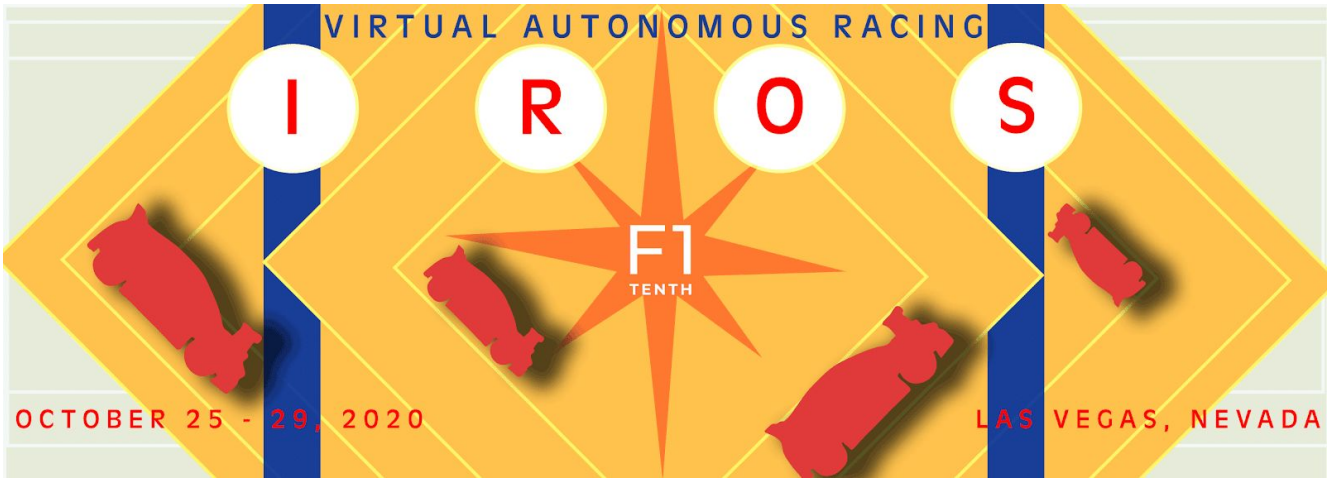


F1TENTH @ IROS2020

8th F1TENTH Autonomous Grand Prix

https://f1tenth.org/iros2020.html

https://iros2020.f1tenth.org/

Table of Contents

1. Event Timeline

Last edited on 2020/08/11 09:27 EDT

2. Introduction

The 8th F1TENTH Autonomous Grand Prix is a virtual race. Just like a real Grand Prix, teams first qualify in order to determine seeding and then compete head-to-head in an intense battle of algorithms. Since vehicles and hardware are standardized, teams must develop robust perception, planning, and control algorithms that can deal with the uncertainties of a new track and new competitors. The qualifying phase is a timed trial. The Grand Prix phase pits virtual competitors against each other on the same track. The event timeline can be seen in the previous section. When you are ready, register to enter your agent in the competition! Gift cards will be awarded to the winning teams.

3. Overview

Please read through this document carefully from start to finish. If you have any questions, ask them on the #iros2020 slack channel. The FAQ will be appended as questions are received.

The submission portal, leaderboard, and live stream will be available at http://iros2020.f1tenth.org/.

There are two phases to the race.

1. *Qualifying:* Timed Trials
2. *Grand Prix:* Head-to-Head

The results of **Qualifying** will be used to seed the brackets for the **Grand Prix**. In order to qualify for the Grand Prix, you must be able to complete the **obstacle avoidance test**. This is a binary qualification.

1. Orientation 1, August 14th

During orientation, we will go over the simulator environment, demonstrate what the race will look like, and answer any questions. It is useful to go over this document prior to attending the orientation in order to maximize the benefits. This orientation will be recorded and the link will be added afterwards. There are development tips in the presentation as well!

2. Orientation 2, September 2nd

During this orientation, we will go over any new environment updates and answer questions.

3. Overview and Race Track Released, October 12th

During this meeting the map of the race track will be released. We will also go over submission and answer any questions. After the meeting, you are free to practice and tune your algorithm.

The submission window will open at **13:00 EDT on October 13h**. Make sure to submit your code before the submission window closes on **October 20th at 20:00 EDT**. Note that registration will close on **October 20th at 20:00 EDT**. You **MUST** be registered in order to submit.

4. Qualifying Ends, October 20th

You **MUST** be registered in order to receive your unique submission passcode. Once the window closes, we will spend the next few days determining the results of the timed trials and invite teams that qualify to submit for the Grand Prix.

5. Grand Prix Begins, October 21st

The seeded brackets for head-to-head will be live on http://iros2020.f1tenth.org/. The submission window for the Grand Prix opens at **13:00 EDT**.

6. Grand Prix Ends, October 26th

The submission window for the Grand Prix closes promptly at **20:00 EDT**.

7. Race Day!!! (Time TBA)

The head-to-head racing will be streamed live on http://iros2020.f1tenth.org/.. The schedule will be posted accordingly. The stream can also be viewed at https://www.twitch.tv/f1tenth.

4.How to Compete

1. Developing and Testing using the F1TENTH Simulator

Follow the README and install the F1TENTH gym here.

2. Qualifying Evaluation

Qualifying evaluation consists of two tasks. The same submission needs to be able to compete in a time trial on the map without any obstacles, and finish the laps without collision in the same map with obstacles that are unknown prior to run time. A live leaderboard will be maintained to show the lap times, and a table will also show the result of the obstacle avoidance task (pass or fail).

3. Grand Prix Evaluation

Grand Prix evaluation is a head to head tournament seeded by the time trial leaderboard from Qualifying. Only submission that passed the obstacle avoidance test will be able to participate in the tournament. The map for the Grand Prix will be the same for Qualifying.

5. Submission

The submission portal, leaderboard, and live stream will be available at http://iros2020.f1tenth.org/.

You **MUST** be registered in order to receive instructions on how to register. A passcode along with submission instructions will be sent to you shortly afterwards. This passcode is required when submitting your code. The last day to register is **October 20th at 20:00 EDT**

Requirements:

1. ROS package as a github repository
2. Launch file or bash command to start code
3. README
4. Unique submission passcode

6. Rules

<Subject to change, F1TENTH reserves the rights of final interpretation>

1. General

a. The map used for all evaluation tasks will be the same (the map for the obstacle avoidance task will have added obstacles).

b. All evaluations will be two laps.

c. F1TENTH reserves the right to reject any submission that we deem illegal due to circumstances such as exploiting the simulation environment.

d. F1TENTH reserves the right to assign blame in the case of agent collision in the head to head tournament. The justification will follow section 4 of the rules.

2. Timed Trials

a. If an agent can finish without collision with the environment, it'll receive a lap time and update its ranking on the leaderboard.

3. Obstacle Avoidance

a. The location and the size of the obstacles will not be revealed prior to submission. Some example maps with obstacles will be provided.

4. Head-to-Head

a. Seeding
   i. The elimination bracket of the tournament will be filled by submissions that passed the obstacle avoidance test and seeded by the result of the timed trials. For example, with 8 teams, the bracket of the first round will be (#1 vs. #8, #2 vs. #7, #3 vs. #6, and #4 vs. #5).

b. Overtaking, car control and collisions
   i. Overtaking, according to the circumstances, may be carried out on either the right or the left. Manoeuvres liable to hinder other drivers, such as deliberate crowding of a car beyond the edge of the track or any other abnormal change of direction, are strictly prohibited. The stewards will have the final say in whether a driver is in violation of the rule.
   ii. Causing a collision (decided by the stewards) for more than 3 times will result in automatic disqualification.
   iii. The stewards reserve the right to assign blame to both agents in some cases of collisions. In this case, both participants lose a chance to cause a collision (out of 3 times).

7. FAQ

1) I have a question!

   a) Read this FAQ. If your question isn't here, ask on the #iros2020 slack channel.

2) I don't want the moderators to see my source code or my build process is complicated.

   a) You can create a docker image/container that only includes the binary of your source code and make sure to provide the proper launch command, and proper communication between your container and the provided sim container.

3) I have proprietary/dedicated hardware, and my code runs fast on it but slow on your server.

a) Everyone will run on the same hardware. There is no open class in this virtual race.

4) I need a camera sensor stream but I'm not provided with one.

    a) Short answer is no, only laser scans are available. As an option, you can rasterize the lidar scans and retrain. We're working on providing a camera stream in a future version of the simulation environment and we might provide a camera stream in future virtual races but there are no plans to provide one for IFAC.

5) What happens if I crash during the race?

    a) Please consult the rules section.

6) Is slipping implemented in the simulator?

    a) Yes.

7) Regarding computer power, what is the model of the hardware that is evaluating the car?

    a) Both cars run on the same machine at the same time. Each agent in its own container. Reasonably powerful desktop running ubuntu (Ryzen 9 3900X, RTX 2080, 32G RAM). Typically stronger than the TX2.

8) In the 2 car scenario, how are the drive messages from both sides handled?

    a) There is a message sync node implemented with message_filters in ROS that runs in the backend container.

9) What happens when one agent sends drive commands, the other is still running?

    a) There is a small time window in which the other agent can still send drive messages. If it is not able to send it in time, the last drive message will be used.

10) Is there the possibility to use hardware acceleration?

    a) The server will have a single RTX 2080. To use the GPU in your solution, you'll HAVE TO create your own container based on the official Docker images provided by Nvidia for TensorRT or CUDA. And use nvidia-docker https://github.com/NVIDIA/nvidia-docker so that docker can interface with the hardware.

11) Is there a way for the agent to know which phase of the race it is currently participating in? In particular, will there be an easy way to distinguish between Phase 1a (timed trial, no obstacles) and Phase 1b (collision avoidance, unknown static obstacles)?

a) It is correct that you'll be able to use two different solutions for phase 1 and 2. For phase 1a and 1b, we felt like it is pretty simple to detect whether or not you're racing on a track with obstacles so we left it as the same submission. We won't be changing the message header this close to the competition, but for sure it'll be taken into consideration as future improvement of the sim. In the long run this will require more consideration. We will see how the competition goes. The point of the obstacle phase is to make sure your controller reliably avoids obstacles. Changing the controller kind of defeats the purpose, but we won't change the rules now.

12) How are loading times/initialization delays being handled?

a) The timer only starts once both agents are publishing drive commands.

13) Are messages sent over the `/odom` (and `/opp_odom`) topics in vanilla ROS nav_msgs::Odometry format? Is the data itself ground truth odometry, i.e. unperturbed by noise? Are the pose and twist covariance fields not used?

a) Yes. Yes.

14) Will the parameters of the vehicle dynamics simulation, referring to the ST/KS models, as described by M. Althoff, which are being used in the F1TENTH simulator, remain the same as specified here,? Are these constraints used? Can we assume that `/scan` will consist of exactly 1080 (equidistantly spaced) range measurements, perturbed by Gaussian noise (mu = 0.0m, sigma = 0.01m), FOV=4.7 rad (=269,29 deg.) ...?

a) Regarding the dynamics of the vehicle, yes it'll be exactly the same. Regarding the scan, we don't recommend making the assumption since all the information is provided in the `LaserScan` message except for the noise. But yes, it will all remain the same.

15) Is the following true: An agent has no direct control over the vehicle's acceleration. Instead, it must post a "desired velocity" (via `ackermann_msgs::AckermannDrive::speed`), which is fed through a baked-in proportional controller (in the simulator, see:) to compute the acceleration? By looking at the protocol specification (racecar_simulator_standalone::StepRequest), only `ackermann_msgs::AckermannDrive::steering_angle` and `ackermann_msgs::AckermannDrive::speed` are being used and the remaining `ackermann_msgs::AckermannDrive` fields get ignored?

    a) Yes.

16) Why are constant random seeds (example) used for the random generators?

    a) This is a good catch. It's an artifact from our research when we were trying to keep results reproducible. But in future competitions it shouldn't be a constant random seed.

17) Is the second part of the qualification (Phase 1b, avoiding static obstacles) timed?

    a) It is not. There is no cap on the agent's velocity for phase 1b, but note that there will be a timeout if it is too slow. We will not be changing the sim at this point. The timeout is set to a reasonable amount of time of 120 seconds. Also keep in mind that we expect your submission to phase 1b to be reasonably similar to your submission to phase 2. The whole point have having phase 1b is to have your phase 2 submission pass basic tests. We didn't enforce having exactly the same submission because we feel like it's beneficial for the race to let participants to have more time for final tuning. We did state in the rules that we reserve the right to disqualify phase 2 submissions if it's deemed completely different from your phase 1b submission.

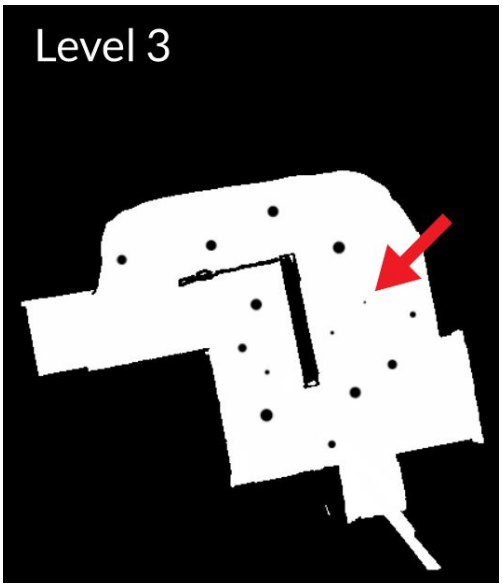
18) Let's assume we're using reinforcement learning to train our agent's drive controller. Naturally, we would tune our reward function (slightly) differently for Phase 1a (shorter lap times get rewarded), Phase 1b (safe driving gets rewarded) and Phase 2. At runtime, we would just switch out the trained controllers. Would you consider this approach acceptable? (In terms of the rules against "completely different" submissions.)

    a) Yes that's acceptable, phase 1a submission doesn't have to be close to the other ones. We're mainly concerned about using completely different approaches in phase 1b and phase 2.

19) How does the simulator determine when/if a vehicle has completed a lap?

    a) Each agent gets a line perpendicular to the track (or more so the centerline of the track) at their corresponding starting positions, and each agent has a counter that increments when the agent enters or exits a small area around that line. Agents start the race inside that area, so 2 counts (1 exit and 1 entry) is one lap of the race for that agent.

20) Will there be some constraints on the shape/size of the static obstacles? (For example, can we assume that obstacles will be convex?) Will there be very small obstacles, e.g. even smaller than shown on the "Level 3" image below (from the orientation presentation)....?



a) You can assume convex and not as small as in the image, they will be big enough that the lidar will be able to reliably detect them.

21) Is the following correct: the message sync is implemented using message_filters.ApproximateTimeSynchronizer. From what I remember, ApproximateTimeSynchronizer's registered callback is only executed when all (not any (!)) of the filter's inputs have changed (this behavior is inherited from TimeSynchronizer).Example: Agent #1 publishes drive messages at a rate of 1000Hz, while Agent #2 publishes drive messages at only 1Hz.This should lead to drive_callback() getting invoked at 1Hz (on every 1000th message from Agent #1 and every message from Agent #2), there is no "re-use" of old messages from Agent #2.

a) Yes. The public branch is a fast and dirty implementation so people can test out 2-agent races. The evaluation backend indeed will be using an implementation where if an agent is publishing too slowly an old message will be used

22) How would I go about synchronizing a received /scan message with the corresponding simulation timestamp? What the agent wants to know is: At what simulation time `t_sim` was this scan generated?

a) Currently, it is not possible. If you really want to do that there is an alternative. The scans and odom and race_info are all published inside

the same timer callback. You could potentially sync the scan with the elapsed simulation time in race_info if you want. The timestamps for those two should be very close. We don't have the concept of 'simulation time timestamp', those will have to be artificially made, either based on the wall time that the simulation started, or you'll have to use the elapsed time instead. I'll think about how to improve this in a future version, it's a good point that we should set up some sort of simulation timestamp

23) What are the characteristics that we can expect to be satisfied by the track? E.g. min and max track width, dead ends, track length, number of turns and curvatures, open track, smooth borders, etc.

a) The map which will be released on Sunday, July 5th, several days ahead of the actual submission will be extremely similar to the maps listed above. You can compute the minimum and maximum track widths from them and count the average number of curves if this is somehow an input to your algorithm. Montreal (`MTL.pgm`) is probably the most similar to what we plan. It is quite wide in sections to enable overtaking and has a chicane/hairpin type turn as well as a long straight. It is not guaranteed that you can pass to both the left and right of an obstacle. However, obstacles will not be a part of the base map, only a part of Phase 1b (runtime obstacle avoidance). We cannot give more details beyond that obstacles will be convex and detectable by lidar as the whole point is that you don't know about them ahead of time. As in past competitions there may be gaps in the boundary of the track, you should have a way to be robust to these. You should not expect that taking a measurement to each 'wall' and centering the car or pursuing the largest gap will work well out of the box (popular in the past). Similarly, preplanning a trajectory and tracking it, also popular in the past, won't work for phase 1b or 2 because of the presence of unknown until runtime obstacles. For gaps/geometries that would be troubling to reactive methods see `skirk.pgm` and see `columbia.pgm`. For geometries and features that would be troubling to precomputed trajectories, just draw some blobs in any of the maps. TL;DR: If your algorithm works on the above samples it will work fine on the competition track.

24) Could you please confirm (or correct) the following assumptions about the race format?

- The race (=Phase 2) will be held in a single-elimination bracket format (i.e. not Swiss or round-robin, where everyone races everyone else).
- Seeding for the first round is based on fastest lap time from the Qualification (Phase 1a, timed trial).
- Seeding for subsequent rounds are based on the winner's fastest lap time from the previous round?
- In any head-to-head pairing, the higher seed will get pole position (inside lane), the lower seed will be placed on the outside lane?

a. Each head-to-head pairing has been 2 races (each being multiple laps) with the start positions alternating (you would get 1 inside and 1 outside). Best win-rate advances, if it is a tie (likely) we will repeat 2 more races in same format. If still a tie, best lap time advances.

25) Will the seeds for Phase 2 get published before or after the submission deadline on July 13?

a) Yes

26) The agent will be allowed to continue racing, there won't be any time penalty, but the collision will count towards a total of three "strikes" (*), after which the agent is automatically disqualified. Three strikes per race, or three strikes total?

a) Note the 3 strikes rule applies to pairings (opp1 vs opp2) each of which will have at least two 'races' (see above).

27) Is an agent causing a collision (example: "overtaking" the opponent by simply driving through it) still able to win the head-to-head race?

a) In the 'overtaking' scenario you propose the opponent would not be able to win. We would reset and decrement the opponents points. Penalty points do not carry through.