# ROB521 -- Connecting to the University Computers

This semester, we will be completing part of the labs using a simulated version of our robots. The robot platforms we would normally be using in the course are the TurtleBot 3 Waffle Pi. The TurtleBots are faithfully rendered in the simulation and your coding for sensor measurements, estimation and control will be the same as though you would be applying it to the real robots. You will be using the Robot Operating System (ROS), which will be presented in a companion document released on Querucs.

The simulation environment we have developed for the course is based on Gazebo, a 3D robotics simulator. You need not concern yourselves with the details of the modeling. We provide the options of interfacing with the lab computers through a web client, noVNC, using the UTORvpn. The aim of this tutorial is to provide background on the workings of noVNC. Gazebo will be running on the lab computers dedicated to ROB521. Using noVNC will allow you to use the Gazebo simulation environment, which resides on the lab computers, from your local computer at home or from wherever else you choose to do your work.

# Getting Access to the Lab Computers

Each lab computer will be assigned to two groups from different practical section in ROB521 who will be using two different accounts. To minimize time sharing between the two groups on one computer, we assign groups running labs from different weeks to share a desktop. For example, a group from PRA0101 will share a desktop with a group from PRA0102.

The lab computers can allow multiple groups access at the same time. However, the computers will not handle well when running more than two simulations at once. Therefore, **please be mindful and do not run simulations for extended hours unless your lab session is within a week**. Moreover, we ask you **not to log in** during other groups' lab sessions. This will at least ensure that each group will have dedicated access during their lab session. TAs will primarily be available during the lab sessions. The IP address, VNC port number, user account name, and password of the computer for your group will be communicated to you.

# Gazebo and ROS

For ROB521, you will be operating your robot using ROS and running it in our simulation environment. Gazebo, the platform on which the environment is built, is a 3D robotics simulator that allows robots to be simulated in complex indoor as well as outdoor environments. It's typically used for designing and testing robots under realistic, dynamically authentic conditions. You may use any of the methods highlighted in the Options for Setting Up ROS document on Quercus to get access to ROS and Gazebo. The lab starter files will be provided through Quercus. Below, we will go over the using our campus computers option.

# Using the Lab Computers Remotely

The first step in connecting to the lab computers is ensuring that you are connected to UTORvpn. You can set it up following the instructions here.

Each group will be given an IP address, port number, account, and password specific to that team. After connecting to the UTORvpn using the AnyConnectClient, you can access your user account in a couple of different ways, which we will list here.

We request that you only use and modify files in `~/catkin_ws`, and additionally a data folder for saving or loading data if necessary. Please don't modify other files in your home directory.

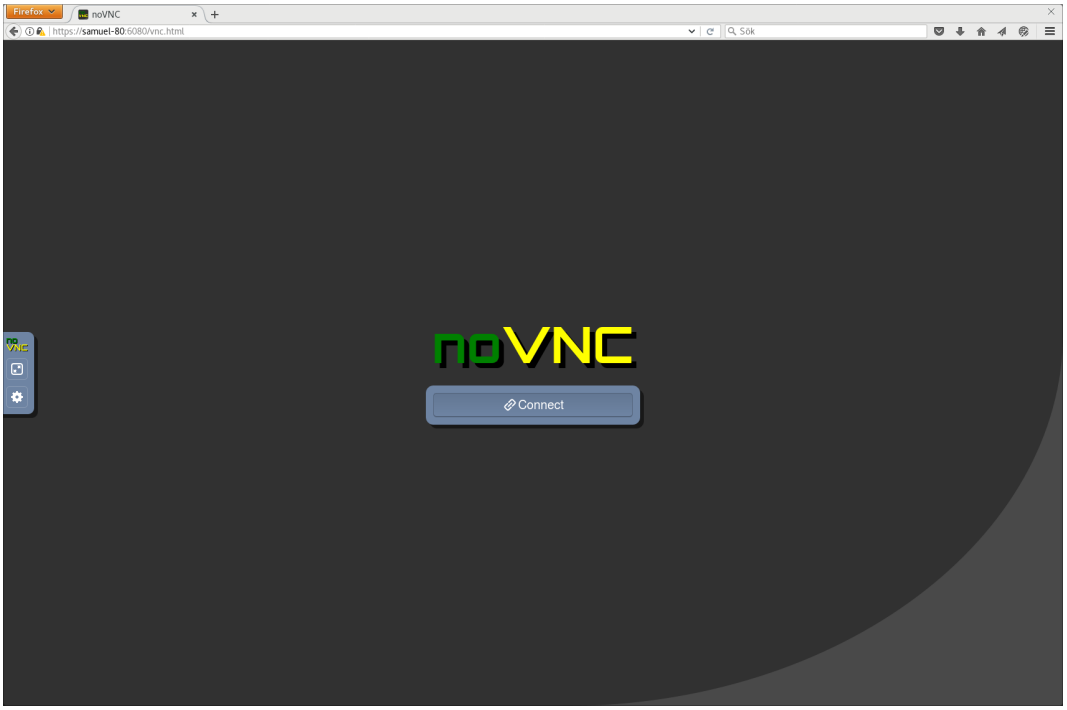
# Using the noVNC Browser Interface

For running anything that requires visualization beyond a simple terminal (e.g. Gazebo, Rviz), you can use the VNC server designated for your group. Note that each group has just a single VNC Server, so if more than one group member logs in at the same time, you will see the exact same desktop screen and running progams.

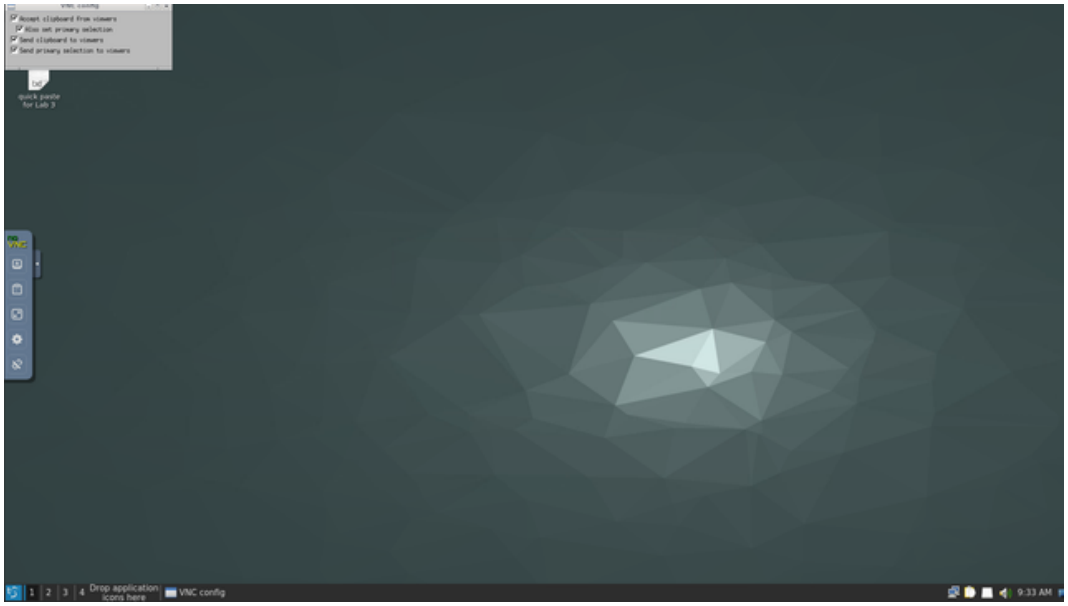
noVNC is a browser-based VNC client for accessing a VNC server without any additional software. To access your group's account, go to your browser and enter the following:

```
http://100.69.127.XX:608X/vnc.html?host=100.69.127.XX&port=608X
```

Make sure to replace the `X`s with the values given to your group on Quercus. You should see the following screen:




Upon clicking connect, you will be prompted to enter your lab group's specific password. The following screen will be the desktop of the lab computer:



If you now see some errors related to "The application shortcut keys has closed unexpectedly" or "Power management crashed too many times", just close these windows -- they won't affect your desktop. If you see notifications for updating Ubuntu 20.04, please close them as well. (Note, there might be more than one such windows. Click as many times as you need to close all of them).

You may need to use the blue toolbar on the left side of the screen to resize screen so that you can see the full desktop. Take some time to explore the desktop. Most software can be opened using the blue hummingird application menu button at the bottom left:

- `terminator` can be found under "System Tools" -- this will be the most important thing for the course.
- `atom`, `gedit`, and `vim` can be found under "Accessories".
- To start vscode, you can enter `code` from within a terminal.

From inside this browser, you are able to access the lab folders, add and edit files, and run programs on the lab computers. You will be using the same desktop and user account for in-person lab experiments so you can put any files you need beforehand.

Now, you are ready for Lab 1. Please continue to Lab 1 handout on Quercus for setting up Gazebo Simulator.

As stated, the computers have a variety of text editors, so if you want, you can complete the labs directly inside this browser window without any other tools. However, you may also want to develop code on your own computer -- for that, there are a few other options for communicating with the lab computers.

# (optional) Using ssh through a terminal

If you have access to a linux-based terminal (through any linux distribution or macos, or with PuTTY or WSL for Windows), you can also use `ssh` to directly access your user account. Simply run

```
ssh user@100.69.127.XX
```

where `user` and `XX` will be communicated to you, and enter the same password that you use for noVNC.

You can now run ROS nodes, copy files back and forth from your local machine with scp or rsync, or use any other linux-based tools you're familiar with.

# (optional) Remote Development

The computers are set up for remote development with VSCode. You can install VSCode on your own computer, and then develop on a remote computer using the Remote-SSH Extension.

> Written with StackEdit.