ORTOP Workshop III
Robot Navigation & Mission Planning

CITY SHAPER
INSTRUCTORS

Terry Alexander
taalexand@gmail.com
503-970-9243

Don Nelson
don-nelson@ortop.org
503-550-8028

Roger Swanson
swanson@hevanet.com
503-297-1824

Ken Cone
kencone@gmail.com
503-415-1465

Jim Ryan
jrryan57@gmail.com
503-803-0388

Loridee Wetzel
loridee-Wetzel@ortop.org
503-486-7622 ex. 102
ORTOP Program Manager
ORTOP W3 GOALS

- **Goal:**
  - Help Teams Learn Robot Navigation & Mission Planning Skills

- **Agenda:**
  - Common Navigation Problems
  - Waypoints & Error Zones
  - Using Rotation & Color Sensors
  - Navigation: Best Practices
  - Mission Planning: Identify waypoints to Food Growth Chamber
  - Color Sensor Review
  - Turning Methods
  - Line Followers
  - Missions: Space Travel Ramp M01 & Food Growth Chamber M10
  - Any questions from W1 or W2?
  - Slides at: ortop.org/Workshops
GOING STRAIGHT

- Common Navigation Problems
  - Test your intuition: What navigation problems do you observe?
GOING STRAIGHT

Common Navigation Problems
- Video from a real team, Search Engine mission
- Items to look for:
  - Starting position in base
  - Moves & turns
  - Robot design - light sensor & tool motor
  - What happens when the robot approaches the Search Engine mission?
- Return to base
- Repeatability
- Lets rerun the video and look for these items
- Questions/Comments?
Going Straight Using Rotation Sensor

- Tape 2 pieces of paper on table, about 2 feet apart
- Program the robot to move 2 feet, starting on the first piece of paper and stopping on the second
- Start the robot at exactly the same location for each run. Draw a line on the first piece of paper and place the front axles on the line, marking their position. Also, mark the rear ball starting position.
- Using the front axle marker, place a dot or small x where the robot stops on the second piece of paper
- Run 4-5 times at speeds of 20 & 60 recording data for each run
- Draw a box around the stopping points to show X and Y position, this box is the error zone
- Go! 10 min.
Going Straight

Going Straight Results

- What was your **rotation sensor error zone** for the going straight lab?
  - speed 5: X side to side ___ , Y front to back ___
  - speed 20: X side to side ___ , Y front to back ___
  - speed 60: X side to side ___ , Y front to back ___
- Note that the robot tends to **wobble** from side to side. This is due to motor sensor feedback. Slow one motor and the other motor slows down. The robot **tries** to go straight.
- Class discussion of results
Going Straight Using Color Sensor

- Place a strip of blue tape on the second piece of paper
- Program the robot to move forward (speed 20) to detect the blue line and stop
- Use “Color Sensor - Compare - Color”
- Mark where the robot stops using one of the front axle positions
- Repeat the test with speeds 5 & 60.
- Go! 10 min.
GOING STRAIGHT

- **Color Sensor Results**
  - What was your **color sensor error zone** for the going straight lab?
  - speed 5: X side to side ___ , Y front to back ___
  - speed 20: X side to side ___ , Y front to back ____
  - speed 60: X side to side ___ , Y front to back ____
  - Class Discussion of results. Notice any overshoot in these exercises?
WAYPOINTS

- **Waypoints & Error Zones**
  - Introducing **waypoints** - points along the path from starting position to end goal
  - Each **waypoint** has an **error zone**
  - Search Engine Mission starts at Base and ends in front of the Search Engine, ready to pick a knowledge block
  - We will address waypoints more in a few minutes.
Error Zone Discussion:
- How can we relate error zones and going straight back to the Search Engine mission?
- Going straight & error zones are key navigational skills
- Final thoughts …
Oregon FLL Championships - Jan. 2015
- Distribution of high & low table scores
- Low table scores below 100 ~ 15 teams each day
- High table scores above 200 ~ 10 teams each day
- What are the navigation skill differences?
ORTOP watched lots of competition videos, here is what we observed:

- **PROBLEMS**
  - **Aiming** the robot in base
  - **Moves & Turns** without finding waypoints
  - **Heavy** attachments - tool motor has little control with a massive attachment
  - **Single** finger attachments
  - **Team members** not knowing what to do next

- **BEST PRACTICES**
  - Fixtures to align robot in base
  - **Plan** each mission’s waypoints
  - **Gears** w/small motor, or use large motor
  - **Fingers** to cover error zones
  - **Practice** mission changes, watch YouTube

- **So let’s get started planning a mission!**
NAVGATION SKILLS

- Alignment blocks in base
- Compensation for Error Zones
- Carabiner Example
- Bump the wall or corner
- Follow the wall wheels

Base Alignment  Error Zones  Follow the Wall
Establishing Waypoints

- Plan a route to move the robot from base to the black line in front of the Space Travel Ramp M01, then move forward to launch the Vehicle Payload (easy)
- Plan a route to move the robot to a place in front of the Food Growth Chamber M10 (more difficult)
- Include waypoints, moves, turns, line follows, wall follows in your plan
- Use Mission Planning & Into Orbit table overview sheets
- Go! 10 min. Class Discussion of Results
Establishing Waypoints

- Plan routes to M01 or M10
Mission Planning Sheet

Mission(s): ___________________________________  Maximum Score: _________
Program Name: ________________________ Memory size: ____  Time to run: _____________

(image of field mat)

Plan Overview: ________________________________________
________________________________________________________________________
________________________________________________________________________
________________________________________________________________________
________________________________________________________________________
________________________________________________________________________
Starting Position: ________________________________________
________________________________________________________________________

Sensors Used (Port): ________ (___), ________ (___), ________ (___)
________________________________________________________________________
________________________________________________________________________
Navigation Details (include waypoints): ________________________________
________________________________________________________________________
________________________________________________________________________
COLOR SENSOR

- **Reflected Light vs. Color**
  - Color sensor does not replace reflected light sensor - depends on situation
  - Don’t trust your eyes for color - use Port View
  - Some times: Sample - move forward - sample again

- **LAB:**
  - Open program: ColorGradient50.ev
  - Start at white end of gradients G, B, & R
  - Run the program, mark (in front of the light sensor) where the robot stops

- **What’s happening?**
  - Reflected Light …
  - Red gradient…
  - Change port sensor to Color
  - Record gradient color values
  - Record color bloc

- **Color values:**
  - 0 = No Color
  - 1 = Black
  - 2 = Blue
  - 3 = Green
  - 4 = Yellow
  - 5 = Red
  - 6 = White
  - 7 = Brown
2 wheel “spin” turn:
- on for degrees
- steering slider right or left
- medium power
- wheel rotation 180 degrees - turns robot ~ 90 degrees (depends on wheel size) starting point>>>trial error
- brake when finished
1 wheel turn:
- on for degrees
- medium power
- wheel rotation 360 degrees to turn the robot ~90 degrees trial error>>>
- brake when finished
- LAB exercise at home: record error zones & overshoot for both kinds of turns
LAB: Program a line follower
Follow the LEFT edge of the black line for 3 seconds
Write in English what you want the robot to do:
  e.g. Turn ? and look for a black line, what happens next?
  continue in a loop, wait for 3 second timer
PARALLEL FOLLOWER

- Programming commands you might need:
  Loop, turn right/left, light sensor control block, 3 second timer, break out of the loop.
- Run two programs in parallel
- Follow a line
- Look for sensor input to break out of a loop
Alternate programming commands you might need:
Loop, Turn right/left, light sensor control of a switch
Navigation skill areas in practice
- Stafford GEARS
PROGRAMMING HELP

- **Help** tab at top
- **Show Context Help** - highlight a program block, then click Context Help

- **Show EV3 Help** - takes you to top level EV3 help - help files are on your computer

---

```
<table>
<thead>
<tr>
<th>Block</th>
<th>Mode</th>
<th>Use</th>
</tr>
</thead>
<tbody>
<tr>
<td>Wait</td>
<td>Gyro Sensor - Compare</td>
<td>Wait for the rotation angle or rate to reach a certain value.</td>
</tr>
<tr>
<td>Wait</td>
<td>Gyro Sensor - Change</td>
<td>Wait for the rotation angle or rate to change by a certain amount.</td>
</tr>
<tr>
<td>Loop</td>
<td>Gyro Sensor</td>
<td>Repeat a sequence of blocks until the rotation angle or rate reaches a certain value.</td>
</tr>
<tr>
<td>Switch</td>
<td>Gyro Sensor</td>
<td>Choose between two sequences of blocks based on the rotation angle or rate.</td>
</tr>
<tr>
<td>Sensor</td>
<td>Measure</td>
<td>Measure the rotation angle and/or rate, and get the result on a Numeric data wire.</td>
</tr>
<tr>
<td>Sensor</td>
<td>Compare</td>
<td>Compare the rotation angle or rate to a threshold, and get the result on a Logic data wire.</td>
</tr>
<tr>
<td>Sensor</td>
<td>Reset</td>
<td>Reset the rotation angle to zero.</td>
</tr>
</tbody>
</table>
```
ROBOT MEMORY

- Open Memory Browser
- Tools - Memory Browser

- Shows projects & memory allocation
Space Travel Mission M01

Background
Incredible engineering accomplishments like space travel come about in steps. And many huge, progressive sub-goals need to be met before we can forever leave earth and live to tell about it!

Mission
The robot needs to send payload rockets (carts) rolling down the space travel ramp. The first cart is preset and ready to go, but the robot needs to load the other two from base.

Food Production Mission M10

Background
Gardening is easy, right? You just need a truckload of rich soil, some rain, sun, fertilizer, helpful bugs, CO₂ and a rake but what if you were orbiting Neptune, in a room the size of a minivan?

Mission
Move the push bar the right distance at the right speed, to get into the green scoring range.

Think of your program as making an animated movie with individual shots that move the robot from one waypoint to the next.

You debug each shot, add more character to each, then string them all together into a mission.

Here is an example of moving out of base to the first waypoint. Notice how we document the movement.
Help Your Teams Develop Navigational Skills
- Identify routes and **waypoints** to missions
- Identify and compensate for **error zones**
- Alignment blocks in base
- Find a line, find an object
- Follow a line
- Follow a wall, bump the wall, bump a corner
- The robot knows where it is on the mission field
- Teachers: Goto [oregoncsta.org](http://oregoncsta.org) for computer science summer workshops, including LEGO robotics.
- Classroom Materials here: STEMRobotics:
  - [https://stemrobotics.cs.pdx.edu/node/9360](https://stemrobotics.cs.pdx.edu/node/9360) for ChromeBook.
- Good Luck, and we will see you all at the Qualifying Tournaments!
Mission 10 waypoints 2,3,4