INDY AUTONOMOUS CHALLENGE

WUT Driverless

Warsaw University of Technology

About us

We are a team from Warsaw University of Technology in Poland originating from **Students' Robotics Associacion** (**Koło Naukowe Robotyków KNR**). The team was founded 3 years ago by two robotics geeks, and since then has grown to as many as 25 students from many fields including Software, Electrical, and Mechanical Engineering. We make small scale cars and develop software for some of the most popular autonomous car racing challenges such as: Carolo-Cup, F1/10, and IARRC. Some of our biggest achievements include 1st place at IARRC 2018 and 2nd place at Carolo-Cup 2020. The project has been a source of multiple scientific papers and presentations on international conferences and meetups.

Our plans for IAC

We see Indy Autonomous Challenge as the perfect opportunity to transition from mobile robotics to full-fledged automated (race) cars. As a highly software-focused team, we are very fond of the competition model, in which participants are mainly responsible for autonomous system implementation, without the need to develop their own hardware platform. Furthermore, we are confident that a significant part of the experience and codebase accumulated over the years of autonomous racing with 1:10 and 1:8 scale vehicles can translate well into the task at hand.

Financing

At the time of writing, we have funding from a program led by Polish government, which allows us to cover travel costs only for the first hackathon. This means we need to secure first sponsorships early on in order to be able to take part in remaining Round 3 activities. We hope to achieve that by using materials gathered during Round 2 in order to demonstrate prospective partners our ability to successfully compete in the challenge. We also want to ensure good coverage of our presence on the events in local media, e.g. by recording a video documenting our on-site work.

Software Architecture

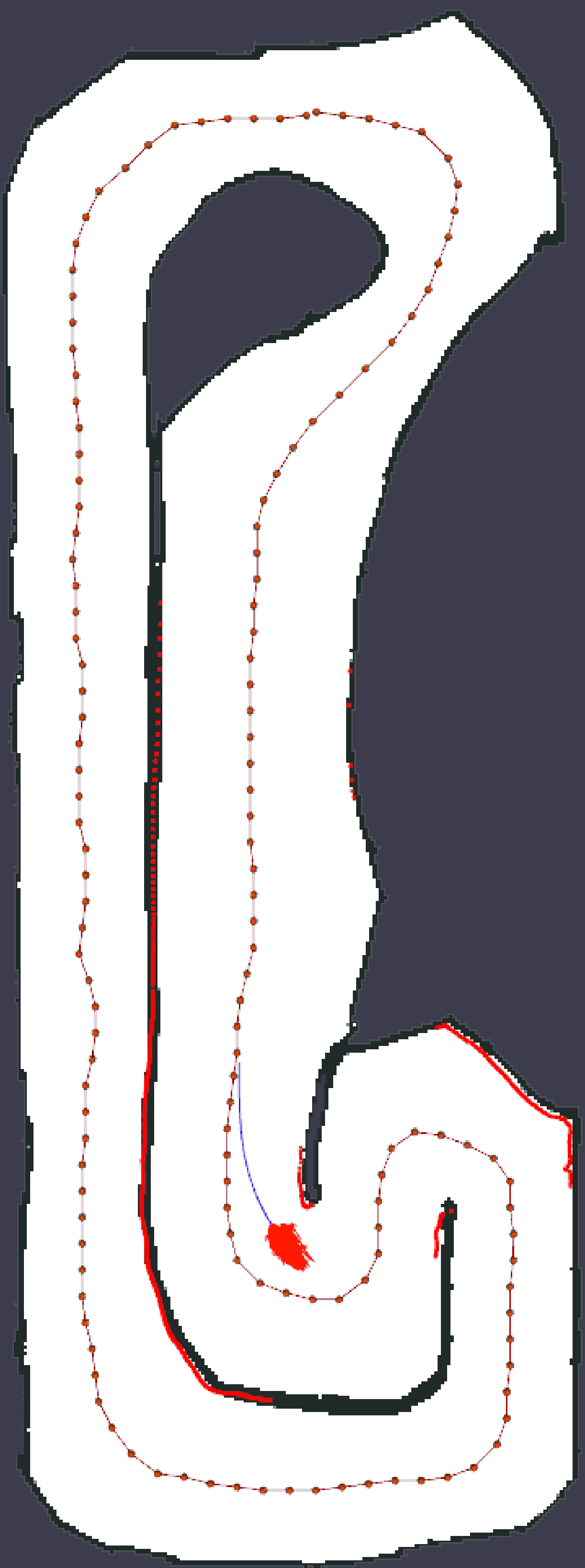
Our approach to automated vehicle software development

One of the most important principles in automated vehicle software is modularity. By dividing a complex system into a few smaller programs we gain a great amount of robustness and flexibility. For that purpose we use Robot Operating System, which is a common and verified approach in the robotics community. The workflow we have worked out allows us to test all parts of the system both individually and as a whole. The architecture of our autonomous system lets us switch between different solutions without altering the main structure, giving us plenty of possibilities for future additions.

Mapping

How does the track look like?

We have experience in 2D mapping based on LIDAR data. We use 2 popular open source solutions: GMapping and Google Cartographer and due to barriers on both sides of IMS we could potentially use them during this challenge.



State Estimation

Where are we?

The main source of car's state information are sensors. They provide us with accelerations, rotations and speed of the vehicle. However, there is one more thing we need to know, which is our position on the track. For this GPS is a tool most designs use. For better accuracy there are also algorithms like AMCL that incorporate laser scans for precise positioning. It is a solution we already use and feel comfortable with.

Perception

Where are the other cars?

Perception is an extremely important aspect of an automated vehicle design. No planning and no control can be done without proper information about our surroundings. It is a crucial part in the aspect of safety as well. Technologies that we have used in this field include camera image recognition and 2D LIDAR scan processing, but we look forward to be able to test different solutions like 3D LIDAR, RADAR or ultrasonic sensors.

Planning

Where do we want to go?

Planning module is strictly bonded to control module and sometimes they appear as one big part in a system, for example in Model Predictive Control setting. While having a plenty of advantages, the greatest drawback of such approach is large computation time resulting in slow response to volatile environment. That is why we intend to use MPC as a planning module, with additional high frequency feedback loop responsible for actuating the vehicle.

Control

How to get there?

This lowest-level module is tasked with realizing trajectories requested by the planning subsystem. Exact controller design is yet to be defined, though it should be able to handle well both lateral and longitudinal dynamics of the vehicle system. Literature offers multiple examples that can be used as a reference.

Round 2 Demonstration

The first step towards full-scale autonomous racing

e-MaksPower is a fully electric prototype vehicle designed to be used by people with locomotive disabilities. Since Dec 6, 2019 it is in regular use of Max, a 10-year-old originator of the project suffering from muscular dystrophy. Operation of the car is enabled by a joystick connected to an internal drive-by-wire system. Thanks to courtesy of e-MaksPower team, as well as of Max himself with his parents, we will be able to build on existing capabilities to implement a range of automated driving functions.