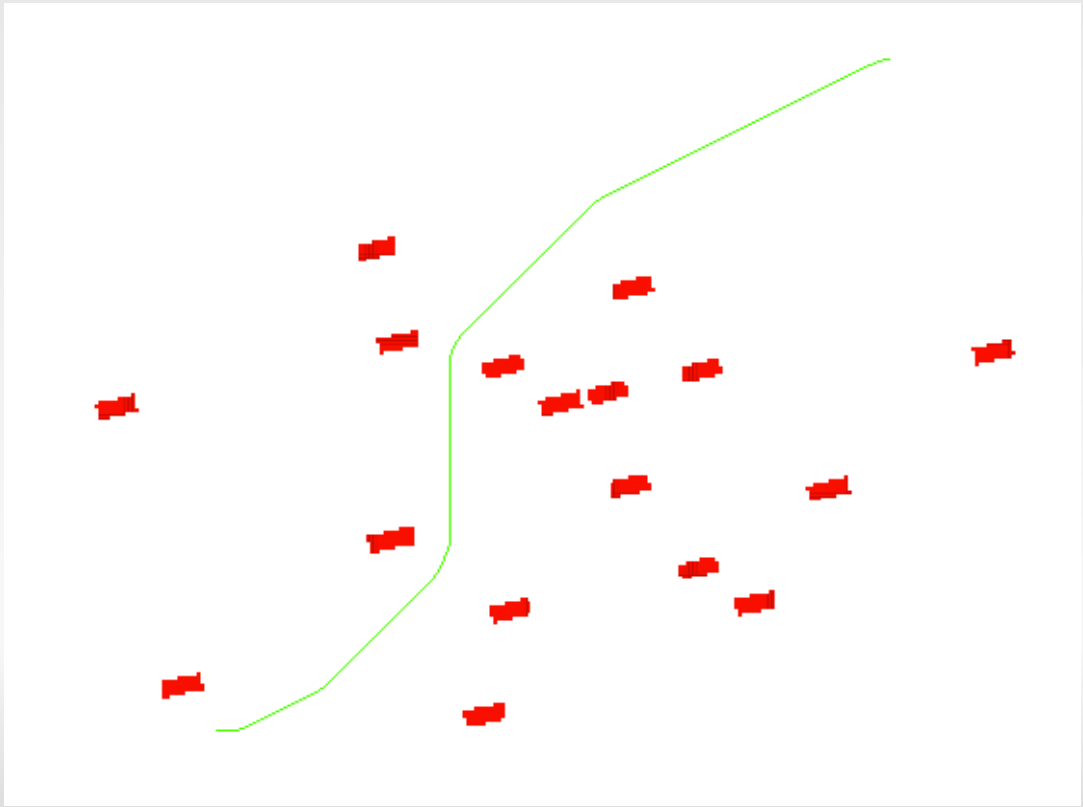
- ARA* algorithm
  - A* with inflated heuristics
    - Actual heuristics multiplied by an inflaction factor ε>1

$$f(s) = g(s) + \varepsilon * h(s)$$

  - Suboptimal but faster → suboptimality is predictable
  - Many iterations → ε is getting smaller
  - If there is time remaining → try with smaller ε

Static Scenario - SBPL Approach

- SBPL integration with Rviz
- User Interface by clicking start and goal poses
- Integration with BMW Software

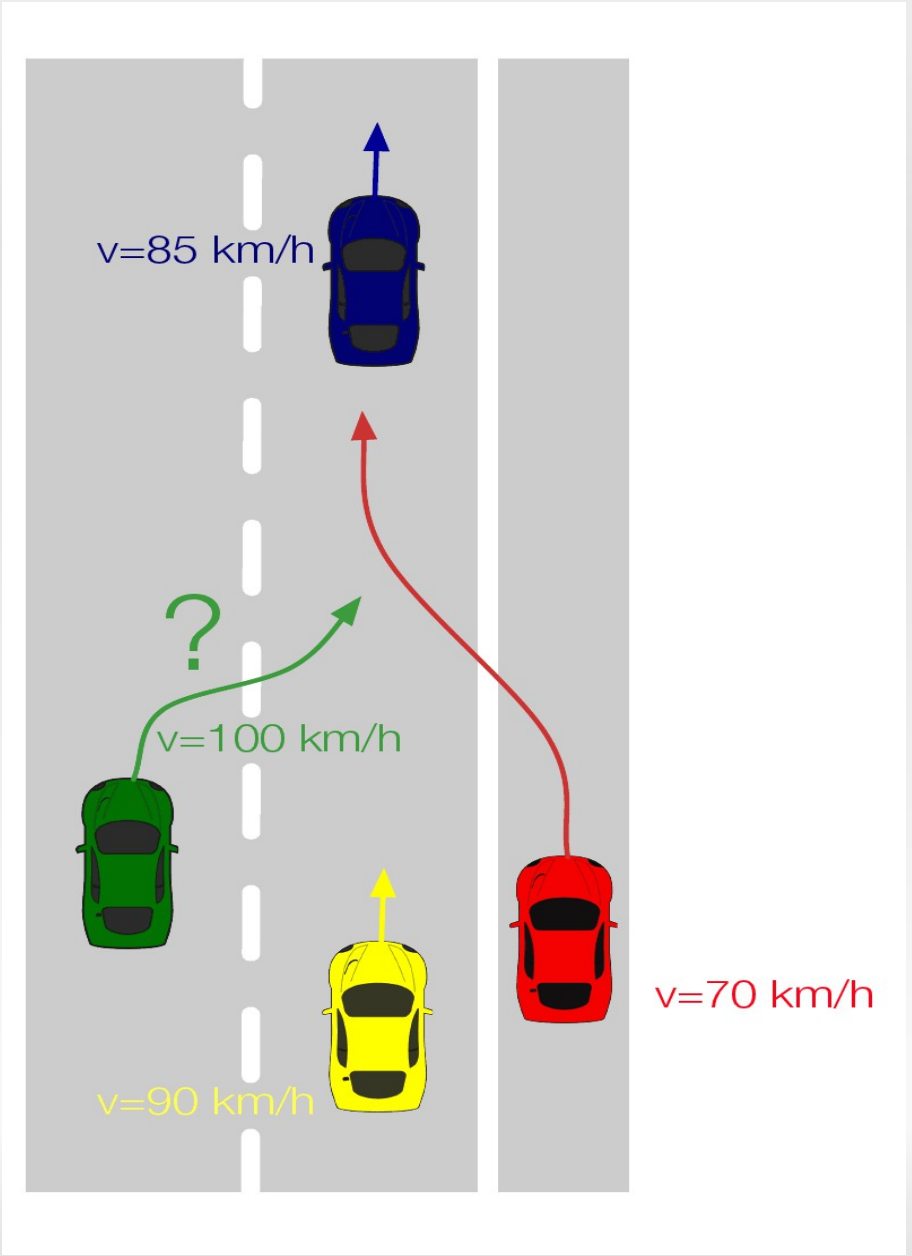


Static Scenario – SBPL Results

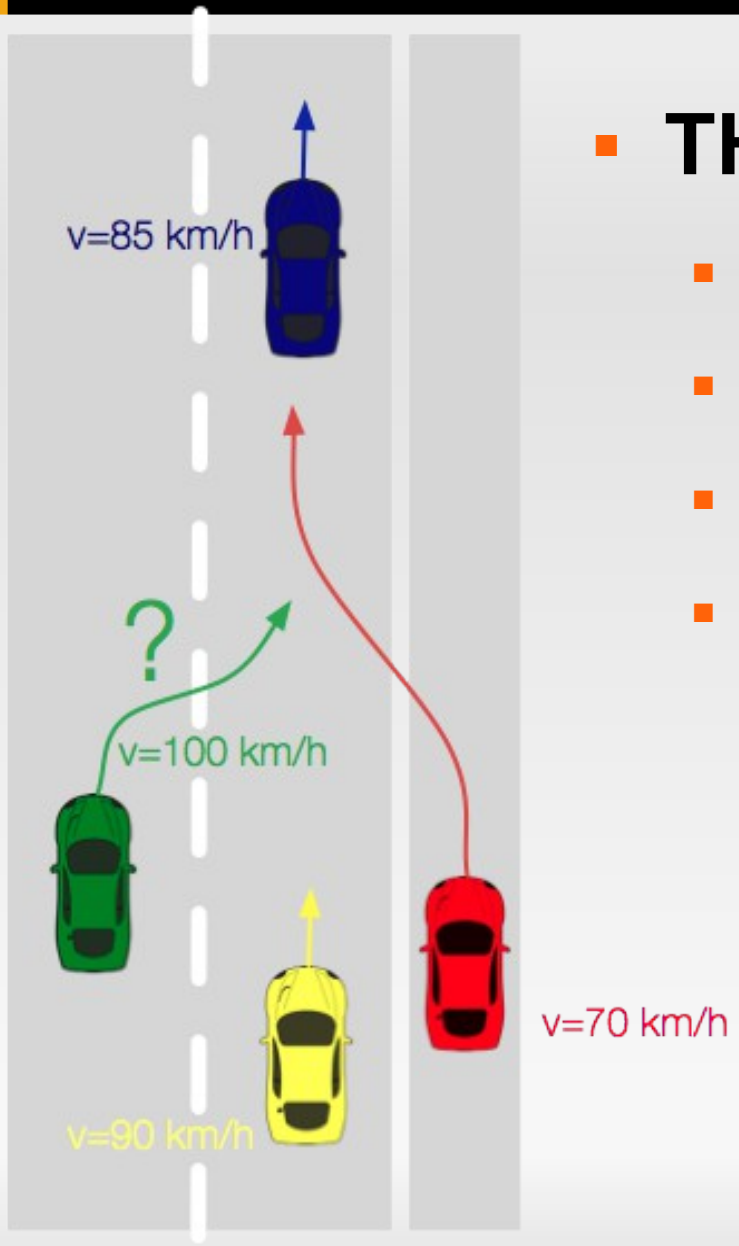
Static Scenario - SBPL Results

Dynamic Scenario

- Overtaking a car
- AI involved?

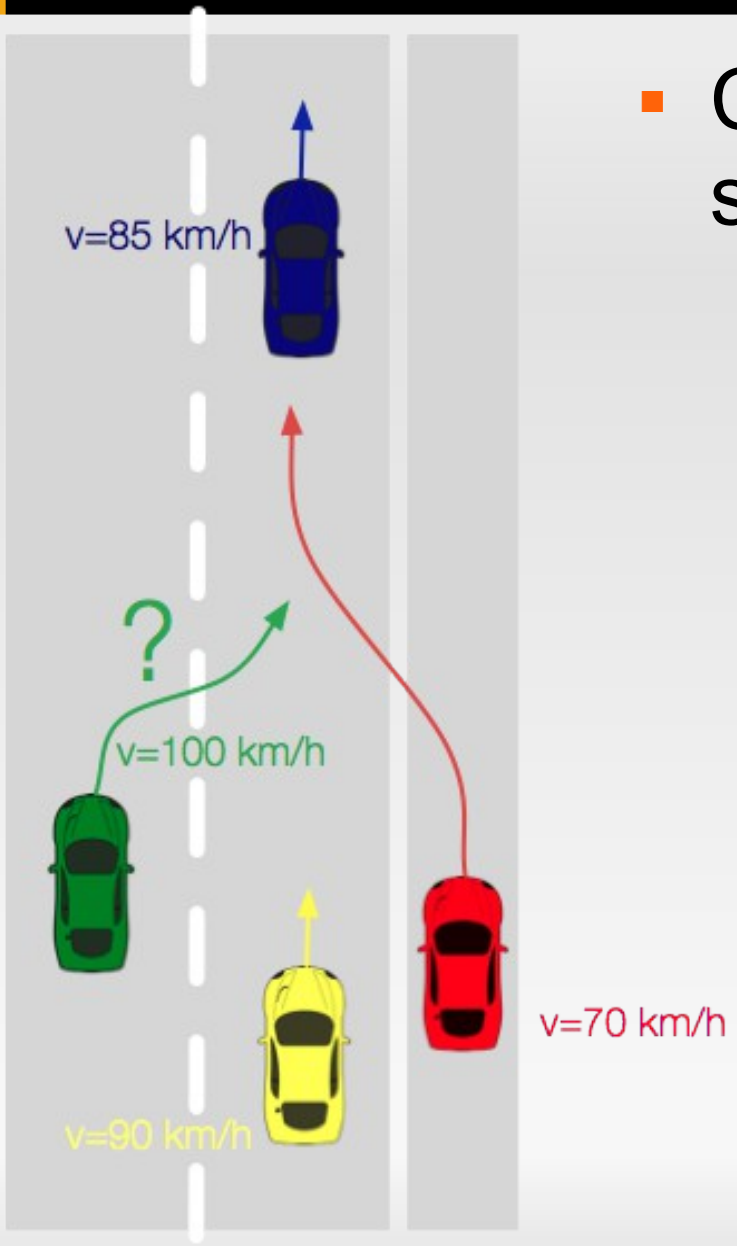


Dynamic Scenario - PLAN



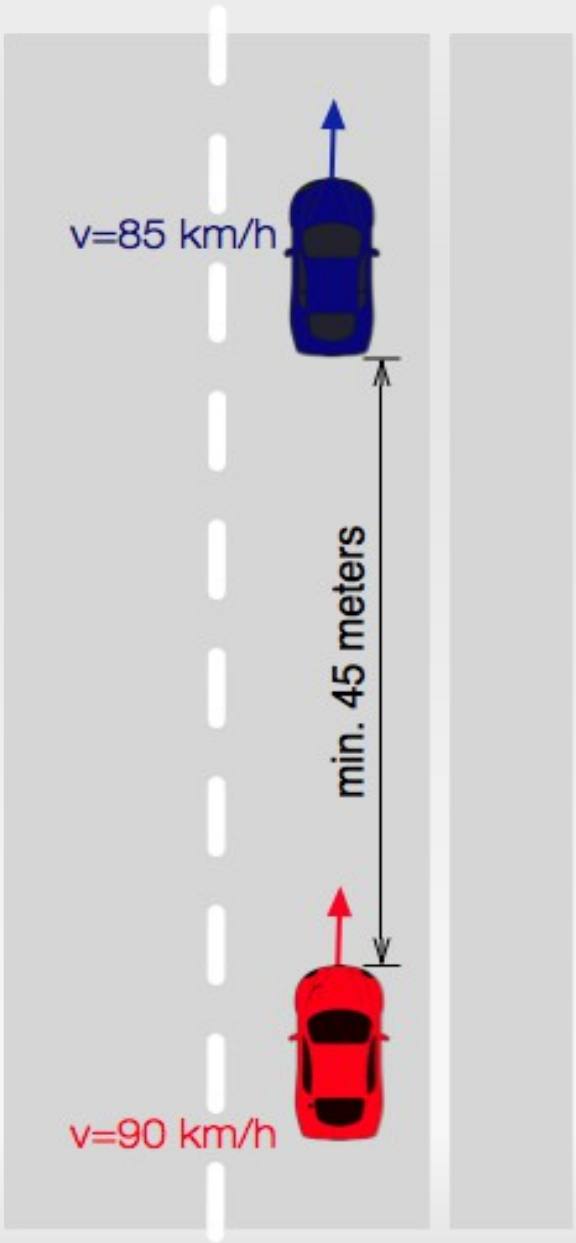
- **THE PLAN:**
  - Accelerate on the emergency lane
  - Change to the lane on the left
  - Again – change to the lane on the left
  - Keep driving straight ahead

Dynamic Scenario – Check list



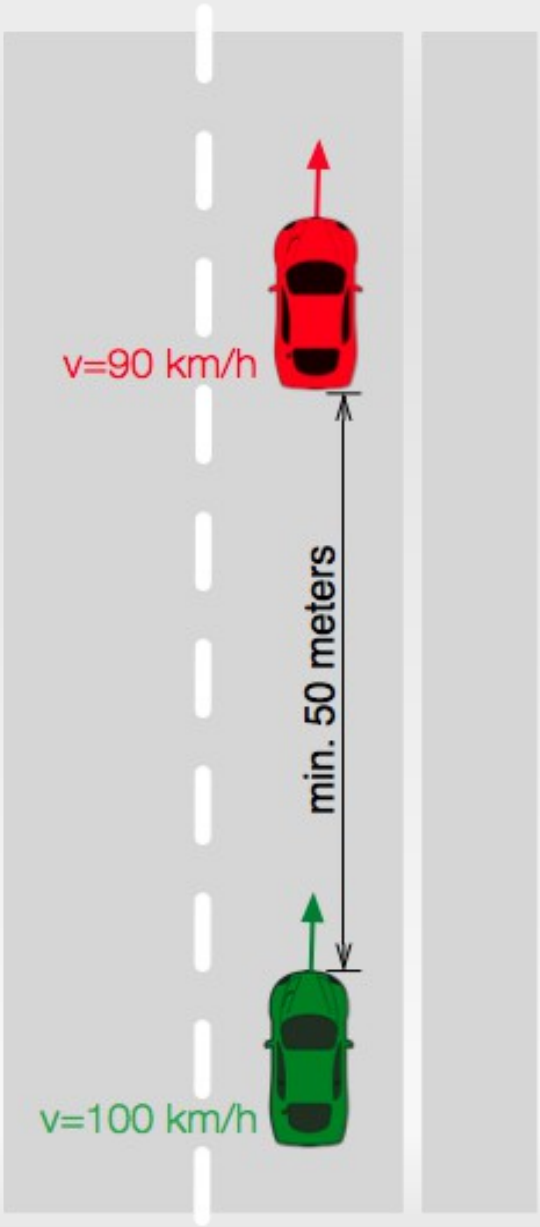
- Check list – to be checked every second:
  - Is there a threat by the car in front of me?
  - Is there a threat by the car behind me?
  - Execute the plan

Dynamic Scenario – Check list



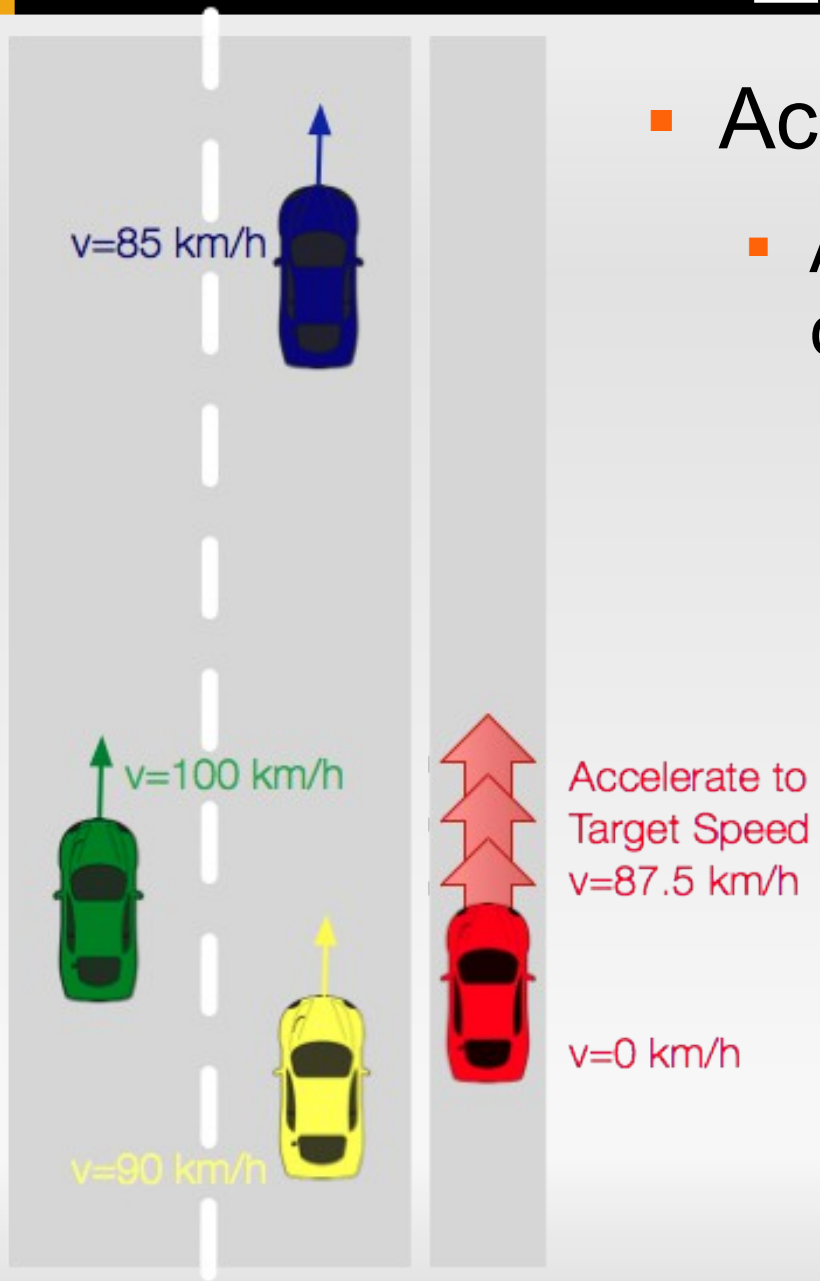
- Threat by a car in front
  - If distance smaller than half of my velocity → slow down

Dynamic Scenario – Check list



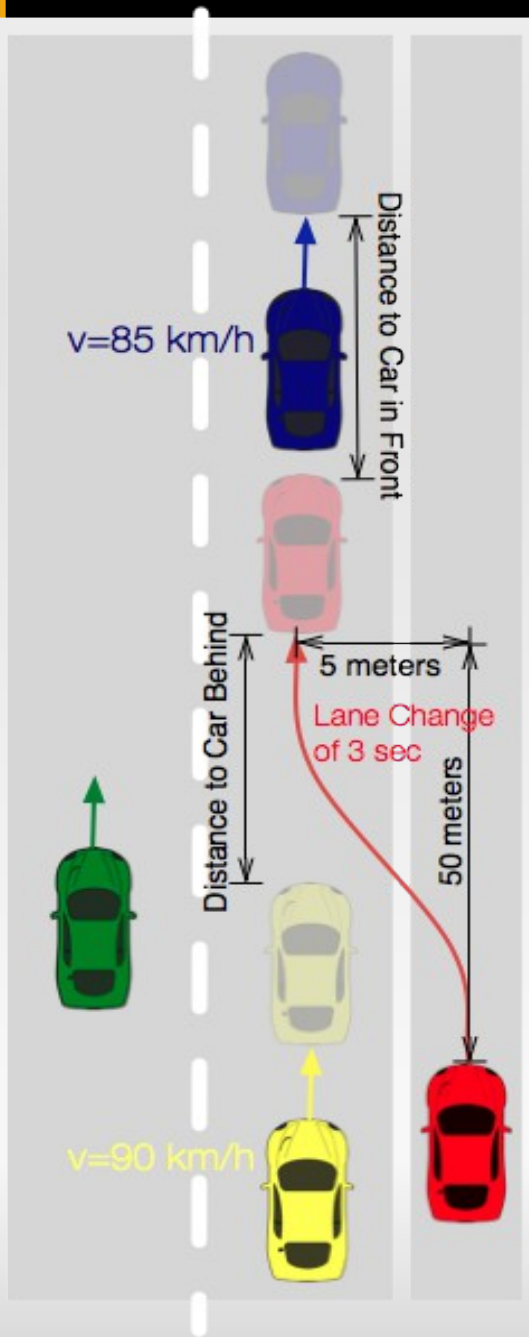
- Threat by a car behind
  - If distance smaller than half of his/her car → accelerate

Dynamic Scenario – Plan Execution



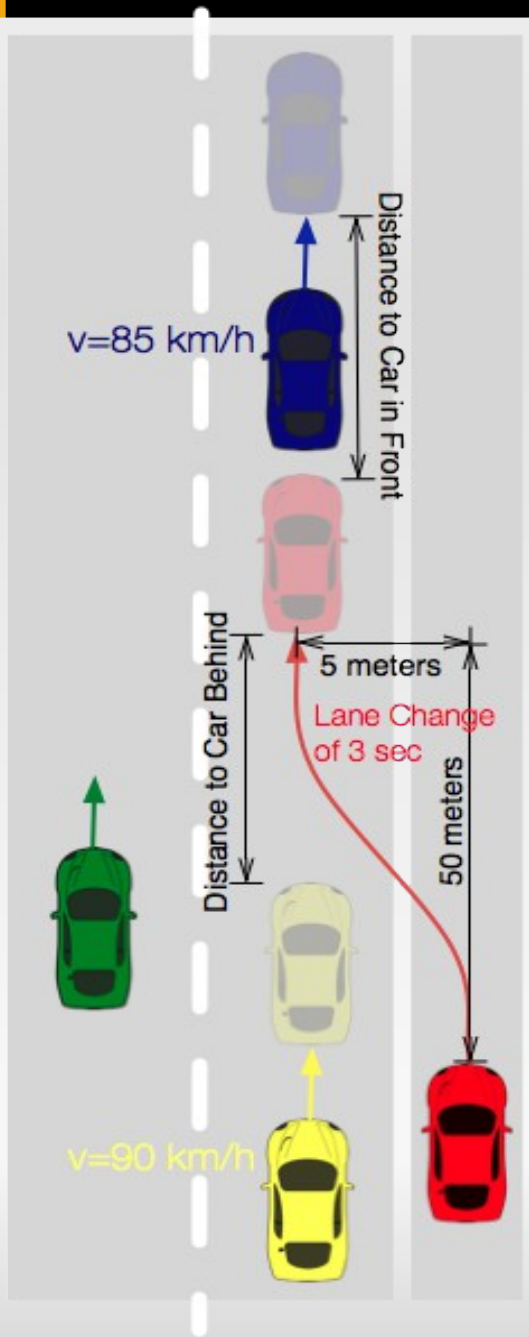
- Acceleration on emergency lane
  - Accelerate to the avg velocity of the cars on the left lane next to me

Dynamic Scenario – Plan Execution



- Can I change a lane?
  - Lane change takes 3 sec.
  - Where would I be if I change the lane?
  - Where would other cars be in 3 sec.?
  - Calculate hypothetical distances to the car in front and behind me
    - If distances too big → do not change the lane

Dynamic Scenario – Plan Execution



- If a lane change is not possible
  - Is the problem a car in front of me? → slow down
  - Is the problem a car behind me?
    - is it slower than me? → accelerate in order to change the lane
    - Is it faster than me? → keep the same velocity so that other car can overtake me

Dynamic Scenario – Final Result?

Dynamic Scenario – Final Result

Thank you

- QUESTIONS?