MIT DRIVERLESS

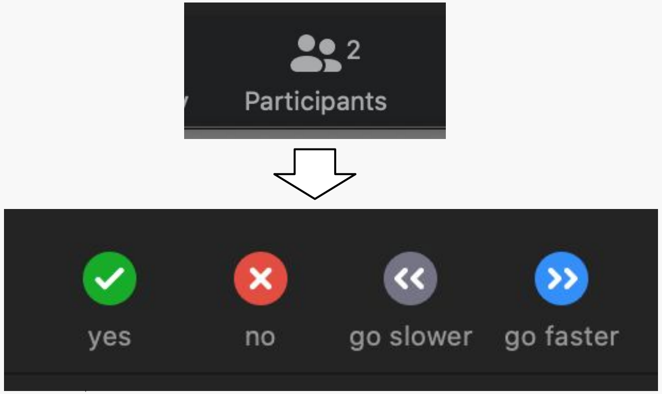
End of Summer Showcase

August 28, 2020

Agenda

- Introduction
- Project presentations
  - Architecture
  - Perception
  - Planning and Controls
- Q&A

Q&A: Indicate in Zoom which room you want to join



General*

Architecture

Perception

Plan. & Controls

*Stays in main room

MIT DRIVERLESS

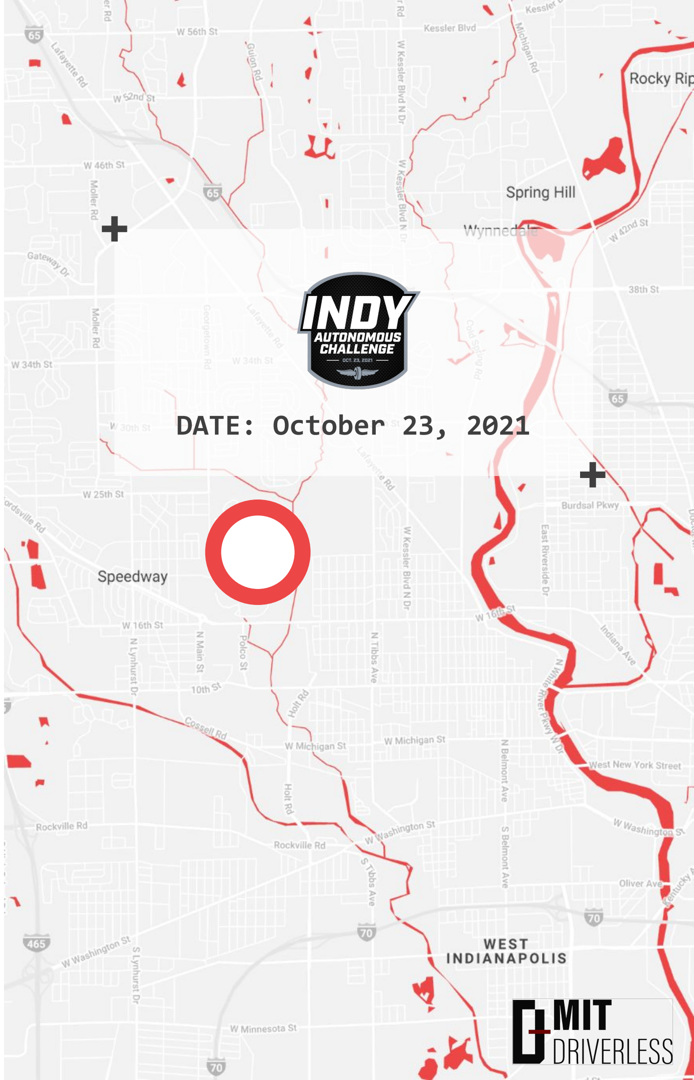
OUR MISSION

- To be the center for **applying** autonomous vehicle technology at MIT.
- To create and implement cutting edge technology to compete against and **learn** from our peers around the world.
- To build relationships across campus and beyond, while **developing leaders** in the AV, robotics, and tech industries.

OUR NEXT CHALLENGE

Autonomously drive an IndyLights racecar **head-to-head** at the world famous **Indianapolis Motor Speedway**. Speeds up to **180 MPH** against **31** of the top universities from around the world.

“If everything seems under control, you’re just not going fast enough.” — Mario Andretti



Progress to date

We took a project-based approach to our summer work.

Our team focused on 3 key challenges:

1. Defining the **requirements** and creating **system architecture** concepts

2. Building a “**faster LiDAR**” for 3D object detection

3. Developing our first **planner and controller** for **multi-agent** racing

MIT DRIVERLESS

Interested in our work?

- Become a sponsor! Email us to learn more: driverless-business@mit.edu
- Join our team! Apply here: http://driverless.mit.edu/join-the-team
- Follow us on social media!

MIT DRIVERLESS

Architecture

Architecture Team

Nick Stathas

Tejas (TJ)
Chafekar

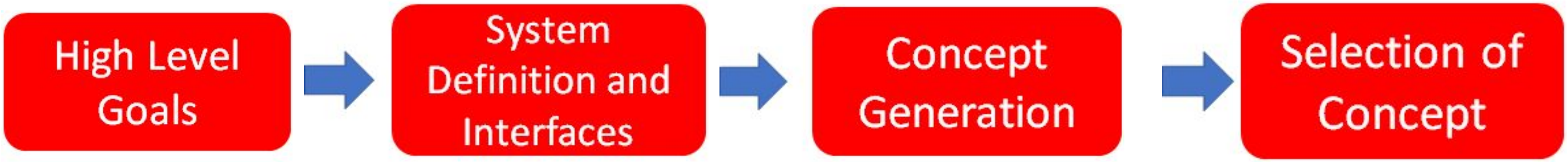
Bouke Edskes

MIT DRIVERLESS


Cars on Track
485 Formula Student Team Delft Driv...
2019/08/10 9:39:52
Car 485: NL Delft TU // Lap: 3 // 75.77 s
Share with #FSG19

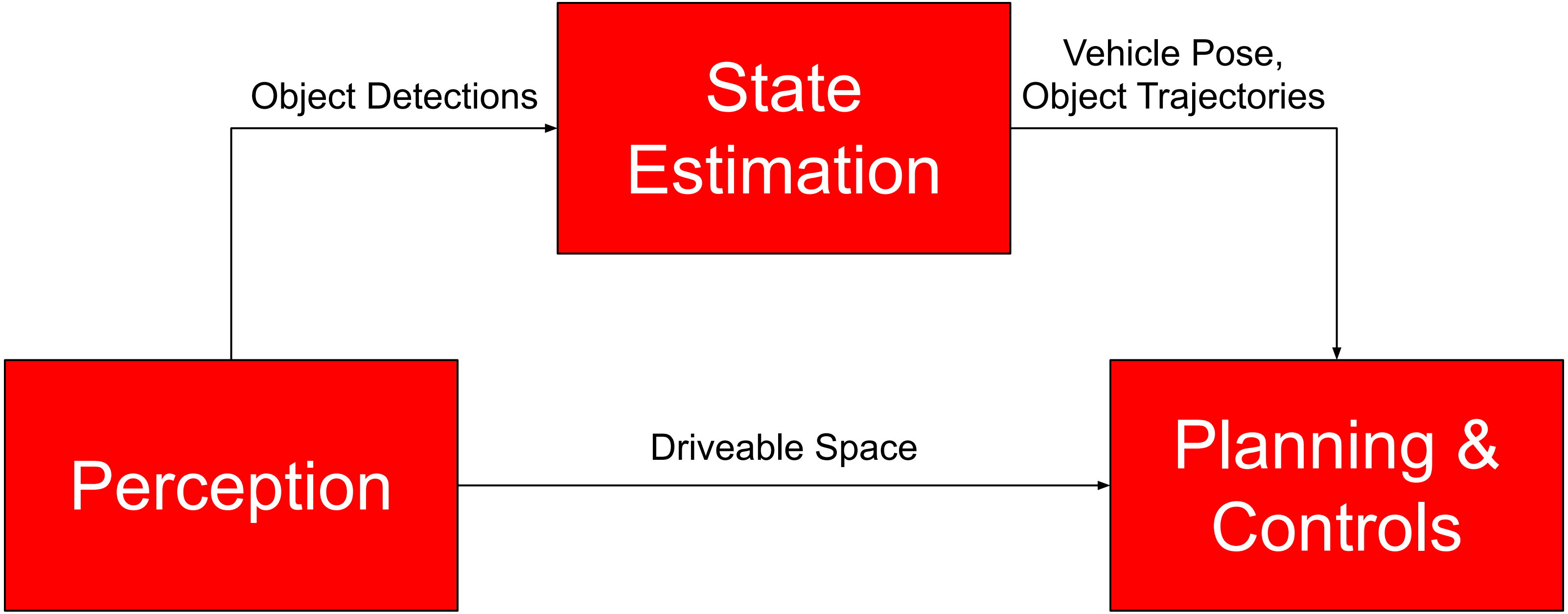

NBC Sports
NETWORK
MONACO
SUN 7:30

System Thinking



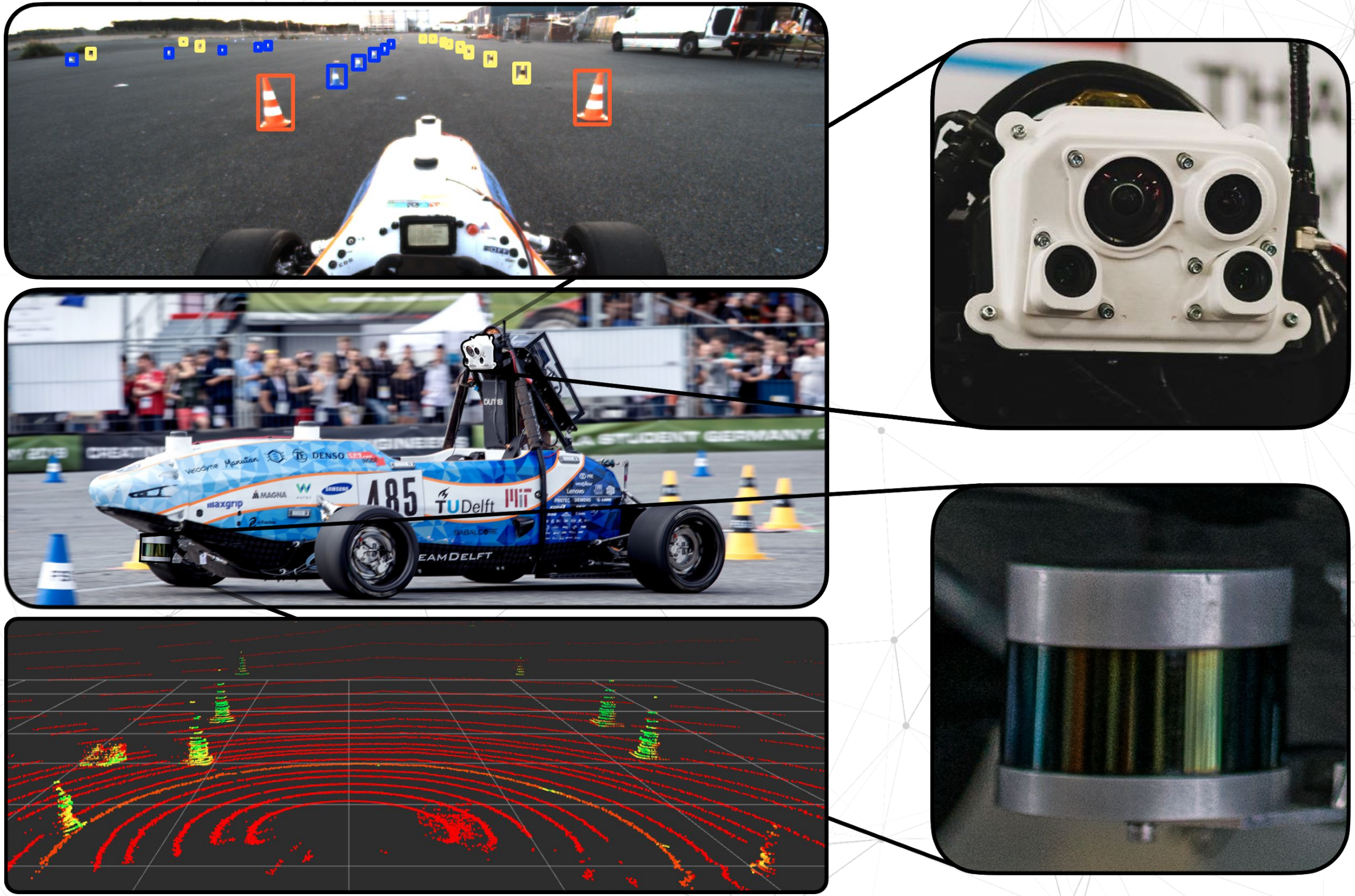
MIT DRIVERLESS

Subsystem Breakdown



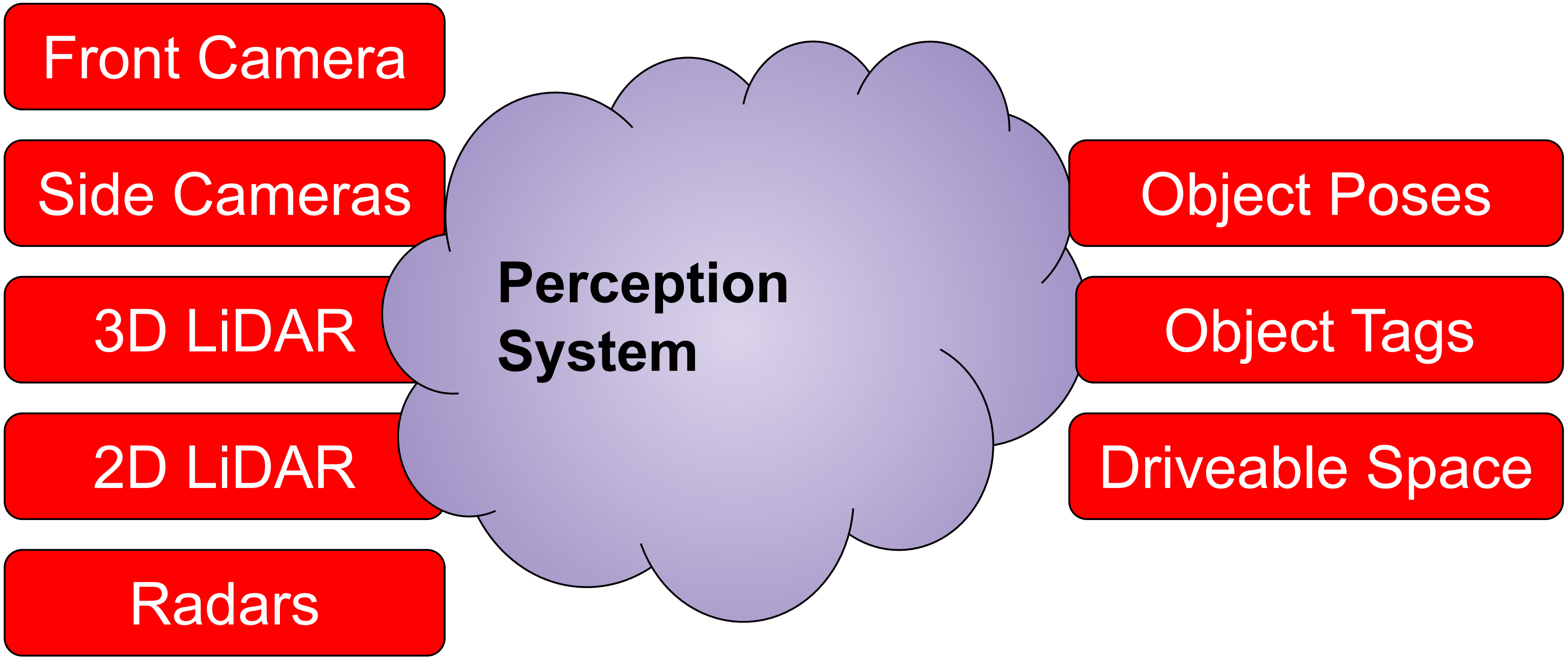
Perception Architecture

MIT DRIVERLESS



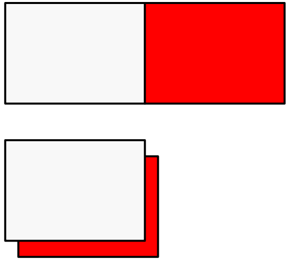
DENSO
485
TUDelft
maxgrip
MAGNA

High Level Goals



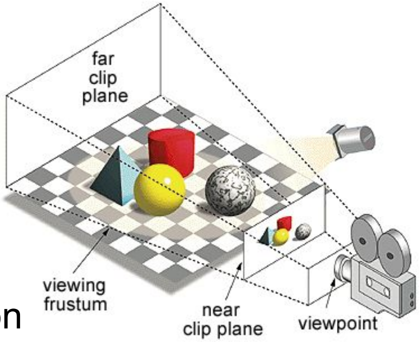
Modular Concept - Monocular Camera Object Detection

Every camera frame is fed to an object detector.



Dimensions

- Multi-sensor
  - Jointly
  - Batched
- Temporal
  - Current frame
  - Past few frames
  - Temporal Shift Module

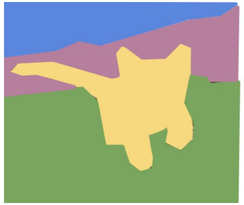


Depth Estimation

- Ground plane assumption
- Neural network depth estimation
- Project bounding box to frustum on LiDAR pointcloud
- Render-based depth estimation (ask us about it!)
- Inverse perspective-n point on regressed key points
  Example key points: wing tips, mirrors, tires

Detectors

- Bounding box
- Instance segmentation



State Estimation Architecture

High Level Goals

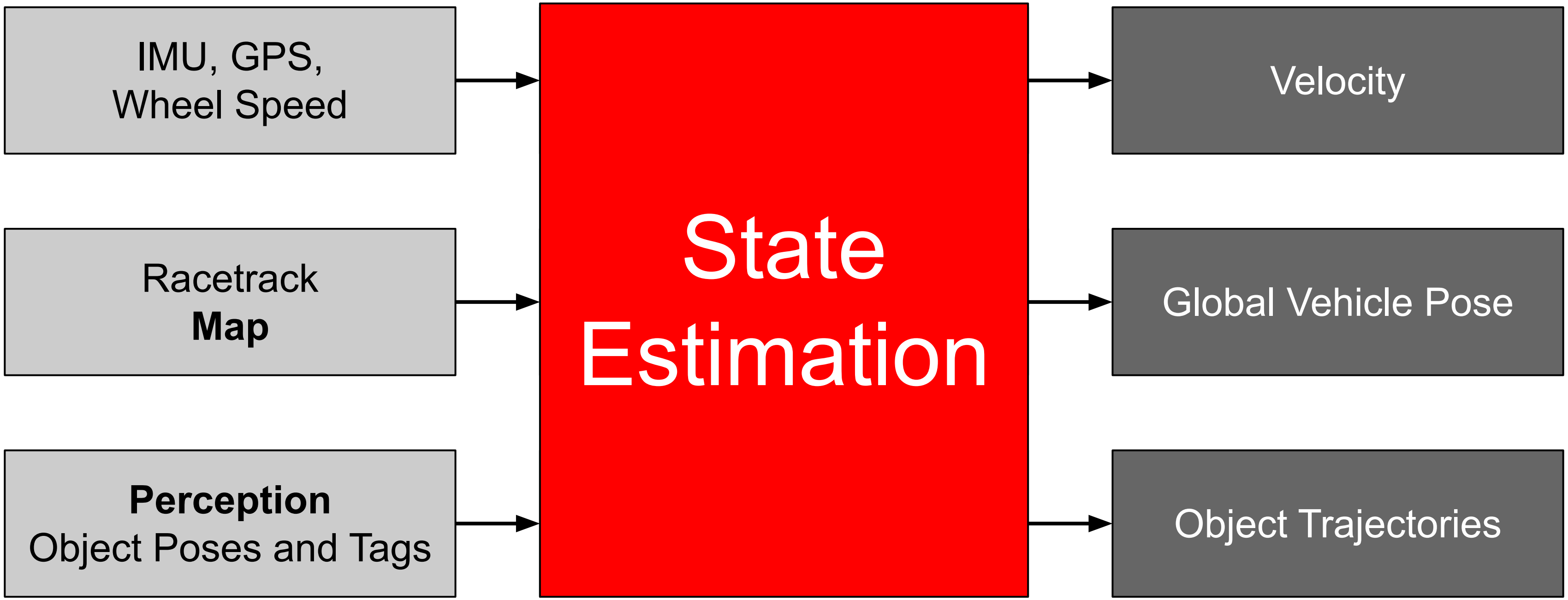
Where Are We?

- **Local Estimation** — How is our position and heading changing over time?
- **Global Estimation** — Where are we on the racetrack map?

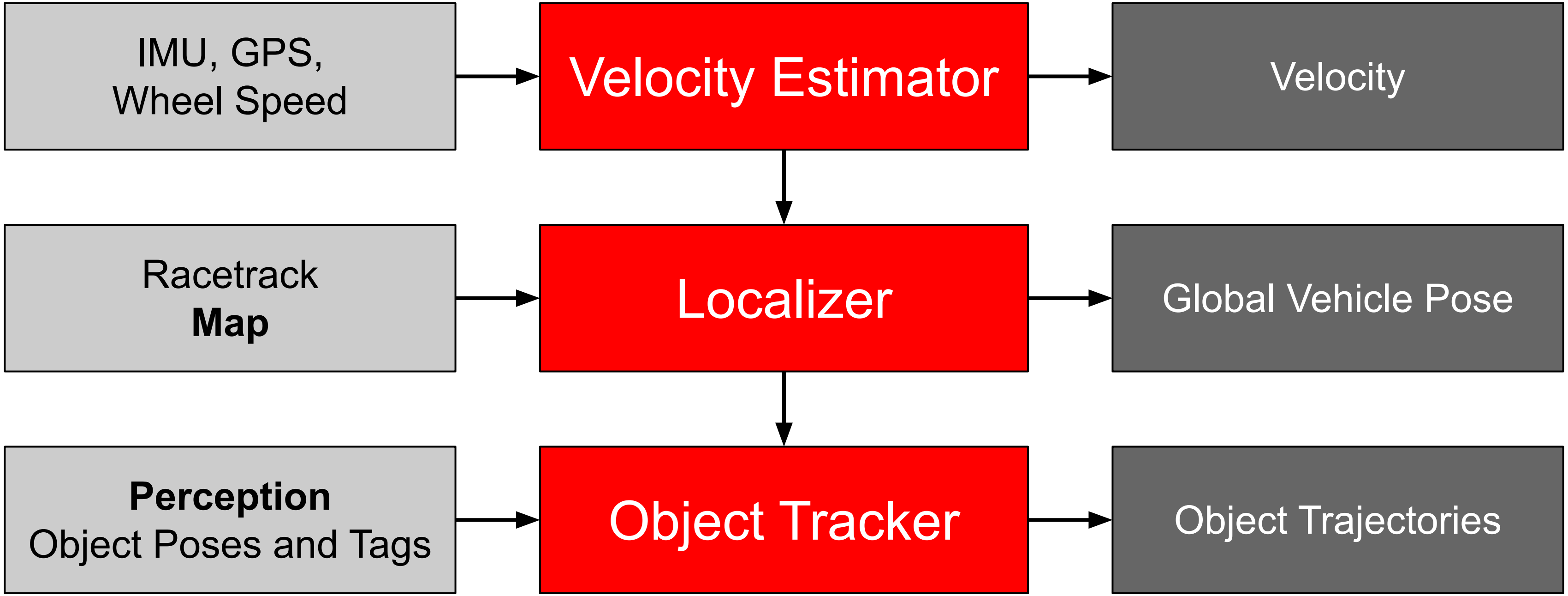
Where Are Other Vehicles?

- **Object Tracking** — What are the trajectories of other vehicles?

Interface Overview

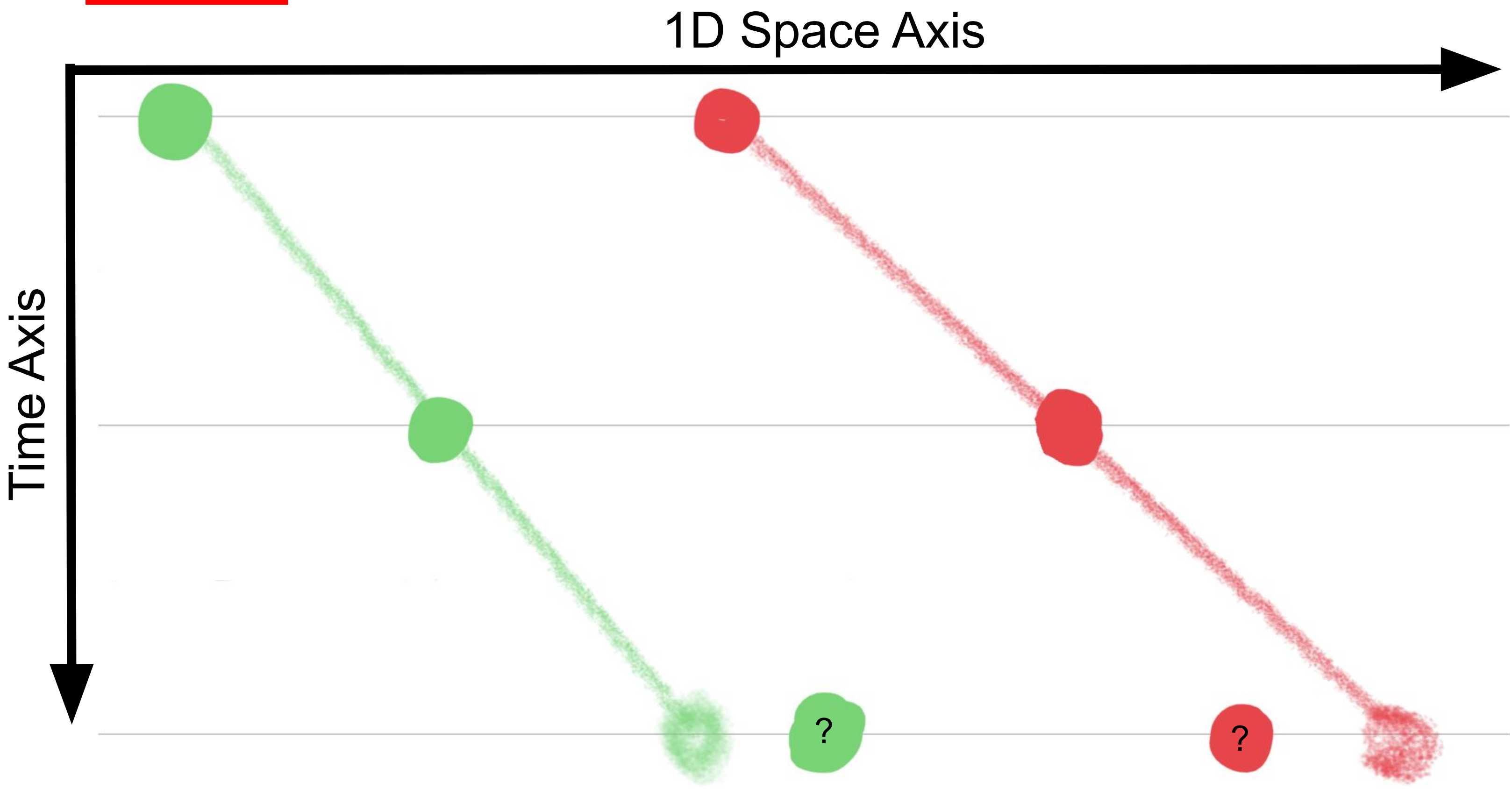


Concept Overview



MIT DRIVERLESS

Object Tracking Intuition



MIT DRIVERLESS

Planning and Controls Architecture

MIT DRIVERLESS

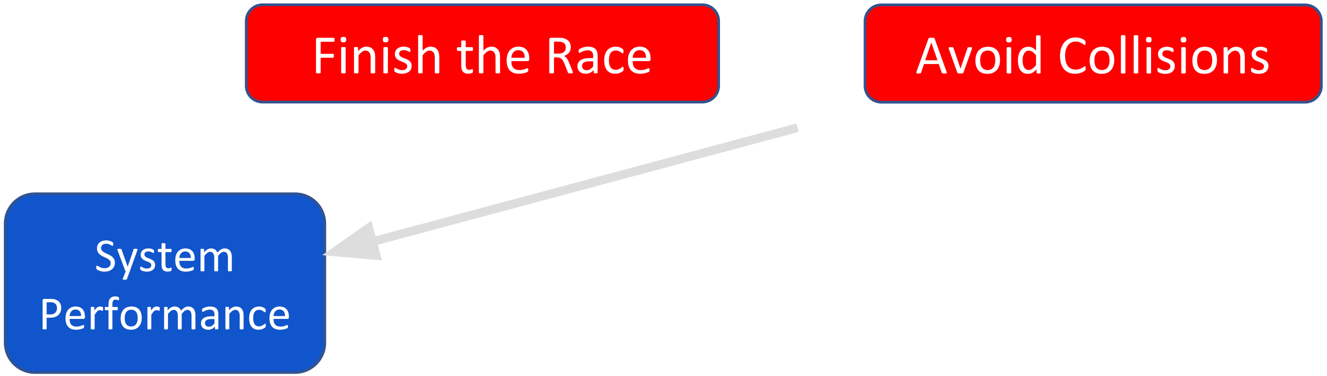
High Level Goals

Finish the Race

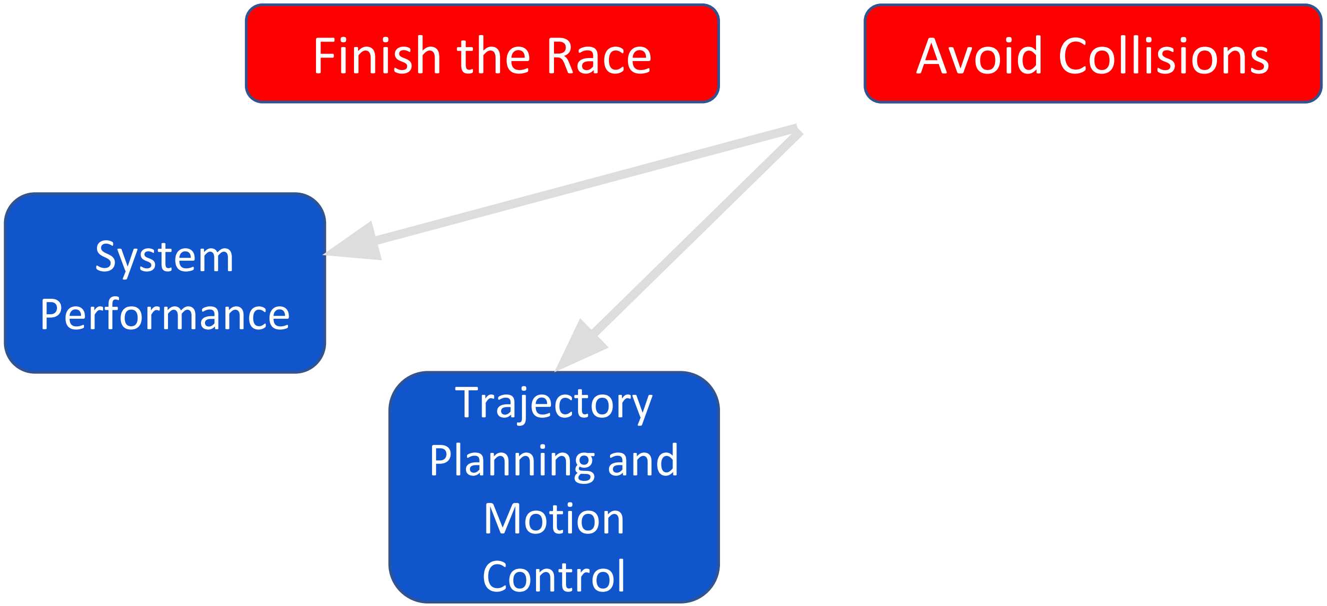
Avoid Collisions

MIT DRIVERLESS

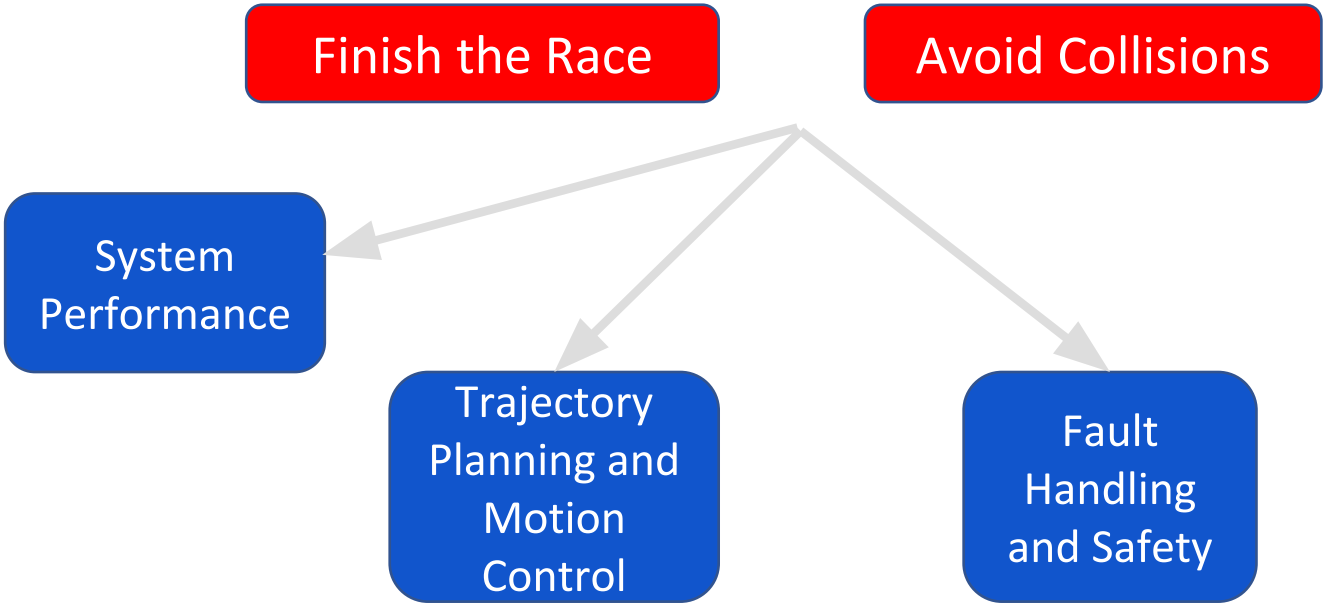
High Level Goals



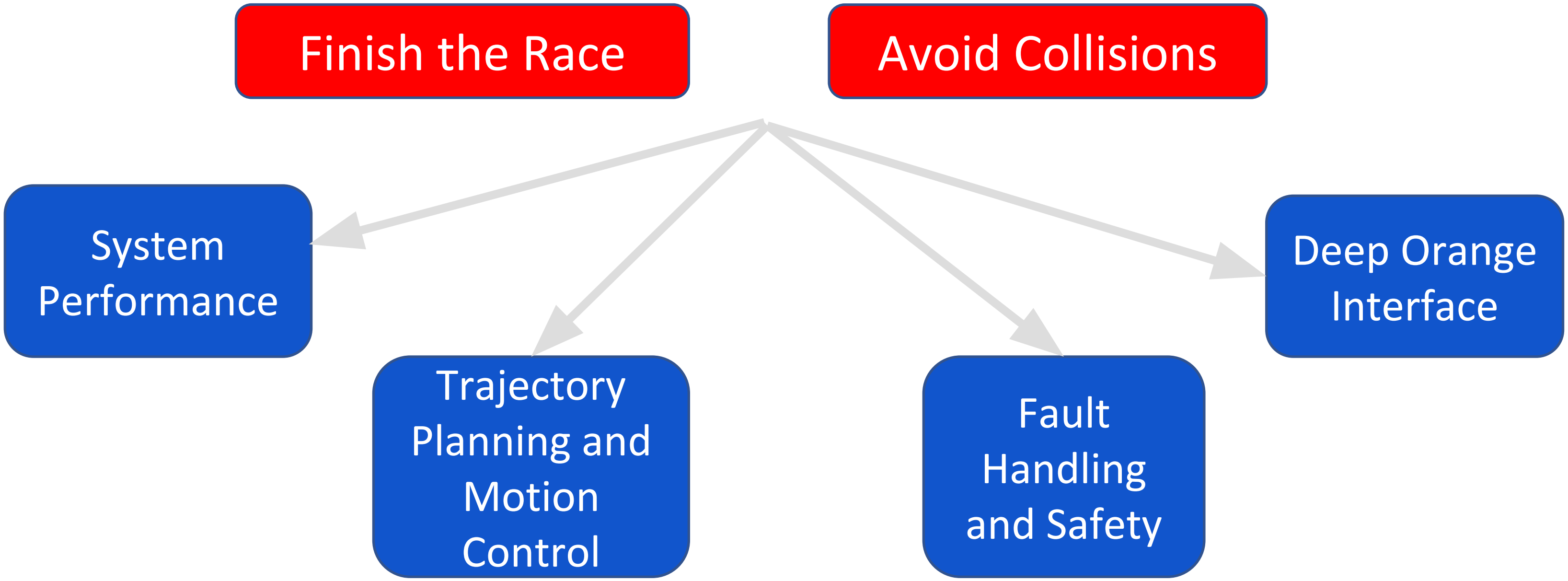
High Level Goals



High Level Goals

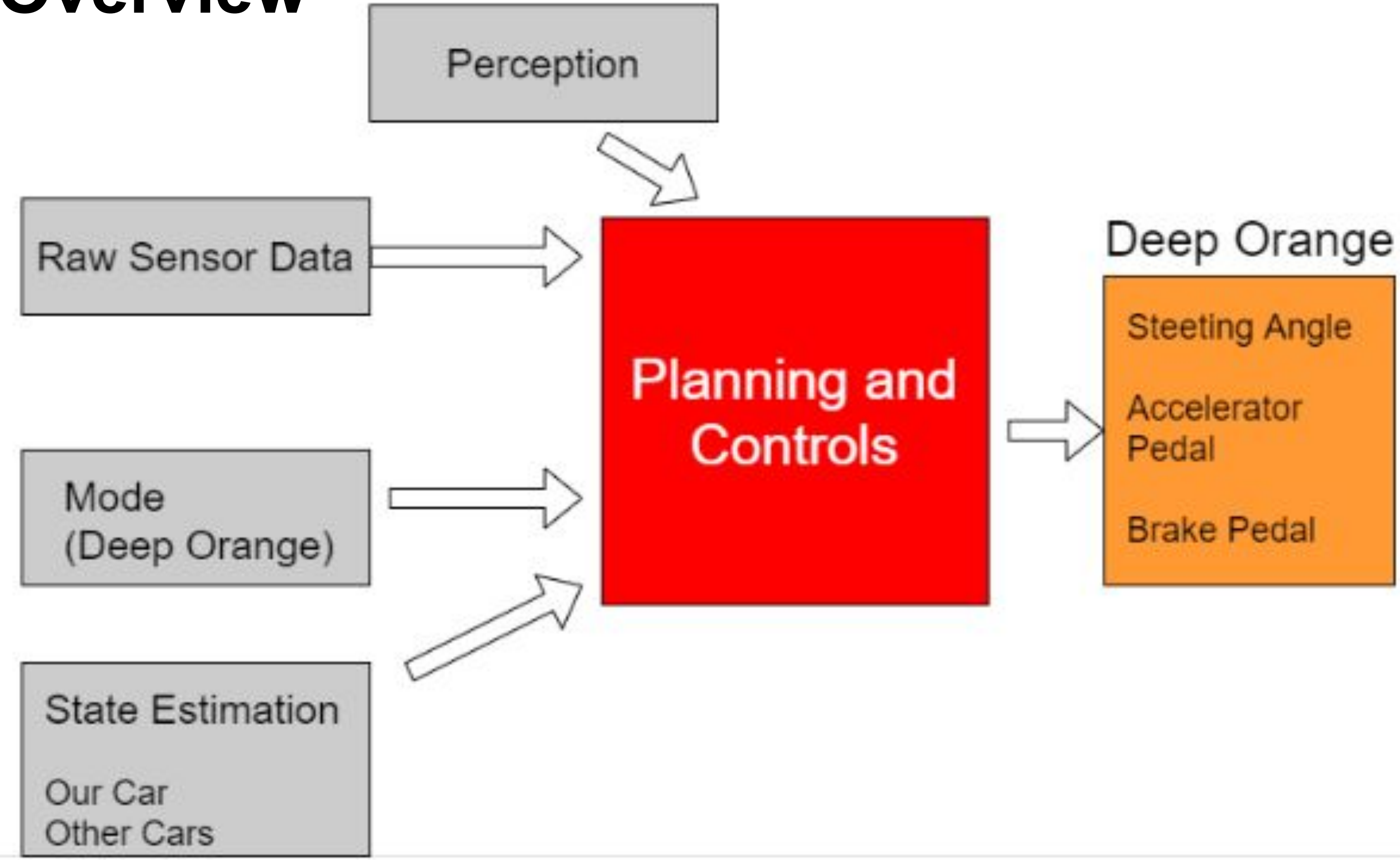


High Level Goals

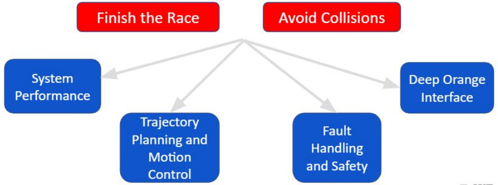


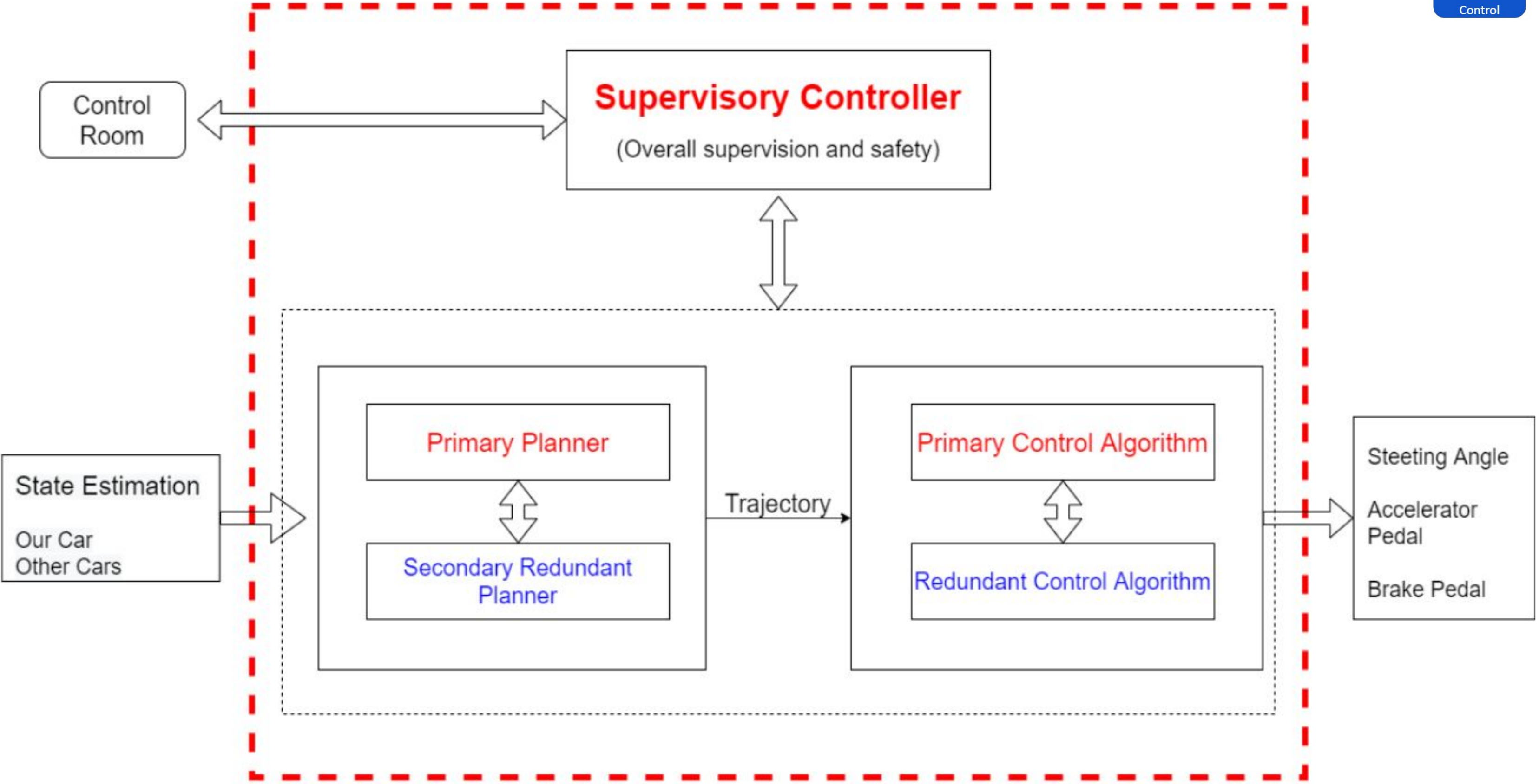
MIT DRIVERLESS

System Overview



Concept


Finish the Race
Avoid Collisions
System Performance
Trajectory Planning and Motion Control
Fault Handling and Safety
Deep Orange Interface


Control Room
Supervisory Controller
(Overall supervision and safety)
State Estimation
Our Car
Other Cars
Primary Planner
Secondary Redundant Planner
Trajectory
Primary Control Algorithm
Redundant Control Algorithm
Steeting Angle
Accelerator Pedal
Brake Pedal

Perception

Interested in hearing more? Join the Q&A by clicking

Perception Team

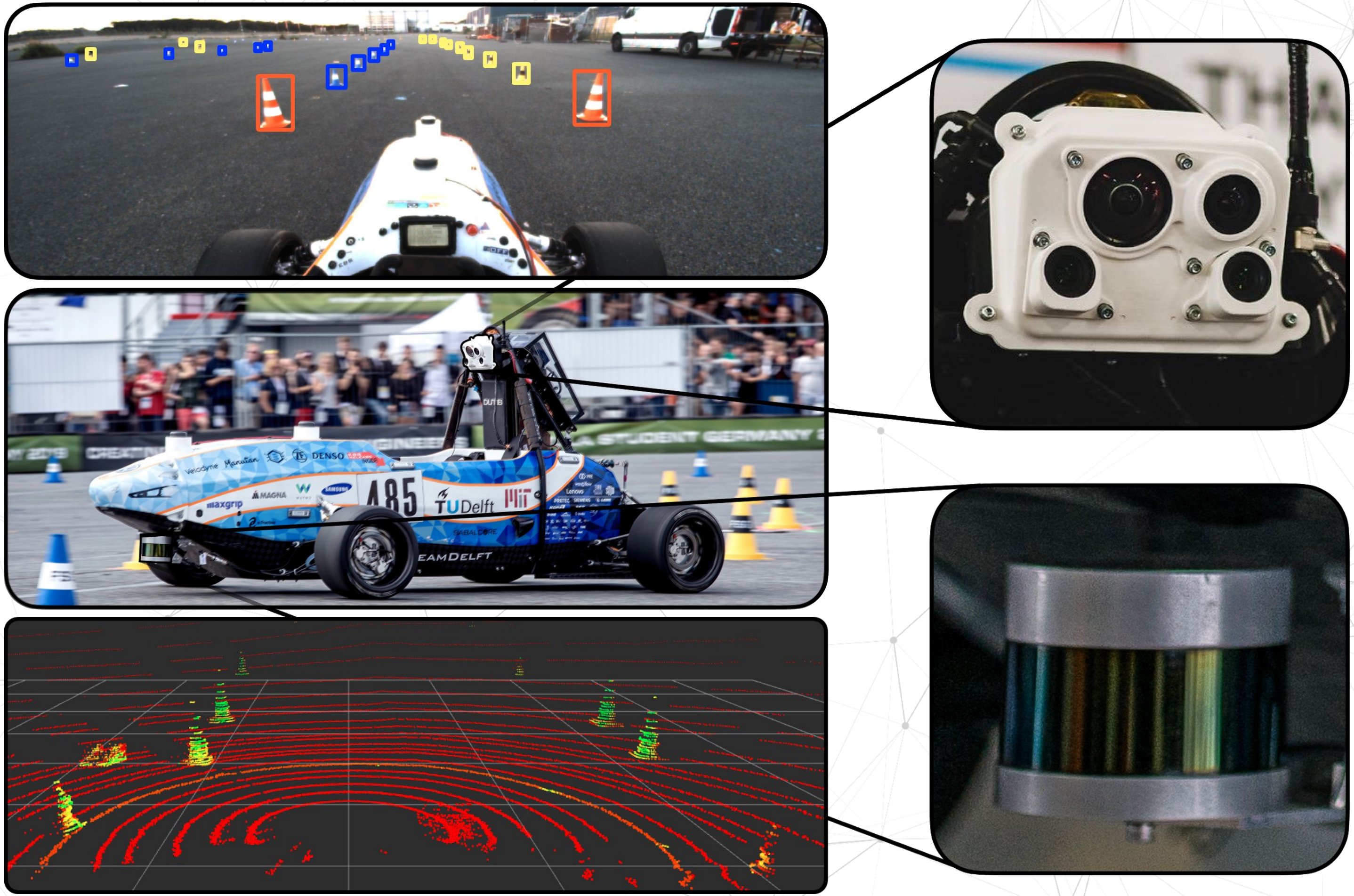
Sibo Zhu

Zhijian Liu

Sophie Zhai

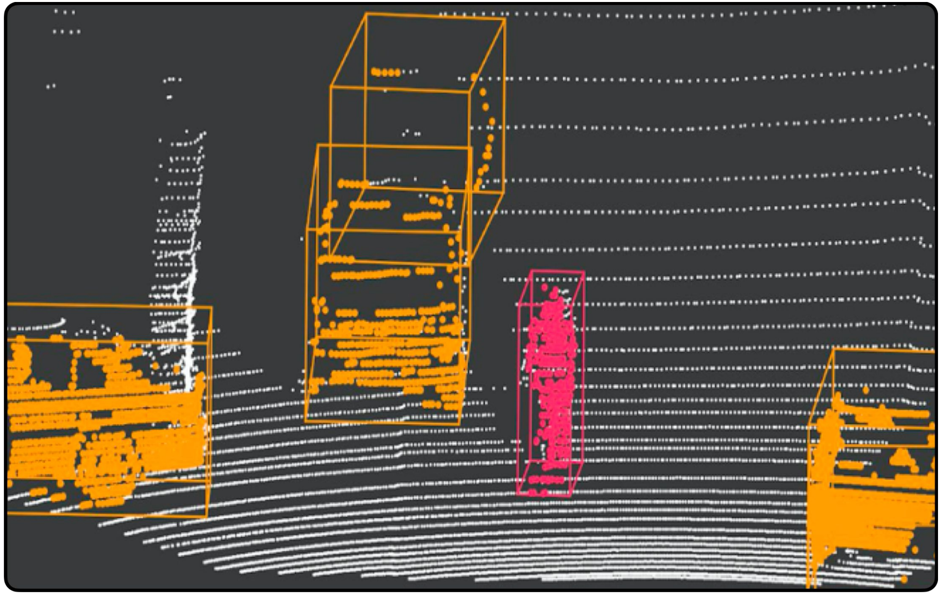
Matthew Millendorf

MIT DRIVERLESS


DENSO
485
TUDelft
EAMDELFT

Goal #1: Multi-Sensor Perception - PointPainting

A method where information inferred from a camera is combined with a LiDAR point cloud to improve performance in 3D object detection.



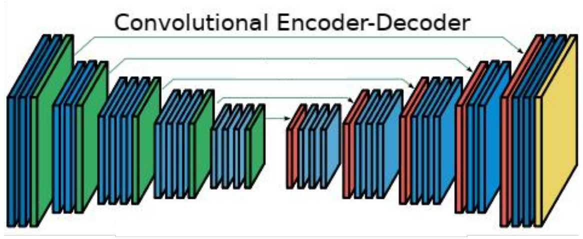
Vora, Sourabh, et al. "Pointpainting: Sequential fusion for 3d object detection." *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*. 2020.

MIT DRIVERLESS

Step 1: Inferring information from a camera

How is this done?

Camera Image



Semantic Segmentation Model

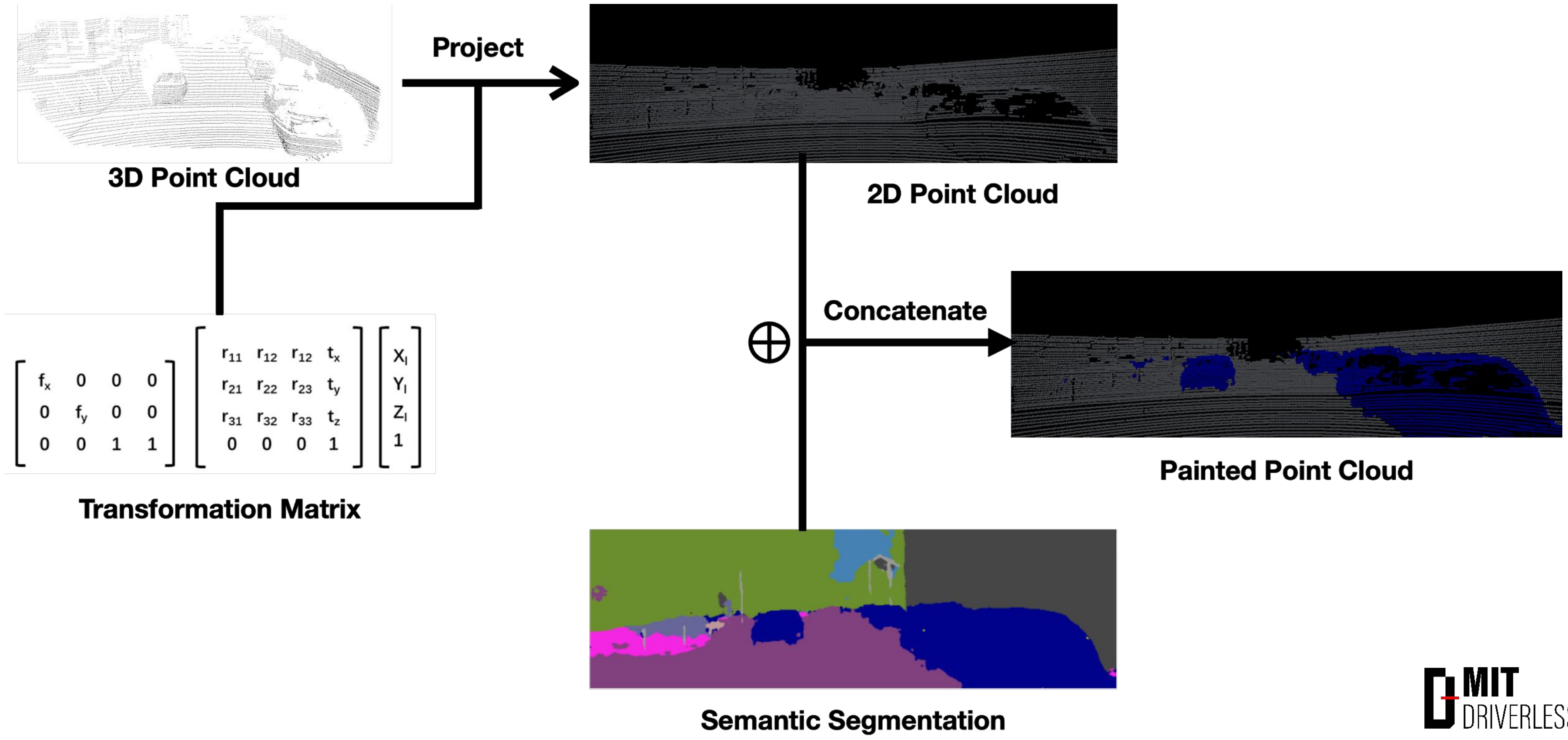
Pixelwise ‘class’ Scores

L.-C. Chen, et al. “Encoder-decoder with atrous separable convolution for semantic image segmentation.” *Proceedings of the European conference on computer vision.* 2018.

MIT DRIVERLESS

Step 1: Inferring information from a camera

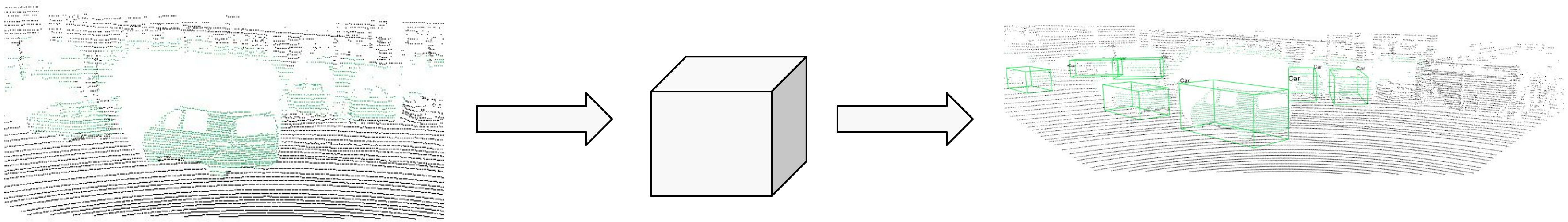
Step 2: Combine with LiDAR Point Cloud



MIT DRIVERLESS

Step 2: Combine with LiDAR Point Cloud

Step 3: Develop 3D Object Detection Model

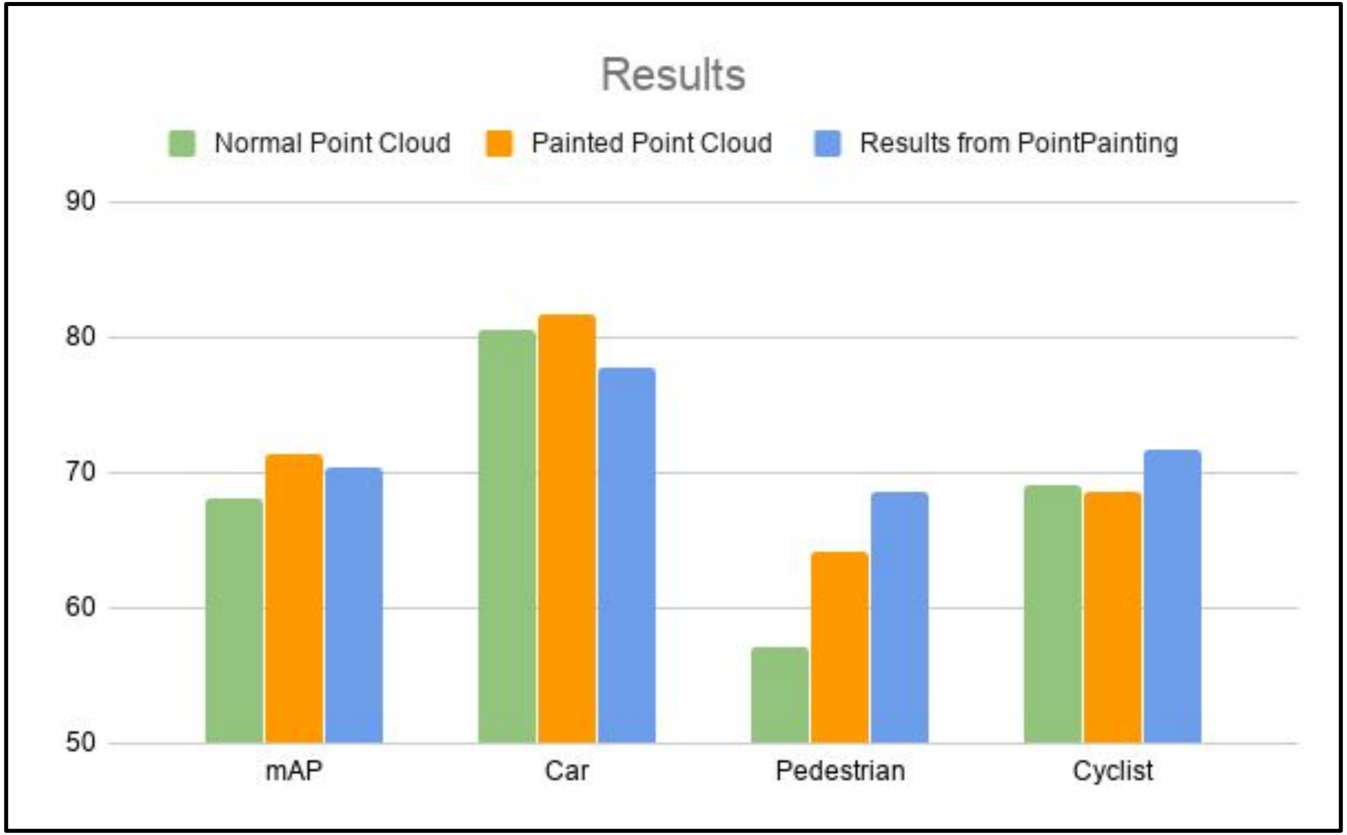


‘Painted’ Point Cloud

3D Object Detection Model

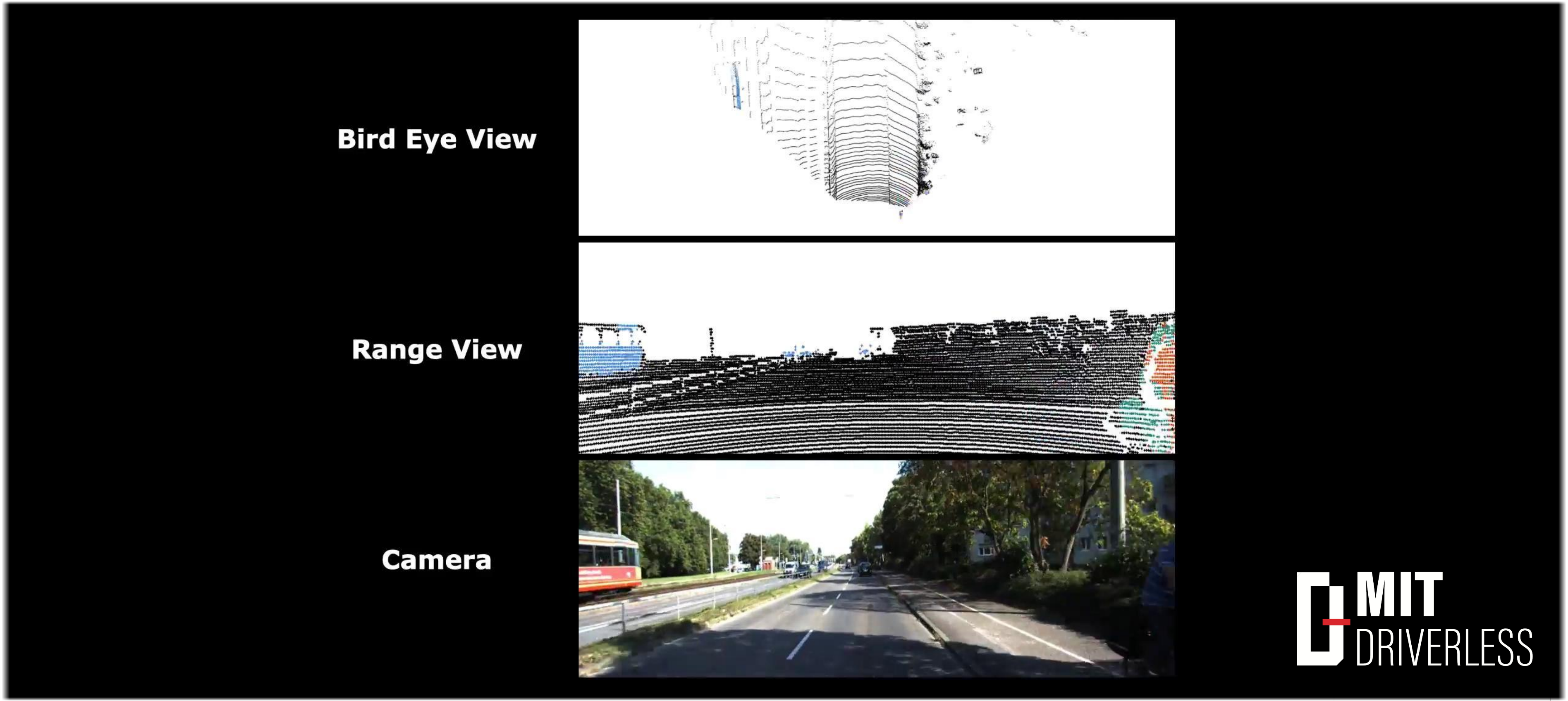
Detection of objects in point cloud

Results

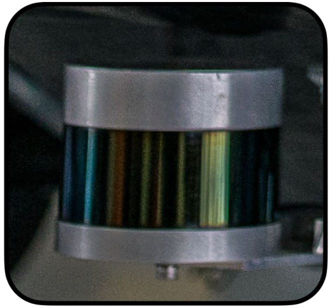


MIT DRIVERLESS

Conclusion: Putting it all together



Intuition for Goal #2



Camera: 37 hz

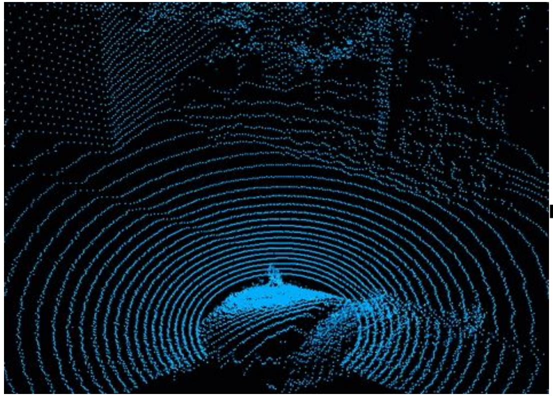
LiDAR: 10 hz

50 mph

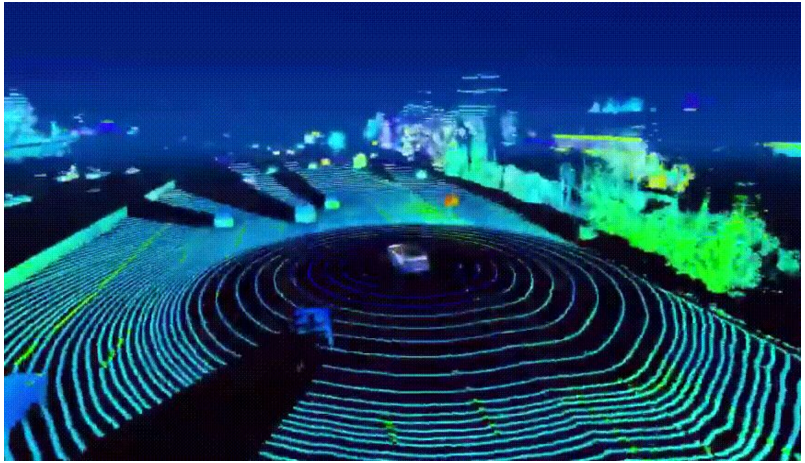
200 mph

Goal #2

37hz

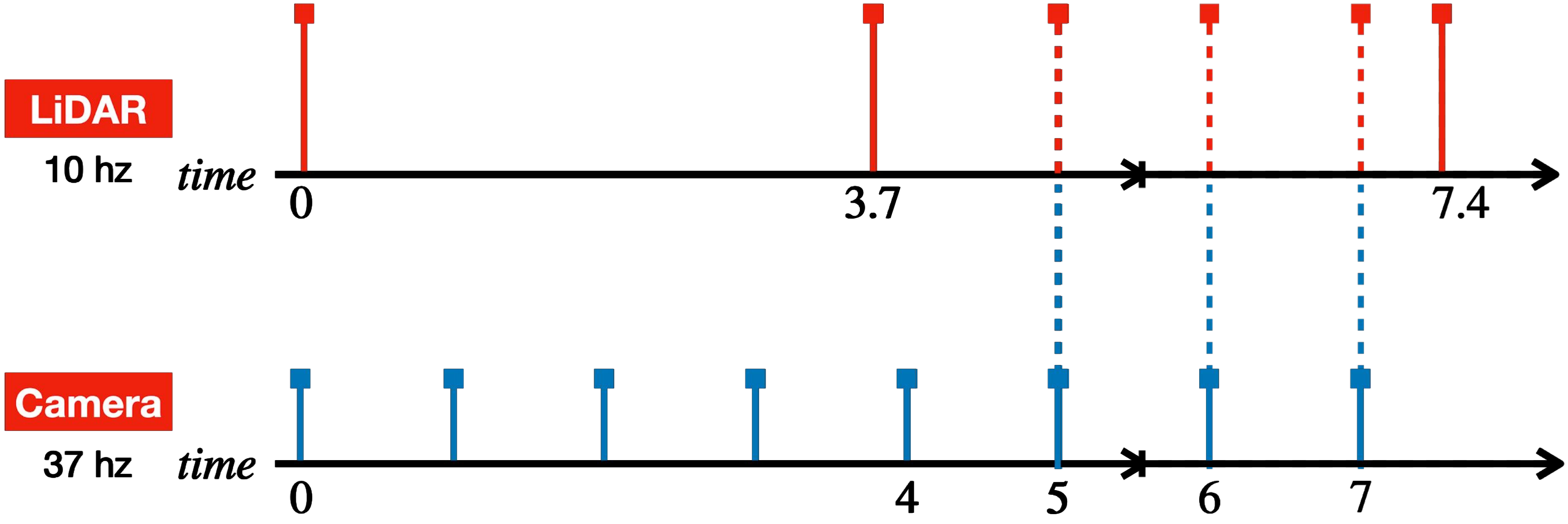


10hz



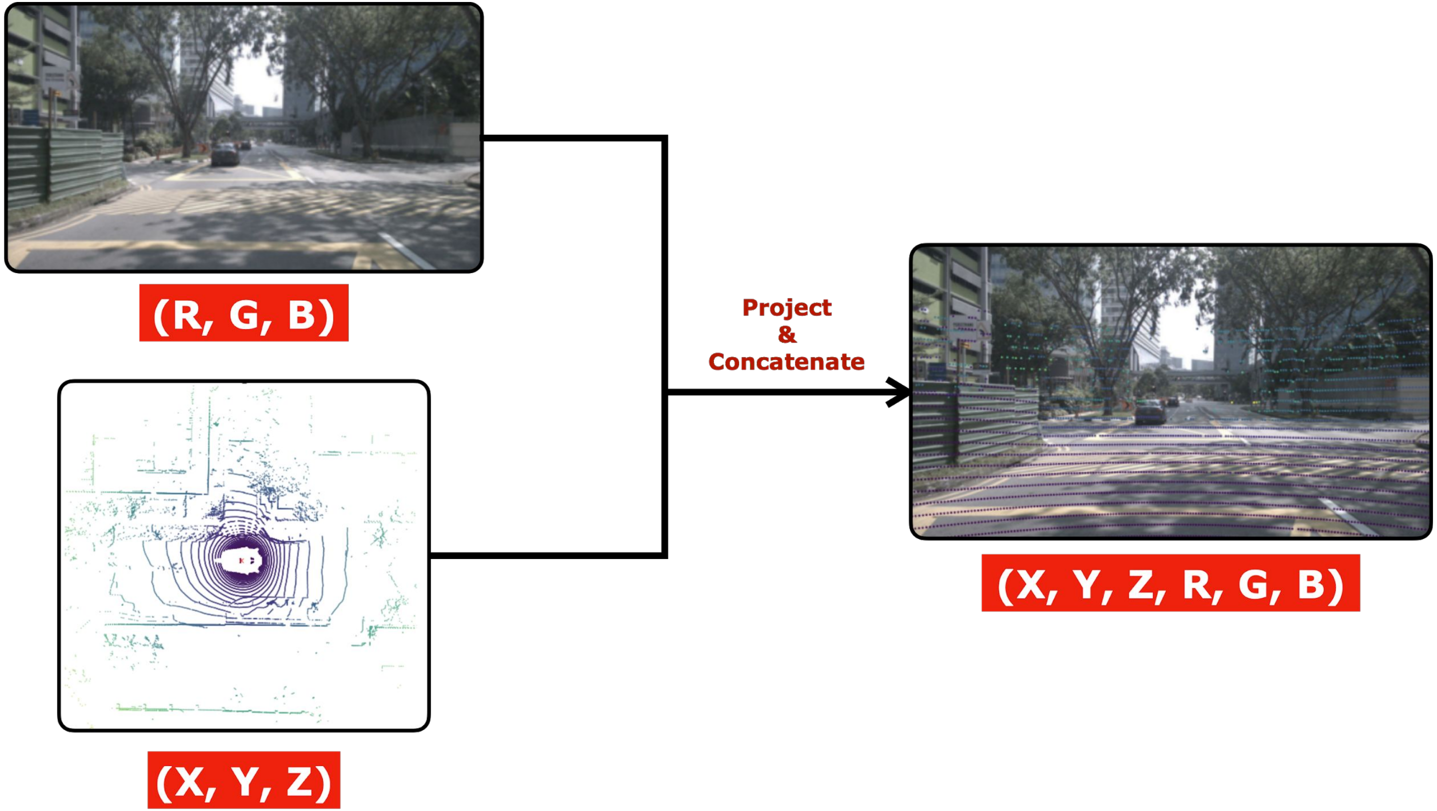
Goal # 2: Faster LiDAR

Goal #2: Faster LiDAR



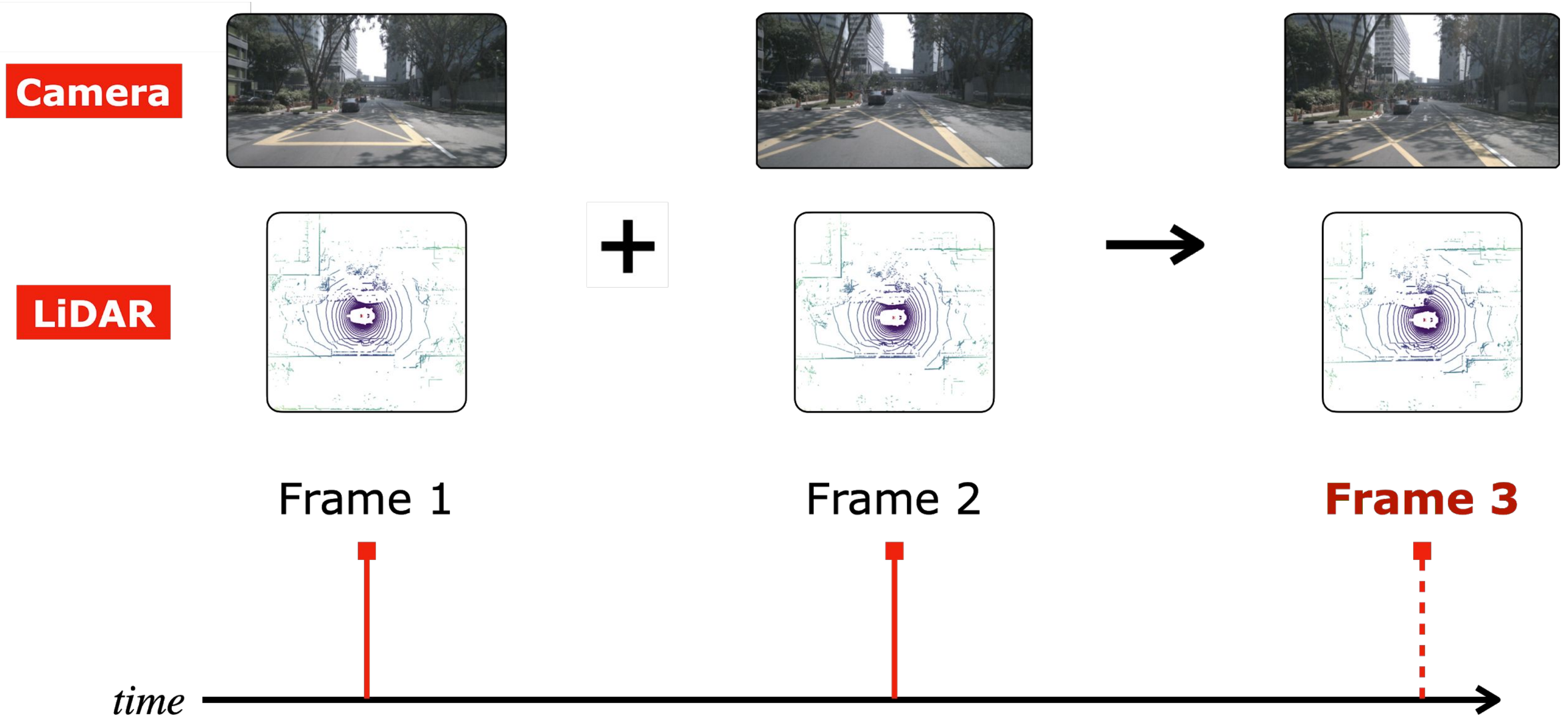
MIT DRIVERLESS

Design of Faster LiDAR



MIT DRIVERLESS

Design of Faster LiDAR



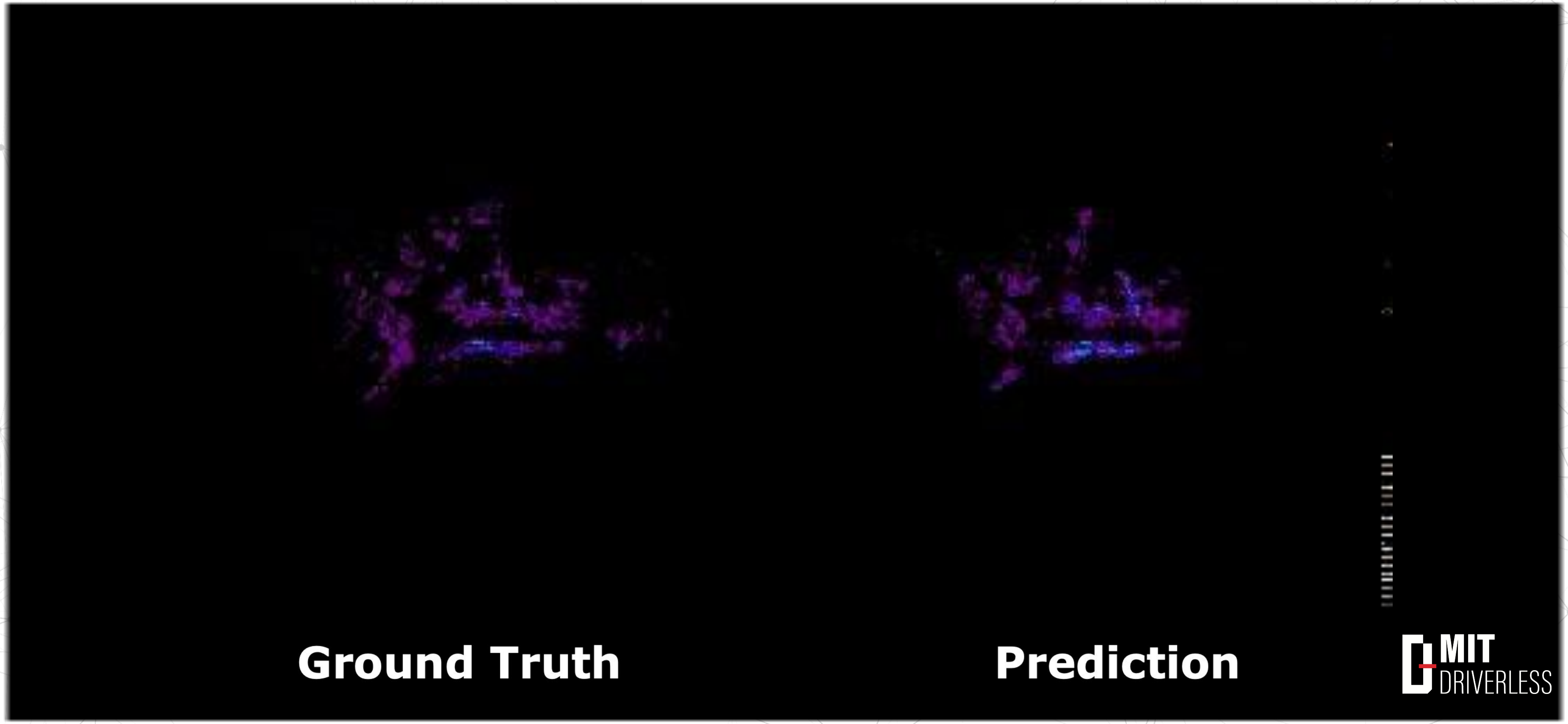
MIT DRIVERLESS

Performance

Predictive Model	MSE	FPS(Hz)
Single Input	1.861460	22
Double Input	1.060083	16

MIT DRIVERLESS

Visualization



Discussion on Future Improvement

- Use more history information to predict future frames
- Improve efficiency
    a. Convert code to C++ for faster inference
    b. Neural Network Channel Pruning
    c. Int 8 Quantization

The Path Forward...

- Use camera frame to correct our 3D predictive model
- Multi-agent object tracking
- Dual-redundant perception system

MIT DRIVERLESS

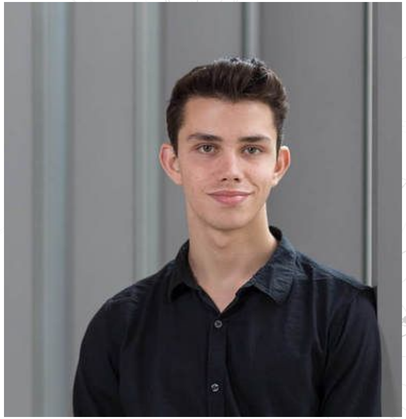
Planning & Controls

Interested in hearing more? Join the Q&A by clicking

Planning and Controls Team

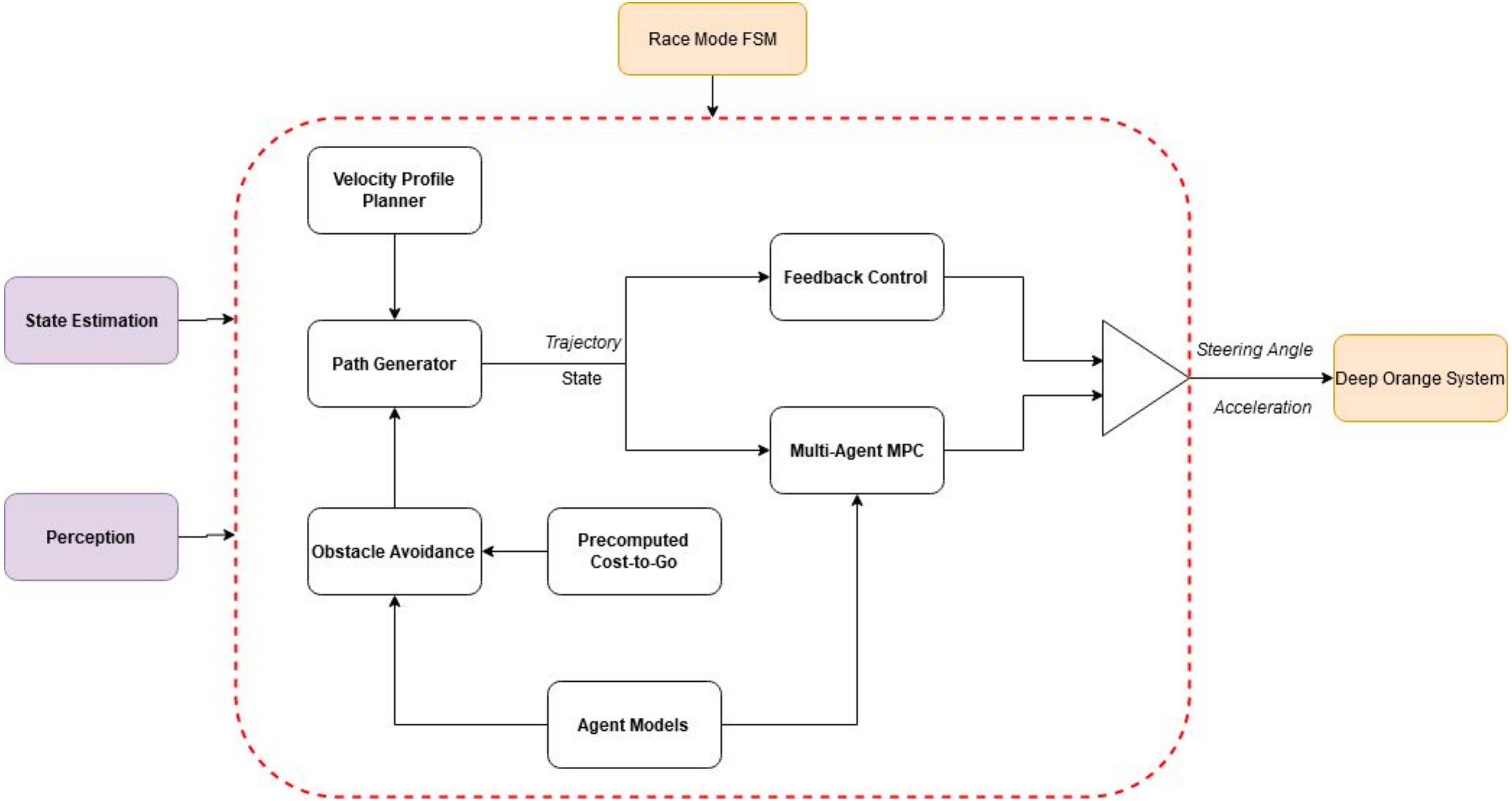
Arjun Albert

Mike Schoder



Adam Potter

Planning and Controls

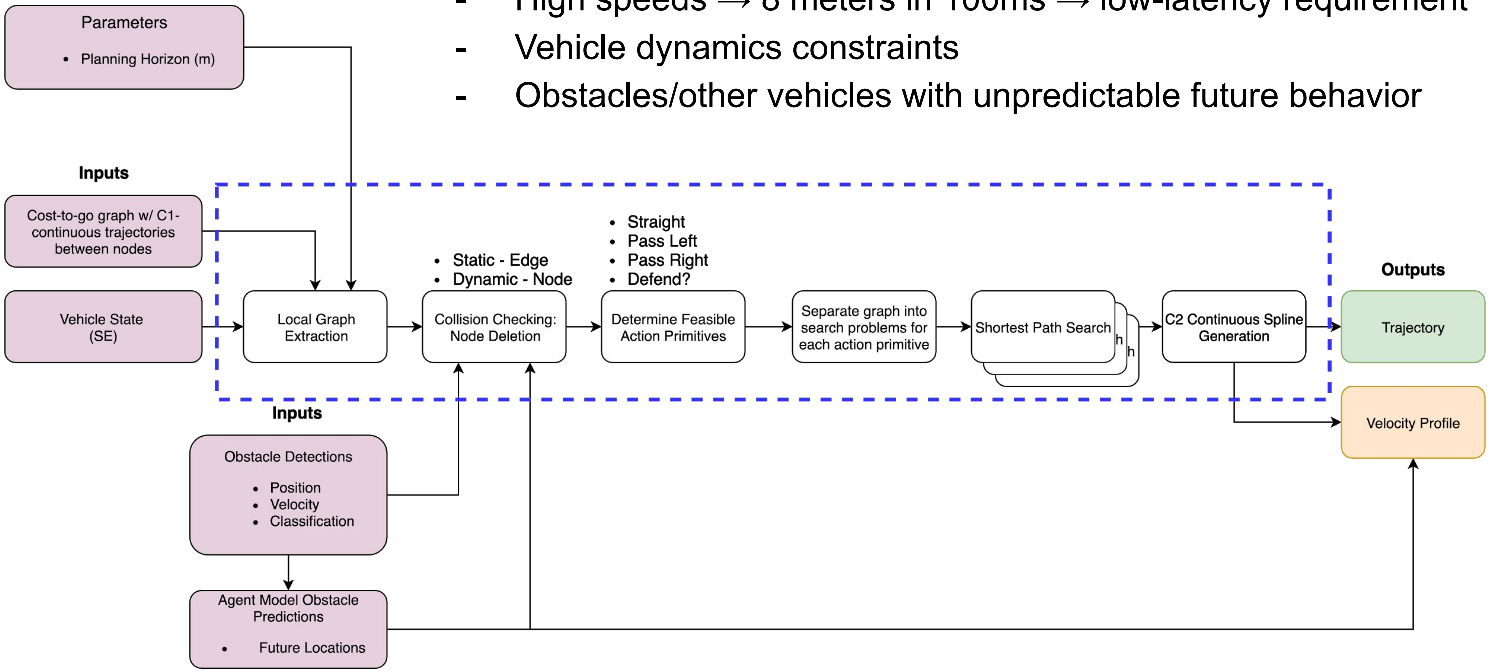


Planning

Goal: Generate the most efficient, obstacle-free trajectories based on the determined race strategy. Operate at low-latency and pass trajectory onto controls stack.

Constraints:

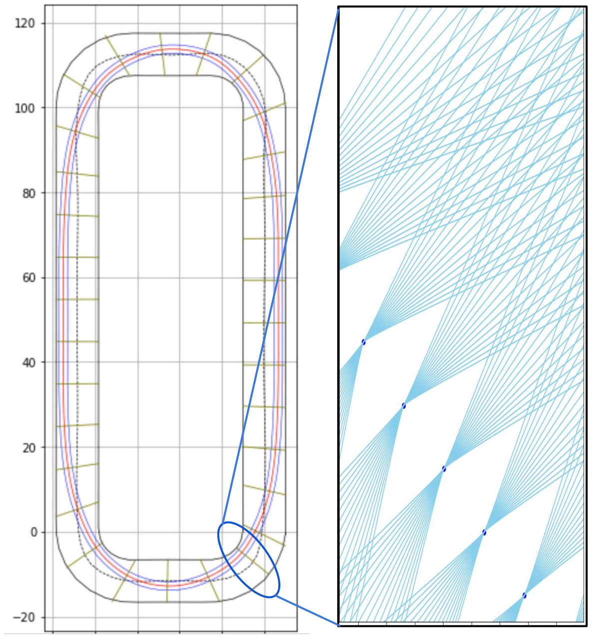
- High speeds → 8 meters in 100ms → low-latency requirement
- Vehicle dynamics constraints
- Obstacles/other vehicles with unpredictable future behavior



Planning framework and architecture heavily inspired by Technical University of Munich work for Roborace: [Multilayer Graph-Based Trajectory Planning for Race Vehicles in Dynamic Scenarios, Stahl et al., 2019]

MIT DRIVERLESS

Cost-To-Go Map

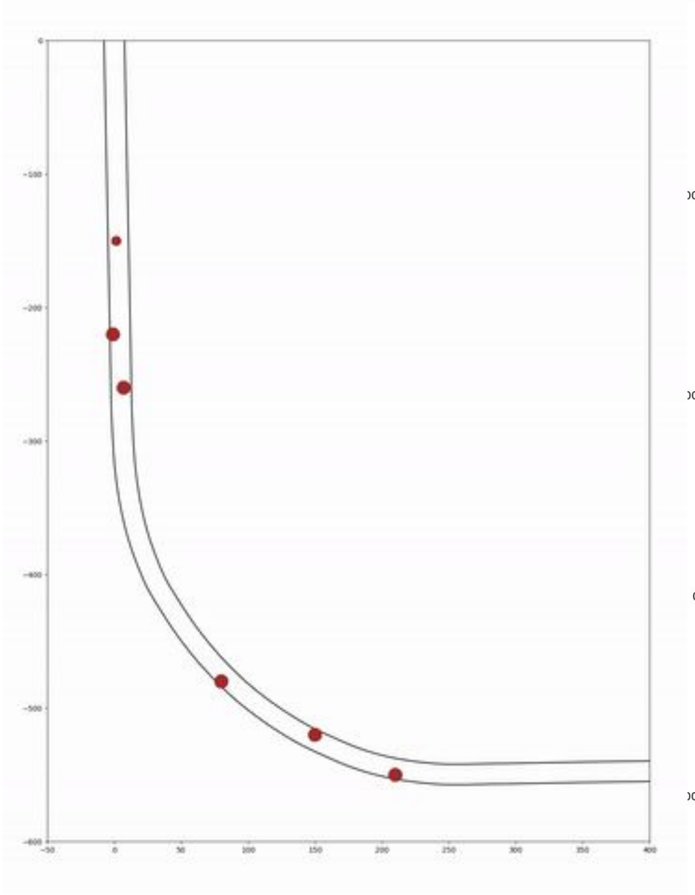


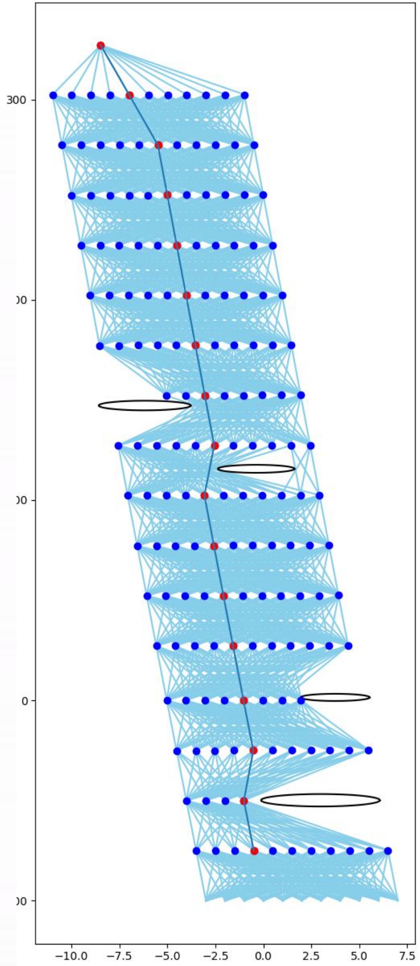
General Idea: IMS racetrack is a highly-constrained environment with many fewer variables than real-world driving. We can pre-compute C1 trajectories and hold cost and position information for a reasonably fine state space discretization in memory, which speeds up the online computations for path planning -- latency is critical at high race speeds

Method:

- Frenet-frame track lattice discretization
- Calculate optimal raceline based on simulated vehicle dynamics
- Cost assigned to each edge based on distance from raceline and curvature
- Graph held in memory; planner extracts segments for local search and planning

Obstacle Avoidance





Pipeline

- Local graph extraction from pre-computed global cost lattice
- Obstacle-based node & edge deletion
- Minimum cost path search
- Curvature-continuous (C2) trajectory generation
- Velocity profile

Current runtime latency: < 50 ms

Upcoming in Path Planning

- Higher level action primitives & decision-making (pass, defend, hold)
- Latency reduction efforts for graph collision checking
- Tighter integration with perception & state estimation
- Testing as part of full stack in simulation

MIT DRIVERLESS

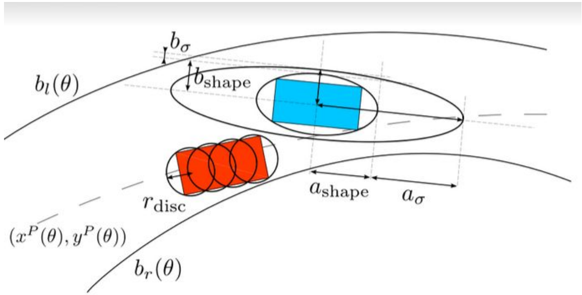
Multi-Agent Models

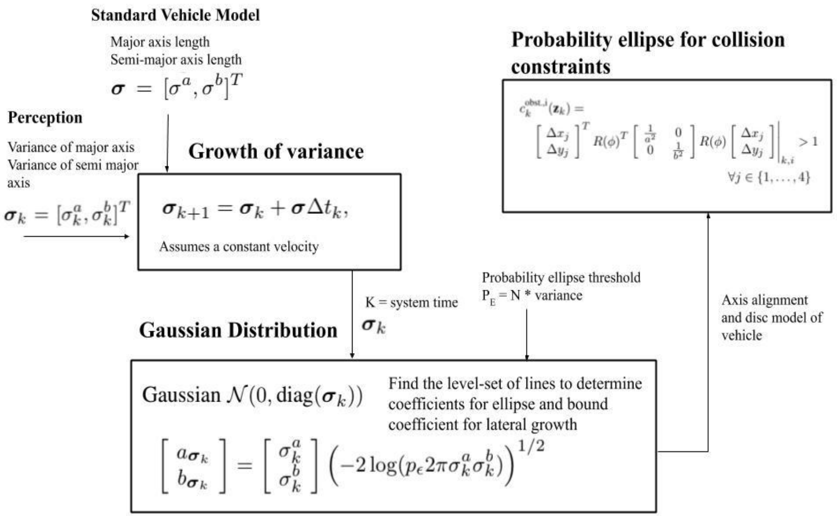
Agent Model Goals

- Give ego vehicle multi agent collision constraints to multi-agent MPC
- Create occupancy probability ellipse for obstacle avoidance

Considerations

- Growth in uncertainty over time
- Lane-keeping likelihood
- Determine boundaries at N-sigma confidence





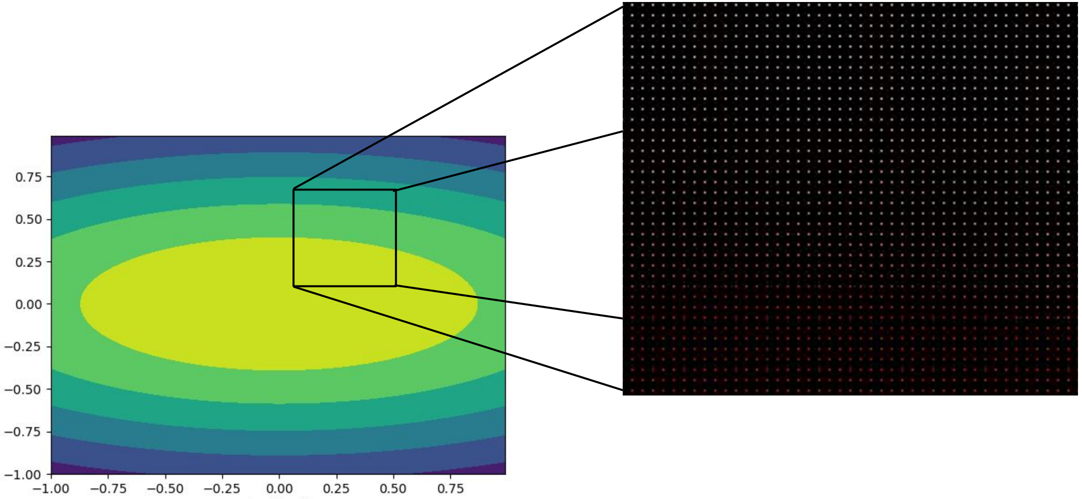
Schwarting, Wilko, et al. “Parallel Autonomy in Automated Vehicles: Safe Motion Generation with Minimal Intervention.” *2017 IEEE International Conference on Robotics and Automation (ICRA)*

MIT DRIVERLESS

Multi-Agent Models

Occupancy Probability Ellipse

- Probability density function feeds into obstacle avoidance
- Easy to integrate future states

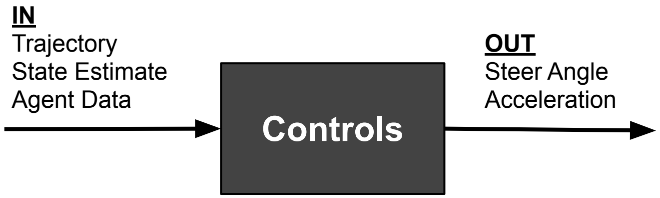


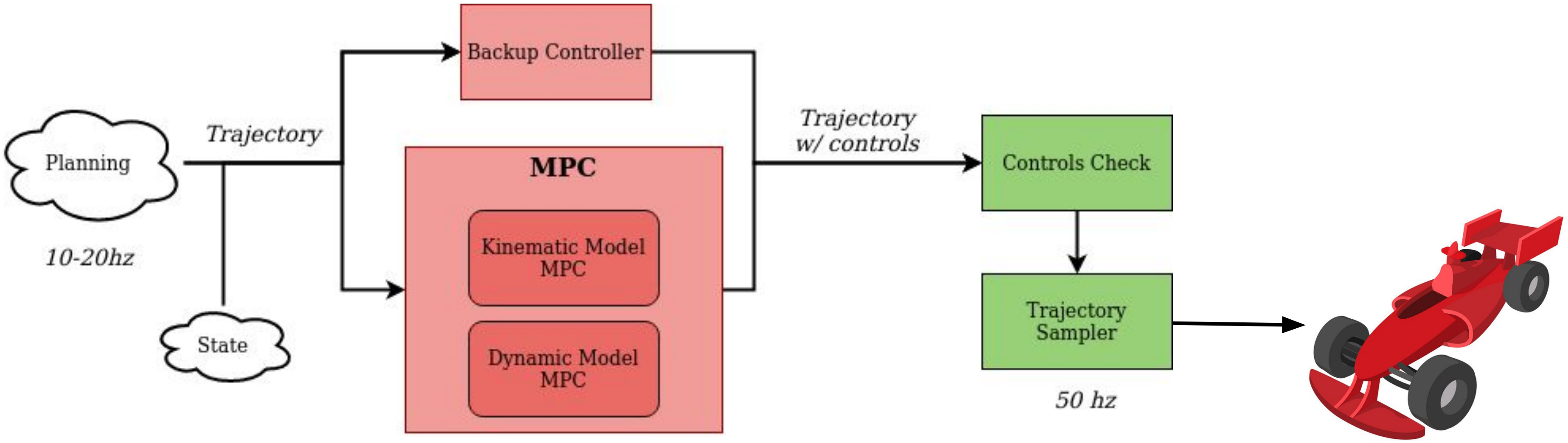
Upcoming for Multi-Agent Models

- Road agent trajectory prediction
- More optimized ego vehicle representation for Minkowski sum
- Testing and integration

MIT DRIVERLESS

Controls Structure



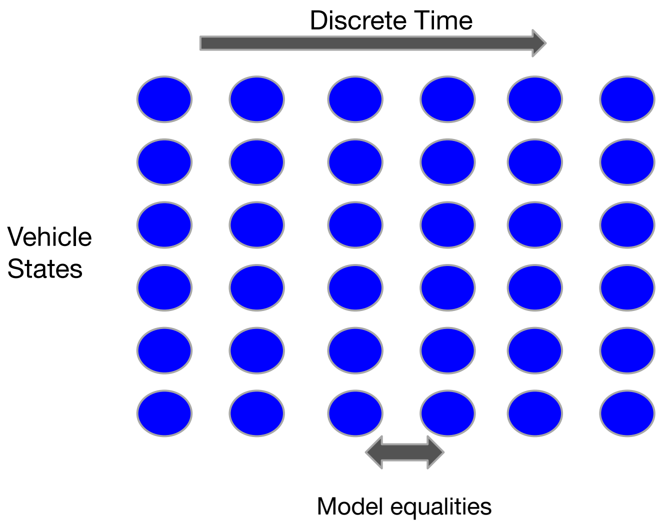


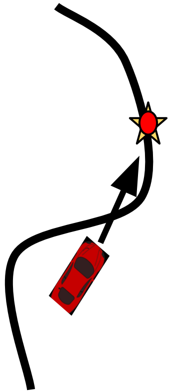
embotech

MIT DRIVERLESS

Controls: MPC Formulation

FORCESPRO

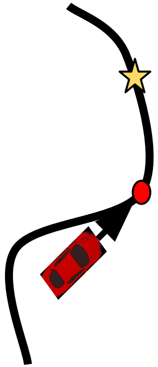




Position Following

Time parameterized

* Velocity tracking implied



Path Following

Arc Length parameterized

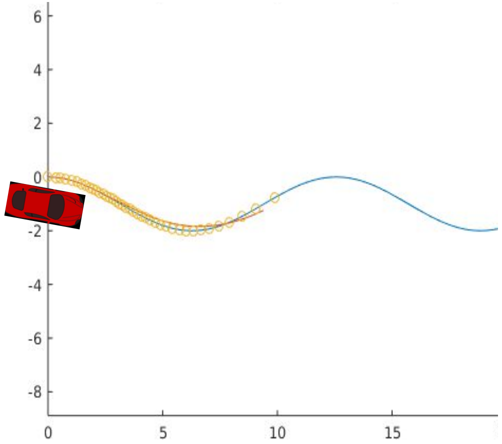
* Velocity tracking in addition

$$min \sum W_c \cdot E^2_{crosstrack} + W_l \cdot E^2_{lag} + W_v \cdot E^2_{velocity} - W_s \cdot V$$

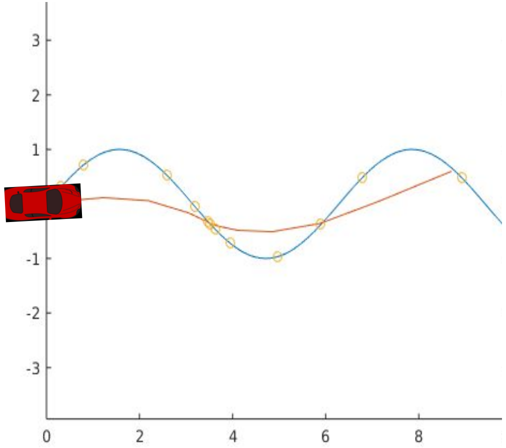
MIT DRIVERLESS

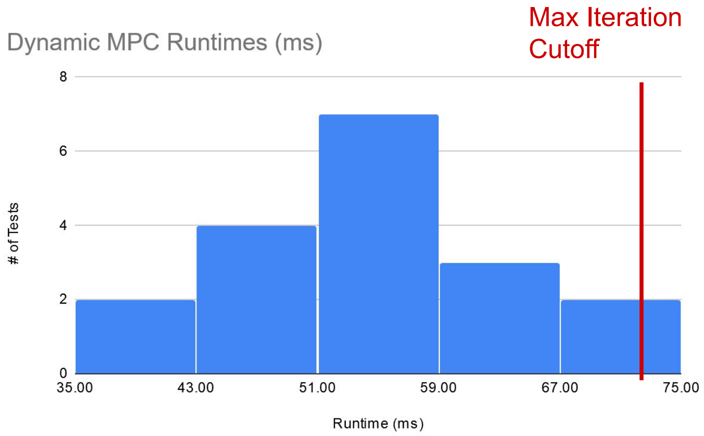
Controls: MPC Progress

Feasible



Infeasible





Constraints:

Agents collision *Control Limits* *Vehicle Slip*

Upcoming for Controls

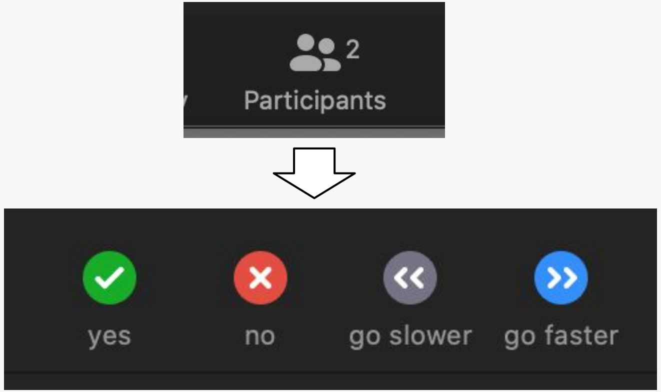
- Delay integration
- Reinforcement learning for better model
- Tire wear, fuel consumption, etc
- More advanced agents

Q&A in Breakout Rooms

Want to join the team?
Want to sponsor?
Want to follow us?

driverless.mit.edu

Q&A: Indicate in Zoom which room you want to join



General

Architecture

Perception

Plan. & Controls

MIT DRIVERLESS