Department of Electrical and Computer Engineering
Course ECSE 211 – Design Principles and Methods
Fall 2019 Project Description
Version 2.1, November 20, 2019

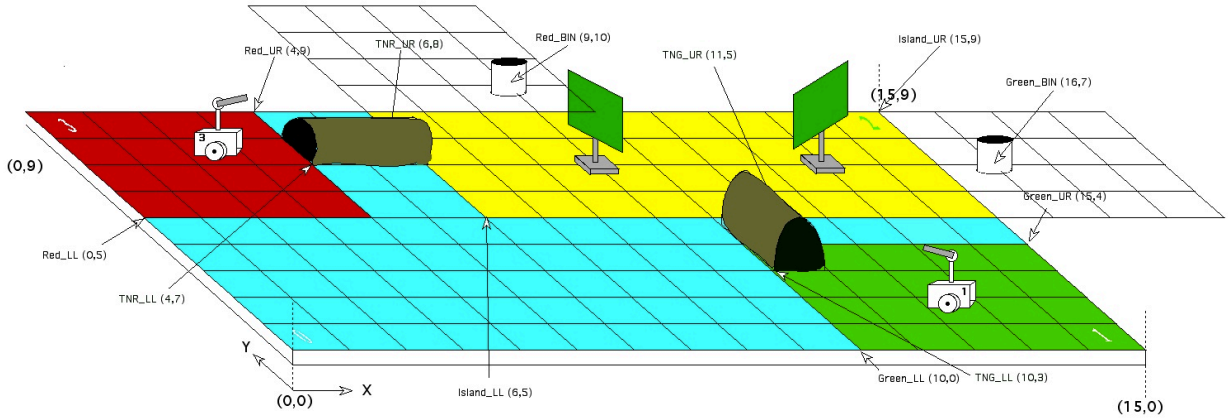


Figure 1

Revisions

This document is largely for clarification based on team meetings – see the Notes section at the end of the document. There is one change, however, that will require a minor change to your code, the addition of separate BIN parameters for each team – see below for details.

Overview

The goal of this project is to design and construct a machine that can autonomously navigate a closed course and launch a ping-pong ball into a bin residing at a known location outside the perimeter as shown in Figure 1. Consider the scenario depicted in Figure 1, with two players labeled 1 and 3. The labels indicate the corners each machine started in, so Player 1 starts in Corner 1 (Green Zone) and Player 3 in Corner 3 (Red Zone). Each of the zones is surrounded by a virtual river (blue regions), connected to a central island (Yellow Zone) by tunnels. Each zone corresponds to a rectangular region defined by its lower left (LL) and upper right (UR) corners relative to the origin. In the example shown in Figure 1, the red zone is defined as Red_LL (0,5) to Red_UR (4,9), and the green zone is defined as Green_LL (10,0) to Green_UR (15,4). Information is transmitted to each machine/player using a provided WiFi class. In the Wifi dialog, these coordinates are passed as individual components, hence Red_LL (x,y) would be sent as Red_LL_x and Red_LL_y.

The playing field measures 15' x 9', with the origin located in the lower left hand corner, (0,0), as shown in Figure 1. At the start of a round, both players are placed in their respective corners at a random orientation and started. Each player waits for a set of game parameters to be downloaded from the game server (more about this later). Once the parameters are received (which describe the layout of the playing field), each player must cross the river over to the island. The key parameters here are RedTeam, GreenTeam, RedCorner and GreenCorner. Each

player has an assigned team number, so it can determine whether it is in the red corner or green corner by matching against RedTeam or GreenTeam. Once the team color is identified, the starting corner can be located by the RedCorner and GreenCorner parameters respectively. From here two key landmarks become available: the location of the tunnel connecting the starting zone to the island, and the location of the bin into which the ball is to be launched. In the example shown in Figure 1, the red player would cross using the tunnel located at TNR_LL (4,7) to TNR_UR (6,8) and navigate to a launch point that can reach the bin located at coordinates redBin (9,10). Since the bin is defined as located at the intersection of two grid lines, it is specified as a single coordinate pair. Similarly, the green player would cross using the tunnel located at TNG_LL (10,3) to TNG_UR (11,5), and navigate to an appropriate launch point to reach greenBin (16,7). There are two complications that each player must cope with. First, there are obstacles with screens located at random positions on the island. Second, both players **must** avoid hitting each other. The first player to contact the second is automatically disqualified.

Pay close attention to how the tunnels are positioned relative to the red and green zones. Notice, in this example, that the tunnel connecting the red zone joins at the boundary whereas the tunnel connecting the green zone overlaps by one square. This will always be the case when the border separating two zones is one square wide.

Once each player reaches the island, it must navigate its way around any obstacles to reach an attainable launch position. The ballistic launcher you designed in Lab 5 should be able to launch a ball with a repeatable trajectory. Knowing this you can backproject the location of the bin to an arc corresponding to viable launch positions for reaching the bin. There are a number of design challenges implicit in this task. Given the dimensions of the tunnel, there is a limit to how large each machine can be which subsequently limits the size/complexity of your launch mechanism. Each team will be given a set of ping-pong balls which **must** be the ones used in the competition. You are only required to sink one ball over the course of 4 rounds. However, your odds increase the more balls you can carry (resulting in a larger vehicle, loader mechanism, etc.). Since there is a time limit of **5 minutes**, machines must be nimble enough to move with a reasonable speed.

Specific Details:

The WiFi class delivers the game parameters which are summarized in the following section. The procedure that each player must follow is summarized in the following steps and *must* be adhered to:

1. Each robot is placed in the corner specified by the marshal running the competition round. You will be instructed as to where to place and orient your machine.
2. Once placed and the start button pushed, you are no longer permitted to touch your machine (*see details in the Notes section below*). If there is any contact with the machine the team is disqualified for that round.
3. One started, the machine waits for the game server to deliver the parameters for the current run. This is done through a method call which will block until complete.
4. Each machine localizes to the grid. When completed, the machine must stop and issue a sequence of **3 beeps**.

5. Each machine navigates to their corresponding tunnel, transits, and then proceeds to their launch point. Upon arriving, each machine will again stop and issue a sequence of **3 beeps**.
6. Each machine fires ping-pong balls into the bin until depleated.
7. Each robot returns to its starting corner.
8. Upon returning to the starting corner, each robot halts and issues a sequence of **5 beeps**.

Each team will have an opportunity to participate in 4 runs. A design will be deemed "successful" if it succeeds in delivering at least one ping-pong ball over the series of runs. The "competition" aspect relates to the number of points accumulated by each team for completing the different steps outlined above. On completion, each team is ranked in terms of the total number of points acquired. This is for bragging purposes only – your course grade is based on the quality of your design and not just the number of points accumulated.

Parameters

Game play is determined by a set of parameters, which are sent to the client (player) from a server. The following parameters are defined according to the details provided in Figure 1:

RedTeam (i=1,23) – Team starting out from red zone
GreenTeam (i=1,23) – Team starting out from green zone
RedCorner (i=0,3) – Starting corner for red team
GreenCorner (i=0,3) – Starting corner for green team
Red_LL (x,y) – lower left hand corner of Red Zone
Red_UR (x,y) – upper right hand corner of Red Zone
Green_LL (x,y) – lower left hand corner of Green Zone
Green_UR (x,y) – upper right hand corner of Green Zone
Island_LL (x,y) – lower left hand corner of the Island
Island_UR (x,y) – upper right hand corner of the Island
TNR_LL (x,y) – lower left hand corner of the red tunnel footprint
TNR_UR (x,y) – upper right hand corner of the red tunnel footprint
TNG_LL (x,y) – lower left hand corner of the green tunnel footprint
TNG_UR (x,y) – upper right hand corner of the green tunnel footprint
redBIN (x,y) – location of the target bin for the Red team.
greenBIN(x,y) – location of the target for the Green team.

Note that the (x,y) coordinates listed correspond to the grid coordinates shown in the Figure 1. In the WiFi class, point parameters (x,y) are sent individually, e.g., TR (x,y) would be sent as TR_x and TR_y

Parameter Ranges

Red_UR_x – Red_LL_x:	Min=2, Max=10
Red_UR_y – Red_LL_y:	Min=2, Max=10
Green_UR_x – Green_LL_x:	Min=2, Max=10
Green_UR_y – Green_LL_y:	Min=2, Max=10
Island_UR_x – Island_LL_x:	Min=2, Max=10
Island_UR_y – Island_LL_y:	Min=2, Max=10

TNR_UR_x – TNR_LL_x:	Min=1, Max=2
TNR_UR_y – TNR_LL_y:	Min=1, Max=2
TNG_UR_x – TNG_LL_x:	Min=1, Max=2
TNG_UR_y – TNG_LL_y:	Min=1, Max=2

Red_BIN_x:	Min=-5, Max=20
Red_BIN_y:	Min=-5, Max=14
Green_BIN_x:	Min=-5, Max=20
Green_BIN_y:	Min=-5, Max=14

Game Play

Both players act almost independently, so the design can focus mainly on navigation, mobility, handling and launching of ping-pong balls. Some collision avoidance will be necessary to cope with obstacles and the event that both players are in the same vicinity. Each team will participate in 4 rounds for which a cumulative score will be determined. The score is based on points awarded for exhibiting each of the behaviors required to play the game: localization, navigation, traversing the river, avoiding obstacles, reaching launch position, throwing balls, and returning to the starting corner. These points effectively validate the components of your design. On top of this we also record how long it takes for you to complete the entire process. These figure prominently in ranking the performance of the teams with respect to the "competition".

Materials

Each team has up to 3 Lego Mindstorms kits worth of parts available. In addition, a MakerBot Replicator 2 rapid prototyping machine is available for fabricating parts for those inclined. You may also purchase additional materials, but these must receive prior approval from the instructors. Another note – all computation must be done on board the EV3 brick(s); no offloading to an external machine is permitted.

Final Word

This document is the final revision and incorporates feedback received up to Beta. No further changes will be issued unless they are deemed to be critical.

You are being evaluated on your robot's ability to complete the various tasks that make up the game, so it is important to show what your machine can do. Unless your machine does nothing at all, you will be awarded points for what it can do.

Notes:

1. Each robot must display their team number on their robot. An adhesive sticker, paper sign, etc. will do as long as the number is clear and visible from across the competition floor.
2. Your WiFi ID is your team number as in the Beta demo.
3. You may only issue beeps as specified in this document. Violation will result in immediate disqualification.
4. In the Beta demo some teams were confused about the start procedure. To avoid any miscommunication or delays in starting, the following procedure MUST be adhered to:
    a. Software must be **pre-loaded** before entering the competition area.
    b. Absolutely no use of the remote console is permitted to start your robot.
    c. You will place your robot in the indicated starting corner as instructed by the marshals.
    d. You will then start your software with a button press. At this point you are free to have the robot programmed to do any set up required (starting threads, initializing sensors, etc.) – but the robot must not move until parameters are received. If the robot moves before this, or requires any further user intervention to start – the robot is disqualified.
5. There was also some confusion about placement of the robot in the start corner. The rule is that the center of rotation (the black dot in Figure 2 below) must lie in the starting corner. For the placement shown in the figure, the robot can rotate in a fairly large circle (diameter = 1.4 x square side).
6. Each team will be required to produce a checklist itemizing all the critical steps that must be performed prior to placing the robot in the starting corner. The checklist must be run prior to each trial, the document signed by the team, and presented to the competition desk before a team will be allowed to run.
7. Your robot must be placed and ready to go at your assigned competition times. If you are not ready to go at the appointed time, the team forfeits that round.
8. Please note that Figure 1 is only a representation of a possible layout. Layouts will change on each round and possibly within each round.

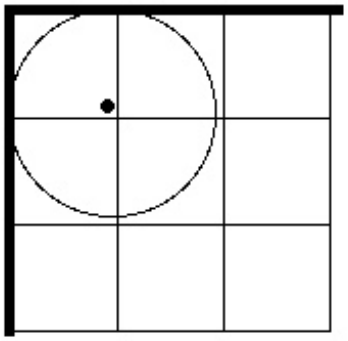


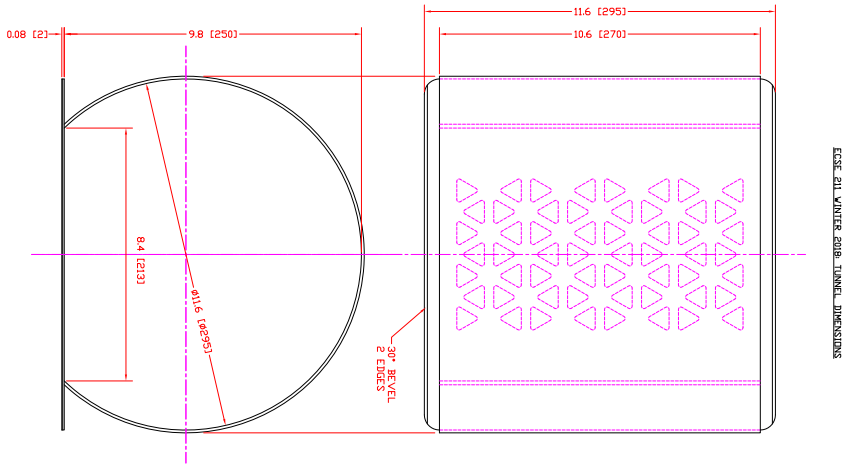
Figure 2

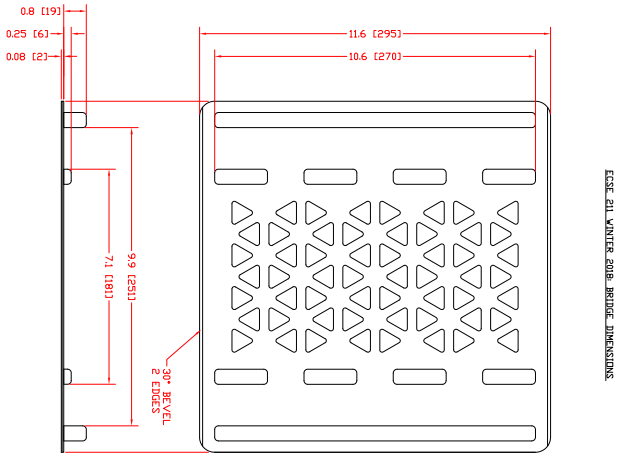
Good luck to all teams!

Version 2.1, November 20, 2019
am/dal/fpf

Appendices

Bridge and tunnel drawings, length = 1 tile
For the Fall 2019 competition we will only be using the tunnel component





Photos: Tunnel (left) and Bridge (right)
(speed bumps on bridge not shown; floor of tunnel is smooth, no grating pattern).

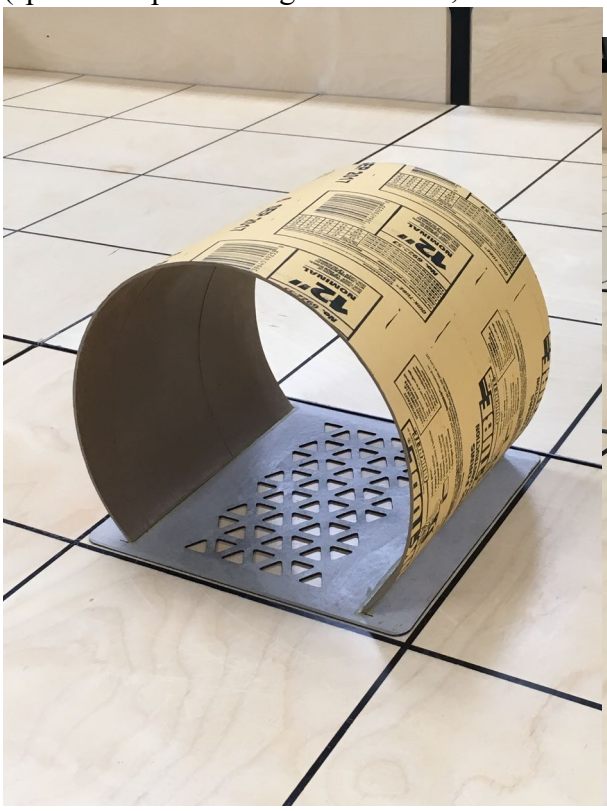


Photo: Bridge with speed bumps