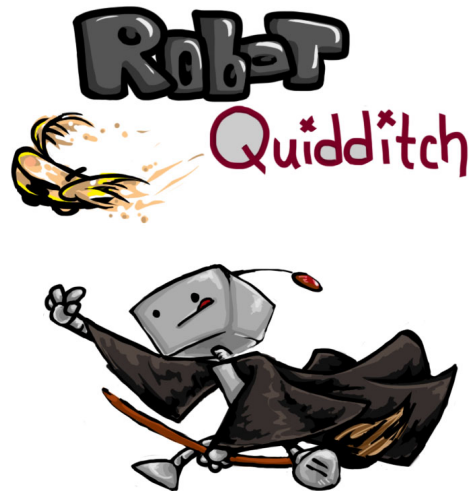


Final Design Challenge: Robot Quidditch

Due: December 4th, 2014



Introduction

The goal of this design challenge is for your team to program your robots to win the ENGI 128 version of Quidditch. On the last class, we will have a competition to determine the victor of the ENGI 128 Quidditch World Cup.

Being muggles and all, some of you might not be familiar with the noble wizarding game of Quidditch. This is rather sad, but not unexpected. You have three options available to correct this deficiency:

1. Good: Read the Wikipedia Quidditch page: <http://en.wikipedia.org/wiki/Quidditch>
2. Better: Rent the "Harry Potter" movies. Years 1-4 all feature good Quidditch matches.
3. Best: Read the "Harry Potter" books. They're really good.

1 Design Challenge Goals

The goal of this design challenge is to develop a complex program using the skills we've covered in this course. In particular, you will need to communicate with other robots, write control loops to govern the behavior of the robots, and design clever strategy to outwit your opponents.

The goal is for each team to score as many points as possible within a limited time. There are two ways to score: 1. Get your *Quaffle robots* to the opposing team's endzone, or 2. capture the *Snitch robot* with your team's *Seeker robot*. All the while, *Bludger robots* from the opposing team will be harassing the movements of your robots.

You will write your software as a team, but each student will also write an individual report on their contribution. Keep notes about your contributions to aid in writing this report.

2 Rules

1. Each match consists of teams of autonomous robots. Each team can field 4 robots with any combination of Quaffles and Bludgers, and up to one Seeker.
2. Each team must decorate their robots with *removable* team livery that is unique to each team, and clearly identifies the robot as a Quaffle, Bludger, or Seeker. See Section 2.5 for details.

3. Each match lasts 5 minutes.
4. Each robot must broadcast its team number and robot type in the first two characters of the neighbor system message: "nT" where 'n' is the team number: {1-4}, and 'T' is robot type: {Q, S, B, N, E} (Quaffle, Seeker, Bludger, sNitch, Endzone). The remaining data in the neighbor message can be determined by the team. Students will call a library function in the 'quidditch.py' library module to set the message.
5. Each team must start with their robots inside the start circle in any orientation.
6. The Snitch robot starts first, then the two teams start a few seconds later. The start command will be given via the radio system with a **radio start message**: the string 'start' over the radio. If a robot starts before the sequence, the robot will be disqualified.
7. Once started, team members can have no further contact with their robots.
8. You must use the provided libraries for basic robot control. Do not edit these library files. If you need a function, or cannot do something, ask the staff; we will help you solve your problem, or add the functionality to the libraries, and make the changes available to all teams. *It will be considered an honor code violation to edit the libraries*, because that would allow you to cheat in various ways.
9. You can use your LEDs to indicate what your robots are doing. In particular, we will provide functions to produce three distinct patterns, "Quaffle scoring attempt", "Quaffle score", and "Seeker tracking Snitch". Refer to Section ?? for more detail on the LED patterns.
10. The maximum speed of any player robot will be 100 mm/s. This will be enforced by the library functions.
11. The Snitch robot is exempt from speed limits or IR communication requirements.

2.1 Quaffle Robots

A Quaffle robot can score **10 points** points by traveling from your end zone to the opposing team's end zone, without touching anything. Sounds easy, but watch out for the Bludgers!

1. To score, the Quaffle robot must get within range of the IR transmitters in your team's end zone. These transmitters will have a limited range of only around 20cm. Next, the Quaffle robot needs to indicate that it is about to make a scoring attempt by displaying the **Quaffle score attempt** pattern on its LEDs. Then the robot needs to travel across the Quidditch pitch until it is within range of the IR transmitters in the opposing team's end zone, *without hitting anything at any time during the scoring run*.
2. When the Quaffle is on a scoring run, the robot must display the **Quaffle score attempt** light pattern. We will provide a function that generates this pattern. Contact with anything by the robot's bump sensor must disable the **Quaffle score attempt** light pattern. If the Quaffle robot gets within range of the opposing team's end zone IR transmitters, it must display the **Quaffle score** light pattern.
3. For this robot to score again, it must return to your team's end zone, receive the IR message from the end zone, and repeat the process. It is allowed to run into obstacles on the return trip.
4. Bear in mind that a good Quaffle program will make as many round trips as possible, but the robot speed limit and the size of the course, limit the number of possible runs. Think about this when planning strategy and selecting your makeup of robots.

2.2 Bludger Robots

The job of the Bludger robots is to annoy and harass robots on the opposing team.

1. Bludgers can collide with other robots.
2. Bludgers cannot purposely harass the Snitch.

2.3 Seeker Robots

The job of the Seeker robots is to look for the Snitch robot.

1. There can only be one Seeker robot on a team.
2. When the Seeker robot is moving towards the snitch, it must display the **Seeker follow attempt** light pattern.
3. The Snitch robot will determine which Seeker is following it, and display the team color, red or blue, of the trailing Seeker.
4. If a Seeker robot can stay within IR communications range of a snitch for 10 seconds it will score **30 points** for the team.

2.4 The Snitch Robot

The Snitch is programmed by the course staff. It's sneaky.

1. If the Snitch robot has a Seeker robot within capture range for 10 seconds, it will indicate that it was caught, and rotate to face the Seeker that caught it. If two Seeker robots catch the Snitch at the same time, the Snitch will be responsible for breaking ties between two Seekers. The decision of the Snitch is final.
2. Once caught, the Snitch will then go into "stealth" mode for 30 seconds by disabling its IR transmitters and moving to another part of the Quidditch pitch. Good luck finding it again!
3. The Snitch will try very hard to not get caught. It will be faster than your robots, it might stop transmitting IR messages, or move in very abrupt patterns.
4. The Snitch will start to get tired after a few minutes. That is the best time to catch it.

2.5 Team Outfits

1. Each team must decorate their robots with *removable* team outfits. These outfits must not interfere with the operation of the bump skirts, or extend beyond the footprint of the robot.
2. Each team's theme must be unique and easily identifiable.
3. The outfits must be different for the Quaffles, Bludgers, and Seeker, and easy to tell apart.
4. These outfits must fit within the robot's footprint, and extend no taller than 1.5" above the antenna.
5. Decorations should only be placed on the top board of the robots.
6. They should keep the LEDS visible
7. They can not block the IR communications system, the light sensors, the buttons, or any other key robot system.

3 Report

Each student will be write a *one-page* report describing your team's design to solve the design challenge, and your contribution to the project. This is an individual report, describing a group project, so keep notes about the problems you solved and the software you developed. Be mindful of the honor code; if you and your teammate both claim to have developed the same part, that is bad for everybody. Feel free to include psuedo-code examples, plots of data collected during testing, or pictures of your robots in action. Links to videos are ok too, but we may not have time to watch them, so be sure to describe the video in the text of the report. Keep your font at size 11. **DO NOT EXCEED THE PAGE LIMIT.** We will stop reading reports after the first page.

4 Group Development Suggestions

Working in a group means you need to share source code, and sharing source code can be tricky. You can try using email or Google docs, but the best way to do this is to use a program called *Subversion*. This is an advanced tool, and outside the scope of the course. If you are gung-ho about setting it up and feel you can serve as the technical support for your group, we can provide limited guidance for getting setup, but that is it. Use this tool only if you are comfortable learning about it on your own time.

5 Bonus Sound Points

We'll be awarding a special prize and bonus points to the team that best uses sound in their design.

DRAFT