Class Information

ECE 683 in 2007

ECE683 will be run in 2007 as a mini-version of the DARPA 2007 Urban Challenge. Students are encouraged to skim the documents related to Rules in the DARPA web site: http://www.darpa.mil/grandchallenge/index.asp

However, we shall have much more simplified rules, and instead of cars and a city street network, we shall have robots and an indoor network defined within the lab DL731.

The DARPA 2007 Urban Challenge

The Urban Challenge is meant for autonomous cars making their way through a small city, abiding by traffic rules. The cars have GPS, to find their absolute position and also to compare to "waypoints" given in a "map". They also have sensors to sense the lanes and obstacles, including other cars on the roadway. Although the map is known beforehand, the actual "mission" in terms of a small set of waypoints to be visited in sequence, is provided just before the race. These special waypoints are called "check-points", and the vehicle is to plan a path and visit each checkpoint and return. Along the way, it may go through intersections (with stop signs, there are no lights), do left and right turns, park, do lane-changes, etc.

The ECE 683 Mini-Challenge

ECE 783 in 2007 emulates some of the issues to be addressed in the Urban Challenge. There are some obvious differences:

- Smaller wheeled robots in the lab are used instead of cars on streets.
- GPS is emulated by a camera system in the lab. A camera on the ceiling locates each robot (aided by a encoded sign on each robot) and transmits location, speed, and heading data by wireless, similar to GPS.
- There will be a map and a single separate mission files. There will be a single mission file including both way-points and check points. So there is no "path planning" in this race.
- A generic intersection is drawn on the lab floor. Portions of this will be used at times, specifically when an intersection is called for. However, this is mostly for visual help for those watching. The prime definition of lanes is in the files indicating waypoints.
- The robots have a single Lidar, a pair of cameras and a set of ultrasonic sensors.
- Teams can practice by
    - Testing individual sensors
    - Testing low-level and high-level control first on the simulator
    - Testing segments of lanes and intersection layouts using the robots

Figure 1: The generic intersection.

Possible practice layouts

A number of turn scenarios can be set up using the generic intersection layout in the lab shown in Figure 1. The ceiling camera can provide the pseudo-GPS readings within the black area. Two regular lane layouts are suggested within that same area.

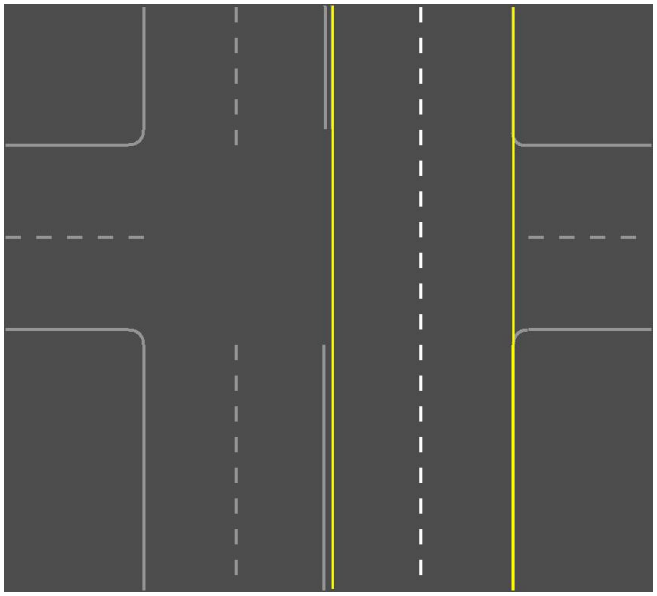


Figure 2. Two-lane straight street.

Figure 2 shows a straight street with two lanes which can be used for same direction or opposite direction driving. If the two lanes are in the same direction, they can be used for obstacle avoidance and single and possibly double lane change. If they are for opposing

direction traffic, it can be set up to test whether obstacles in the adjacent lane bother the robot.

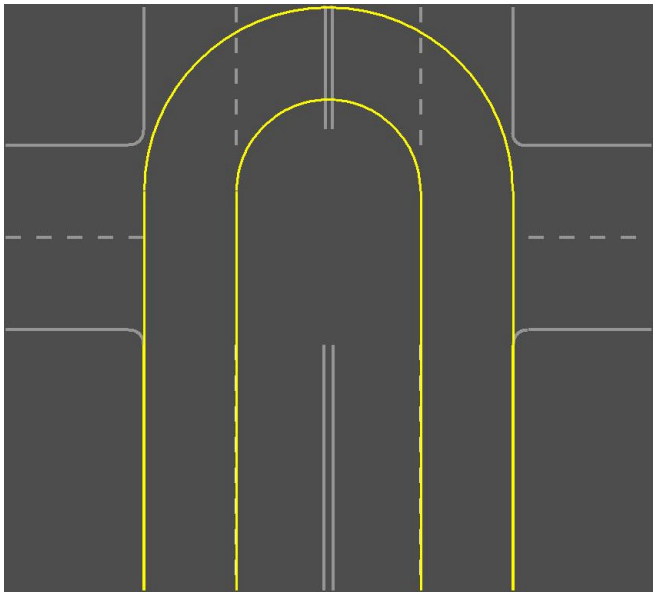


Figure 3. The curved road.

The layout in Figure 3 can be used for a curved road indicating a lane with a 180 turn. It can be driven in either direction.

If teams decide to use lane markers utilizing cameras, some additional lanes may be visually marked.

Network Definition File for Class

A link is one or more parallel lanes traveling in the same direction.

A lane is defined by at least 2 waypoints (the start and the end) and any number of additional intermediate points. Waypoints are in the center of the lane.

In the definition file, lanes are ordered from left to right (when facing the direction of travel). You can only switch from one link to another at the end of a lane (at the last waypoint), and lanes are always entered at the first waypoint.

You can switch among lanes belonging to the same link at will.

When traveling on a link consisting of multiple lanes, you must be in the righthand lane when making a right turn, and the lefthand lane when making a left turn.

Nodes represent intersections. SSFlag= 1 if there is a stopsign at the end of that link, and 0 if not.

File Format: (file is text with line breaks)

```
Number_of_Links (integer)
LinkID (integer)
   Number_of_lanes (integer)
   Number_WP_lane_1  wp_x1 wp_y1 wp_x2 wp_y2 ...
    ...
   Number_WP_lane_n  wp_x1 wp_y1 wp_x2 wp_y2 ...
Number_of_Nodes (integer)
NodeID (integer)
  Node1_Number_Entering_Links EnteringLinkID1 SSFlag EnteringLinkID2
      SSFlag ...
  Node1_Number_Exiting_Links  ExitingLinkID1 ExitingLinkID2 ...
   ...
```

Mission Definition File

You should travel from the first to the last link listed, traversing each link in order. Stop at the end of the last link. When transitioning from one link to another, obey normal driving rules:

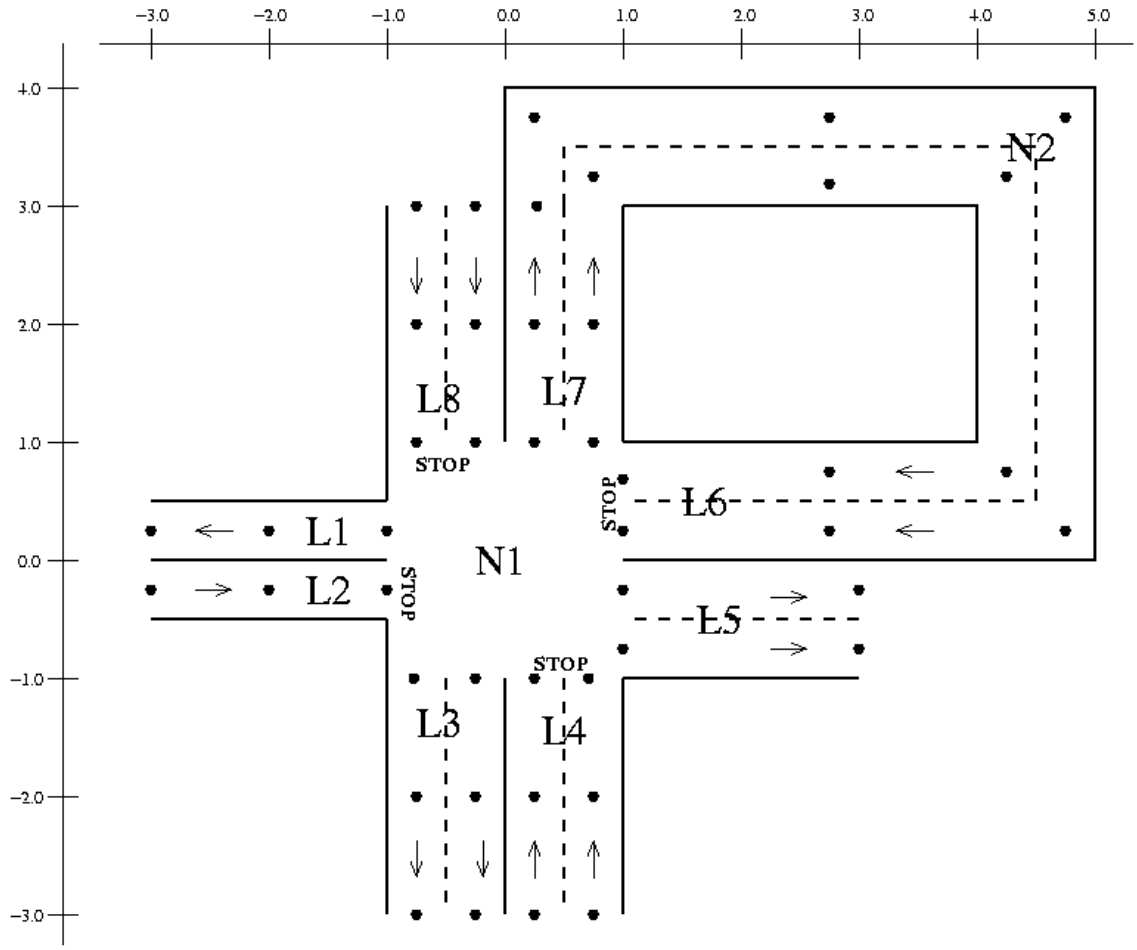
- stop at stop signs (do not stop without a stop sign)
- check for precedence (4 way, 3 way, 2 way intersections)
- proceed when safe

When transitioning from one link to another, smoothly navigate from the last waypoint of the current link to the first waypoint of the next link. Those waypoint may or may not be coincident.

File Format:

```
Number_Of_Links_To_Traverse
LinkID1 LinkID2 ...
```

Example of a Network Definition File




```
# sample network file
#
8     # 8 links in network
1          # link 1
1          #    has 1 lane
3 -1.0 0.5 -2.0 0.5 -3.0 0.5
2          # link 2
1          #    has 1 lane
3 -3.0 -0.5 -2.0 -0.5 -1.0 -0.5
3          # link 3
2          #    has 2 lanes
3  -0.25 -1.0 -0.25 -2.0 -0.25 -3.0
3  -0.75 -1.0 -0.75 -2.0 -0.75 -3.0
4          # link 4
2          #    has 2 lanes
3  0.25 -3.0 0.25 -2.0 0.25 -1.0
3  0.75 -3.0 0.75 -2.0 0.75 -1.0
5          # link 5
2          #    has 2 lanes
3  1.0 -0.25 2.0 -0.25 3.0 -0.25
```

```
3  1.0 -0.75 2.0 -0.75 3.0 -0.75
6          # link 6
2          #    has 2 lanes
4  4.75 3.75 4.75 0.25 2.75 0.25 1.0 0.25
4  4.25 3.25 4.25 0.75 2.75 0.75 1.0 0.75
7          # link 7
2          #    has 2 lanes
6  0.25 1.0 0.25 2.0 0.25 3.0 0.25 3.75 2.75 3.75 4.75 3.75
5  0.75 1.0 0.75 2.0 0.75 3.25 2.75 3.25 4.25 3.25
8          # link 8
2          #    has 2 lanes
3  -0.25 3.0 -0.25 2.0 -0.25 1.0
3  -0.75 3.0 -0.75 2.0 -0.75 1.0
2      # 2 nodes in the network
1          # node 1
4 2 1 4 1 6 1 8 1   # 4 entering links, all with stop signs
4 1 3 5 7           # 4 exiting links
2          # node 2
1 7 0               # 1 entering link, no stop sign
1 6                 # 1 exiting link
```

In this file, # indicate a comment, and everything on the line past the # sign is to be ignored.