# TAS Tasks: Autonomous Driving, Parking and Slalom

Laith Alkurdi

December 4, 2014

## 1 Main Task

The main task is to go around the third floor of LSR autonomously. The car should react to the hallway environment and the corner environment. During the final demonstration, the car should move completely autonomously with the input being only the initial position. Furthermore, there will be static obstacles introduced at random positions in the final demonstration to test obstacle avoidance. To win over the other teams, you should complete the lap in the fastest time without hitting obstacles or the walls (-30 seconds). Every group should present their contribution in a presentation before the demonstration. Some suggested start points are as follows:

1. Initialisation problem: usually the car needs an input for the initial position for the localisation algorithm. However, the laser scans or/and the wifi signal strength information could be used to get an estimate of the localisation/triangulation of the initial position.

2. The car should localise itself properly and the information of location as well as the IMU should be used to get an estimate of the states of position, velocity and acceleration as well as angular rotation using a (Kalman) filter.

3. The local planner that functions currently is sufficiently as it is. However, no control over the speed is currently available, a contribution for speed control could be implemented. Furthermore, the car should function differently around corners than it does in the hallway environment.

4. No dynamic model of the car exists. A contribution could be modelling the dynamics of the car in a Gazebo model.

# 2 Secondary Tasks

For the secondary tasks in general, as a tip, please record ROS bag files. This is because the following tasks depend on learning specific trajectories , in a first step, and trying to generalise them , in a second step when they are to be actually executed, in a manner such they are dependent on initial position and adaptive to environmental data (sensor information).

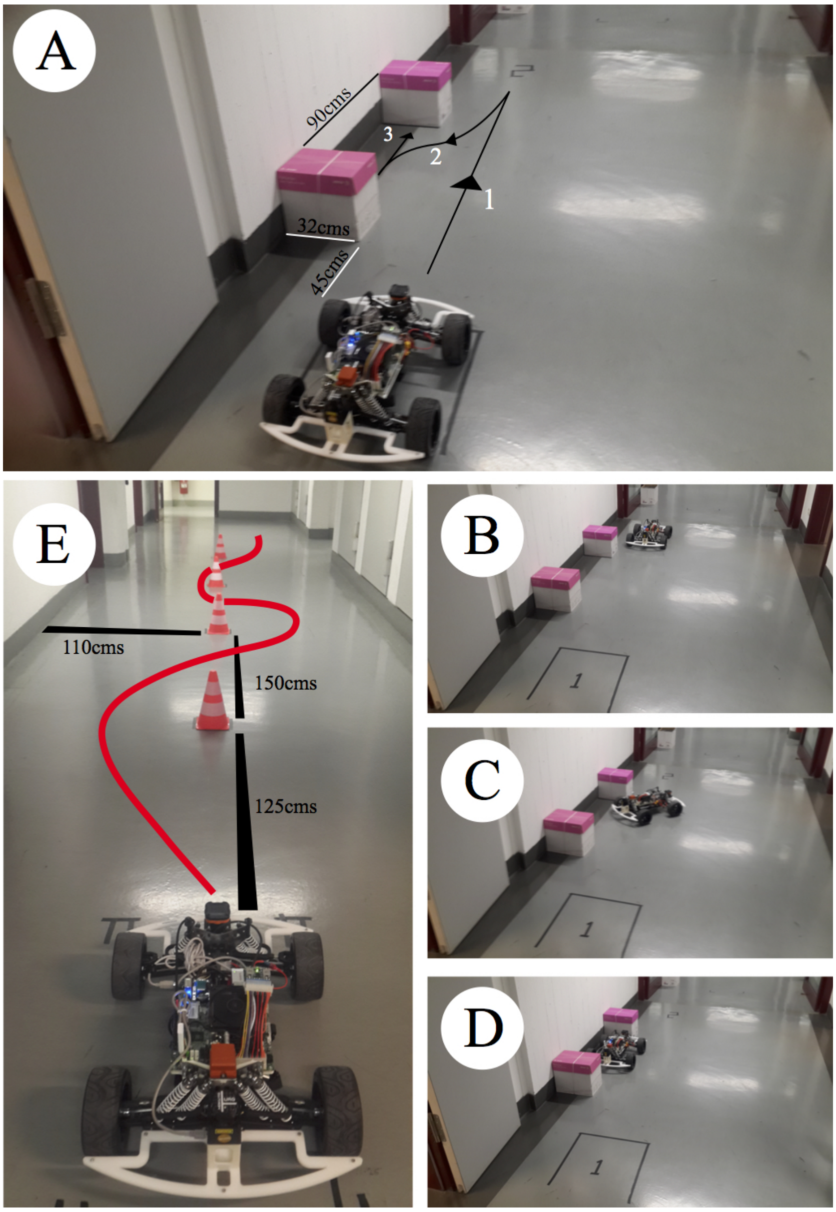
## 2.1 The Parking Task

The Parking task is illustrated in Figure 2.2 (A-D). The dimensions of the obstacles are also portrayed. The car starts from position 1 (shown clearly in 2.2 B) and move forward to position 2 with a trajectory 1 as shown in figure 2.2 A. It should stop when a parking place is detected. The first task then is to write a algorithm to detect parking space. Once this parking place is detected the car should perform a reverse parking manoeuvre that takes into consideration where the car position relative to the parking place is. A second task then is to calculate this relative position. The third task is to manipulate a "reverse parking" primitive trajectory given the input of the relative "stop" position (around position 2). Then the third task becomes to both identify and generalise a reverse parking trajectory such that a correct parallel parking is achieved. A final primitive is a correction trajectory to "fix" the final position. Summarising (suggested) the tasks are then:

1. Move forward from position 1 to position 2.
2. Localise the car at position 2 (stop position).
3. Use the stop position as an input to manipulate a "primitive" reverse parking position.
4. Perform the parallel parking manoeuvre while reacting to laser scan information.
5. Stop before hitting the obstacle.
6. Perform a corrective motion (front and to the right to finalise the car position)

## 2.2 The Slalom Task

The slalom task is shown in Figure 2.2 E. The dimensions are also shown. Just like the Parking task, the idea is to react to environmental information and adjust "primitive" trajectories accordingly. There would be two main primitives: turn right, and turn left. Alternatively, this task could be performed by setting different goal points and focusing on the smooth transition between them using local planner.


A
90cms
3
2
1
32cms
45cms
E
110cms
150cms
125cms
B
1
C
1
D
1