## Gettings setup with NREL’s High Performance Computing (HPC)

Background: NREL’s High Performance Computing (HPC) assets (Kestrel, Eagle, etc.) offer significant computing power and may be required for large-scale or complex REopt models or modeling runs. To access the HPCs, you will need to create an HPC account, request allocation units (AUs) on the HPC, and then run the REopt code through a shell environment on your computer. There is no cost to using HPCs. The sections below summarize the key usage steps.

**Step 1:** Steps for getting setup with an account:

* 1. Request an account here: <https://www.nrel.gov/hpc/user-accounts.html>
  2. After requesting an account, you will receive an email (not immediately because your account has to be manually created by someone on the HPC team)
  3. Follow the instructions in the email to create a new password
  4. Log in using your username and password to the Lex system: <https://hpcprojects.nrel.gov/>

**Step 2:** Requesting HPC allocation

* 1. You can request HPC allocation units (AUs) using the Lex system (linked above)
     1. For a project that is just starting, submit a Pilot request, which is by default 150,000 AUs
     2. Complete the fields for the pilot request. List Julia, JuMP, and Gurobi (if using Gurobi) in the section for “software requested.”
     3. Note: A simple REopt run (PV and BESS sizing without resilience, for instance) uses less than 1 AU. A complex REopt run might use around 10 AUs.
     4. Note: Each REopt project needs to have a separate request for AUs (multiple separate REopt projects cannot be combined into the same allocation unit request)

**Step 3:** Running REopt on the HPC

**These steps are currently written for PCs, and the steps may differ for Macs.**

* 1. Create two scripts for running REopt on the HPC:
     1. Create a julia script that runs the REopt program
        1. From the julia script, you can export a JSON with all of the REopt outputs (the JSON will be saved on your project folder on the HPC). Then you can manually copy the JSON file (using WinSCP) onto your local computer file system and do post processing/plotting of the results locally on your computer. Alternatively, if you are running many computations and the JSON files are too numerous and/or too large, then you can do the
     2. Create a shell script that connects to the HPC and calls the julia script
        1. You can use partition “debug” for initial testing. Partition “short” is typically a good fit for REopt runs.
     3. See examples for these scripts on the teams folder here: [ExampleCode](https://nrel.sharepoint.com/:f:/r/sites/REopt/Shared%20Documents/General/Team%20Resources/HPC%20with%20REopt.jl/ExampleCode?csf=1&web=1&e=cRTAcj)
  2. Access and run code on the HPC through GitBash
  3. Open GitBash and use ssh to log into Kestrel:

ssh <your username>@kestrel.hpc.nrel.gov

**Note**: It might take a day or two after creating your HPC account for you

to be able to log in to your account in HPC

**Note**: When you type in your password when logging it, the characters

typed will not show up (but the password is still being entered)

* 1. Type “cd /projects” to access the projects folder



**Note**: use “cd ..” to go back a folder

* 1. The type cd modelhydro/ (replacing modelhydro with your project name, which was used to get allocation from the HPC)

A picture containing text

Description automatically generated

* 1. Type “module spider julia” to see the available julia versions available on HPC
  2. Then type “module load julia/1.8.5-generic-linux” or whichever version of julia you would like to load

Text

Description automatically generated

**Note:** The “ls” (that is a lowercase L) will show what files are currently

located in the modelhydro environment

* 1. Create a julia environment in the folder on the HPC that has the julia script and shell script

Write this command: “julia --project=.”

Text

Description automatically generated

* 1. Add the necessary packages to that environment, such as REopt, JuMP, Gurobi, DelimitedFiles, JSON, etc.

Note: There may be additional steps for using Xpress instead of Gurobi

This screenshot shows several packages being added to the environment on the HPC:

Text

Description automatically generated

* 1. If using a feature branch for REopt.jl, switch to that feature branch in the environment, using this code in the julia package manager in your project’s HPC environment:

rm REopt -this removes the REopt package

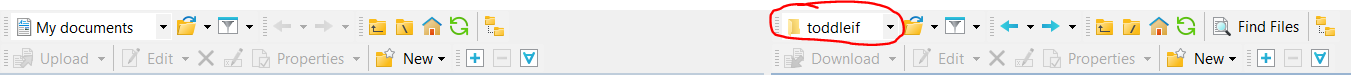
add REopt#branchname -this adds a particular branch to the REopt

model (replace “branchname” with the

name of the REopt branch)

Note: if you update the feature branch locally and commit/push it to GitHub, it appears that you need to remove and re-add REopt to the HPC environment to get the updates in the HPC environment

* 1. Download WinSCP in order to be able to easily access the files in the HPC file system; you can go to ITS and they can download and install it for you in just a few minutes.
     1. Here is the link for WinSCP: <https://www.nrel.gov/hpc/winscp-file-transfer.html>
     2. After opening WinSCP, connect to the Kestrel HPC host: kestrel.hpc.nrel.gov
     3. Sign in using your HPC username and password
        1. List your username without the @nrel.gov or @ kestrel.hpc.nrel.gov after it
     4. Within WinSCP, your computer’s local files will be shown on the left, and the files on the HPC will be shown on the right
        1. By default, your personal folder in the HPC file system is shown in the window to the right
        2. However, you should navigate to your project folder in the HPC file system. To do so:
           1. Click the dropdown (shown below circled in red) and then select the “/<root>” option
           2. Then open the “projects” folder, and then open the folder with your HPC project name
        3. Copy and paste your julia and shell scripts into your project folder in the HPC file system



**Note:** WinSCP might not be available or easy to implement on Mac. Another way to interface with the HPC file system might be to create a repository through GitHub (I think).

* 1. Going back to the GitBash window, run the shell script, using this command:

sbatch submit\_jobs.sh (replace submit\_jobs.sh with the name of your

shell file)

When the job is submitted, the gitbash will return something like:



* 1. Use the following commands to check the status of your run and/or the status of the partition

Text

Description automatically generated

* 1. If you want to cancel a job, you can cancel is using this method:
     1. More info: <https://slurm.schedmd.com/scancel.html>
        1. See examples at the bottom of that webpage page
     2. Can cancel jobs by specifying the Job ID (which is 4313886 below)



* + 1. Can also cancel using your HPC account name using this code (replace toddleif with your own HPC account name please). This likely cancels every job in the queue that was submitted by the specified user.



* 1. Depending on the settings you defined in the submit\_jobs.sh file, you might receive an email when the job is started on the HPC and when it finishes
     1. After the HPC run finishes, you can check the slurm file (which is saved into your HPC system project folder) for any information about the run (stacktraces, errors, any print statements included in the code, etc.)
     2. If you saved a csv (or other file type) with outputs from the REopt model, you can find those in your project folder on the HPC