Set up your computing environment

The following steps describe how to get your computer set up for the MD simulation workshop.

A few of notes on the format.

- Any time I refer to a *local* terminal, this means a terminal navigating your personal computer. When I refer to a *cluster* terminal, this means a terminal navigating the cluster. When you open a new terminal, it is a *local* terminal. After you type `ssh USERNAME@login.talapas.uoregon.edu`, it becomes a *cluster* terminal. (Put functionally: if you type the command `ls` on a *local* terminal, it shows files on your laptop; if you type `ls` on a *cluster* terminal, it shows files on the cluster.)
- In this document, anything in "`this font`" is text that will appear on a terminal (whether typed or printed out as output).
- Anything in "`ALL CAPS`" is a placeholder name you will likely change. For example, the instructions refer to "`PROTEIN.pdb`". You will probably have a file with a different and more descriptive name, but we will refer to it `PROTEIN.pdb` in the instructions.
- In the terminal commands, any text after a "`#`" is a comment. It is not part of a command, but can be safely pasted into a terminal because the computer will ignore any text after `#`.

Set up the cluster

The cluster should already basically be set up for you. The following verifies this is a case and has you launch a test job on the cluster. I am going to assume you know how to do each of the following during the workshop.

1. Use secure copy to copy the file `test-materials/test.srun` to the cluster. On a local terminal, navigate to the directory that has this directory. Then type the following. This will copy the test.srun directory to your home directory on the cluster.

```
scp test.srun USERNAME@login.talapas.uoregon.edu:
```

2. SSH onto the cluster with the following command. After this runs, your terminal becomes a *cluster* terminal.

```
ssh USERNAME@login.talapas.uoregon.edu
```

3. Make a test directory on your `~/shared/` network drive and move "test.srun" into that directory:

```
mkdir ~/shared/test-md/          # make directory
mv test.srun ~/shared/test-md/   # copy file
cd shared/test-md                # change into the directory
```

4. Make sure your environment is working by running `test.srun`. Replace "YOURPIRG" with the PIRG you are a part of (i.e. harmslab):

```
sbatch --account=YOURPIRG test.srun
```

If everything is set up correctly, the terminal will return something like `Submitted batch job 2758295` where the number is a unique job ID. It could take a minute or so to run. You can check the status by:

```
qstat -u USERNAME
```

In the output, the `Use` column indicates the job status. `Q` means the job is queued, `R` that it's running, and `C` that it is complete. The following shows a completed job.

```
Job id               Username Queue    Name    ... Use S Time
-------------------- -------- -------- ------- ...  - -----
2758295              harms    compute  testrun ...  C 00:00
```

When the job finishes (i.e., the `Use` column is `C`), the script should have created three empty files: "hostname.err", "hostname.out", and something like "Fri_Apr_12_11:24:00_PDT_2024" corresponding to the time you ran the script. If something is wrong, "hostname.err" will contain error output to help you troubleshoot.

Set up your personal computer

1. Install the software package VMD (https://www.ks.uiuc.edu/Research/vmd/). You can make sure it is working by opening the file "test-material/output/traj.gro". It should bring up a box of waters around a protein that you can rotate etc. If you want to familiarize yourself with the interface, you can check out the VMD tutorials (https://www.ks.uiuc.edu/Training/Tutorials/vmd/tutorial-html/node2.html).

2. Install MDAnalysis (and the rest of the Python scientific computing stack if you have not already).

   - Download Miniconda https://docs.anaconda.com/free/miniconda/index.html.
   - Install Miniconda. Use a command something like `bash Miniconda3-latest-Linux-x86_64.sh`, where you replace the name of the `.sh` file with the name of the file you downloaded.
   - Install the scientific computing stack by typing: `conda install numpy pandas scipy matplotlib jupyterlab -c conda-forge -c defaults --strict`
   - Install MDAnalysis by typing: `conda install mdanalysis -c conda-forge -c defaults --strict`

3. Open the Jupyter notebook `test-materials/test.ipynb` and execute all cells. It should run without error.