

Design Competition 2014 Workshop 2: Makerbot and Simulation Sensors

Milestone 2 due by Wednesday, 2/19/14, at 5pm, by email or demonstration to Nick

Task:

1. Design **two simple parts** to mount to the 3V worm-drive motor (get one from Nick)
2. Use the **Sonar** sensors to create a wall-following robot controller.

Download:

1. DC14Workshop2.zip, containing DCsonarControl and maze3.

Using the Makerbot

- The Makerbot Replicator 2 is a hobby-quality 3D printer. The minimum (reliable) layer height is 0.2mm, and print material is PLA (not ABS). There are also some exotic materials, like NinjaFlex flexible filament and wood / plastic filament mixtures.
- Printing is pretty easy once you have designed a part and created an STL file. First let's talk about how to print, then we'll talk about how to design for printing.

Printing an STL file

- Download and install **Makerware** from <http://www.makerbot.com/makerware/>
- Our printer is the Replicator 2, so make sure it says that in the bottom right.
- Use 'Add' to insert your part. Make sure it imports with the correct looking size (the default is mm, if you designed your part in inches it should detect that and let you know, but keep an eye out)
- Move your part so that the flattest side is on the platform (be sure to use '**Move to platform**'). Try to orient the part to minimize the number of overhangs. Also try to put the parts on the left or right side of the build platform, which has a slight dip in the middle that can mess up a print.
- When all of your parts are in place, select 'Make'. Use the '**Standard**' settings under the advanced options. Up the temperature to **240 degree C**. I like to check the '**Raft**' box to make it easier to remove the part from the build platform. You can try out the 'Support' option as well, to see how it helps with overhangs, but it can be annoying to remove the support material.
- There are 2 flexible parameters, '**Infill**' and '**Shells**'. A part does not print solid – that is unnecessary and wasteful. You can choose the density, called infill, which will print as a honeycomb filling. This keeps your part light, strong, and printing faster. The outside wall thickness, regardless of density, is the shell parameter. This will determine how many layers of filament get laid down before starting the infill. 10% infill and 2 shells are standard – sometimes increasing shells makes for a stiffer feeling part.
- The file will export as a .X3G file type. Put the .X3G on the SD card using the **USB to SD adapter**, then put the SD card in the Makerbot, and navigate to the file on the Makerbot menu. Note that the button is very sensitive.
- Watch the first couple of layers to make sure the filament sticks. Make sure the filament reel doesn't get all twisted up in the back. If something looks wrong, cancel the print, scrape off whatever printed, and try again.
- The '% done' on the menu represents how many layers are done. Some layers take longer than others.
- Be careful removing the part from the platform. Sometimes it is useful to use a razor, and it is very easy to cut yourself.

Design for printing

- How you make a part is specific to your CAD program. NU teaches Solidworks and NX, so you should be able to get one of those for free (see the attachment in the group email for Solidworks). You can also try 123D Design, Sketchup, and openSCAD.
- When you are designing a part, keep in mind how the 3D printing extrusion process works. If your part has an overhang, the cooling fan will freeze it in place, even though there is no material underneath to support it. The bigger the **overhang**, the more likely the part is to **droop**. The 'Support' checkbox can help with this, but you

can also design your own support in your part, and snap it off after printing. If you can, design a part that requires a large overhang in two parts with little or no overhang, and assemble them into a single part after they are printed.

- Parts tend to print a little large – a 1" cube will be ~1.05" on each side. A hole in the cube that is 0.5" in diameter will be ~0.45" in diameter (and note that the hole will not be perfectly round). The amount that the dimensions are off by, here ~0.05", will change with temperature, filament brand and color, and who knows what else. Try to design your parts to be less sensitive to these inconsistencies, and **undersize parts / oversize holes** if you need to. Before printing a big, complicated part, **print some small test parts** to get measurements off of, and use those measurements to correct your larger design.
- You can design a part to be flexible, such as clip-type parts. Make a few test parts to see how narrow a part needs to be so that it can still flex and be strong. Also try the NinjaFlex filament (although this may take even more experimenting).
- The Makerbot does not have enough resolution to print threads or a tapped hole, but if you print a through hole, and use 3 or 4 shells, you can tap the hole like you would a normal plastic or metal part. We also use brass threaded inserts to fit into holes, for a very strong tapped hole. Ask Nick to see an example.

Makerbot Assignment

- **Design and print two parts** – a mount for the 3V worm drive motor, and a coupler to fit the odd output shaft of the motor. These don't need to be usable parts, and don't glue them on. Just get some practice in CAD and using the printer.

Now, more robot simulation...

Wall-following

- A key ability in all mobile robotics is **wall-following**. Think about navigating down a hallway while blindfolded. Hold out your left arm to touch the wall, and walk. If the wall is within a comfortable distance away, keep walking forward. If the wall gets too close, turn right and go forward slowly until it is arm-length away again. If you lose the wall, turn left as you walk forward slowly until you find it. Keep your right arm out in front of you. If you feel a wall with your right arm as well as your left arm, turn right until you only feel the wall with your left.
- We can break this task into even smaller chunks. Take one step forward. Check the distance to your left. Check the distance to your front. If left and front are too close, turn right. If left is too close, shuffle right. If left is too far, shuffle left. If you are a comfortable distance from the wall, step forward. And continue.
- This set of instructions is called a '**state machine**' – a fancy term for 'really big if...else if' bracket. It is a very common method for controlling a mobile robot. The trick is to be very clear on the possible states, and what action to take for each state.

Visualizing the algorithm

- One way to visualize the states is to draw a big table, with inputs in the columns on the left, and outputs in the columns on the right. For example:

Left Sensor	Right Sensor	Left Wheel	Right Wheel	Comment
Close	Close	Backwards	Forwards	Turn right
Close	Far	Off	Forwards	Turn right
Far	Far	Forwards	Off	Turn left
Good	Far	Forwards	Forwards	Go straight

Using Sonar for wall-following

- This can be done with bump sensors as well, but the Sonar keeps us away from the wall and prevents the robot from accidentally getting stuck on the wall.

- I find that the function SetFwdVelAngVelCreate doesn't work well – try SetDriveWheelsCreate(serPort, rightWheelVelocity, leftWheelVelocity) instead.
- When you sense that you are too close or far from the wall, **stop, back up, turn, then continue** forwards. Keep it simple! Get it to work, then try to maximize your speed and efficiency.
- See the sample code for an example of calling the Sonar and SetDriveWheelsCreate functions.

Simulation Assignment

- Create a wall-following controller to solve maze3, starting in the top middle, and completing a loop around the inner wall.
- What happens if you start outside of the comfortable distance zone?
- Draw a table or flow chart that represents your controller.
- Extra credit: make your controller work without backing up after a sensor threshold is passed. This is trickier than it sounds...

In Workshop 3 we will learn how to design 4-bar linkages, how to incorporate bearings, screws and nuts, and RC motors in designs printed with the Makerbot, and continue to improve the maze solving robot, this time using LIDAR.