

RIFT Tutorial

02: Cluster & Environment Setup

Katelyn J. Wagner

RIFT, **R**apid parameter inference on gravitational wave source via **I**terative **F**iTting, is a parameter estimation pipeline that aims to recover source parameters for coalescing compact binary sources. This tool is highly parallelizable and uses iterative processes to rapidly converge on the parameters of many types of binaries.

Technical Setup

The LVK generously provides computing resources for member groups through the LIGO Data Grid. Access to cluster computing is necessary for the following tutorial. Begin by signing into the cluster - we recommend CIT (Caltech). You can SSH into the general grid and select your cluster/machine:

```
1  ssh albert.einstein@ssh.ligo.org
```

or choose your machine at SSH:

```
1  ssh albert.einstein@ldas-grid.ligo.caltech.edu
```

Here, replace `albert.einstein` with your `first.last` credentials. When you sign on, you should see an IGWN environment activated, i.e. your bash will look something like:

```
1  (igwn) [albert.einstein@ldas-pcdev3 ~]$
```

This conda environment is typically activated for all users by default and (which can be changed by adding a file called `.noigwn` and then logging back in) and contains some latest stable set of packages and versions that have been approved for use. See more about this [here](#). For now, you can deactivate this environment with `conda deactivate`. Now we will use `mamba` to create a conda virtual environment that you will use for your work. Using something like a virtual or conda environment gives the user an isolated development environment where they have full control over the packages contained within it. To begin, create a conda environment:

```
1  mamba create --clone igwn-py310 --name YOUR_ENV
2  conda activate YOUR_ENV
```

This created a conda environment called "YOUR_ENV" and activated it. It creates a copy of the existing `igwn-py310` environment, giving it a new name (YOUR_ENV). You're effectively cloning an existing conda environment using mamba, which is just a faster drop-in replacement for conda. Your bash should now look something like the following:

```
1 (YOUR_ENV) [albert.einstein@ldas-pcdev3 ~]$
```

Now that your development environment is ready, you will need a copy of the RIFT repository.

```
1 cd ~/.conda/envs/YOUR_ENV
2 mkdir src
3 cd src
4 git clone https://github.com/oshaughn/research-projects-RIT.git
```

The RIFT repository is kept in two places, one on GitLab and one on GitHub. This is the GitHub version, which tends to see changes to the source code first. You want to check out a branch to track `upstream/rift_04c` since that is where the most up to date scripts live that you will use. Then, install the necessary dependencies:

```
1 cd research-projects-RIT
2 git checkout rift_04c
3 pip install -e .
```

You will also have to set a few environment variables and paths before you begin. Later when you build a RIFT run directory, some of these will be accessed by the computing grid submit files. To set these variables in bulk, create a file called `setup.sh` using your favorite text editor. This script should contain the following:

```
# User specific aliases and functions
HOME=/home/alber.einstein
export LIGO_ACCOUNTING=ligo.sim.o4.cbc.pe.rift
export LIGO_USER_NAME=alber.einstein
CUDA_VER=11.2
export LD_LIBRARY_PATH=/usr/local/cuda-${CUDA_VER}/lib64/
export CUDA_PATH=/usr/local/cuda-${CUDA_VER}
export CUDA_DIR=/usr/local/cuda-${CUDA_VER}
export SINGULARITY_RIFT_IMAGE=/cvmfs/singularity.opensciencegrid.org/james-clark/
research-projects-rit/rift:test
export SINGULARITY_BASE_EXE_DIR=/usr/local/bin/
export RIFT_REQUIRE_GPUS='(DeviceName!="Tesla K10.G1.8GB")&&(DeviceName!="Tesla
↪ K10.G2.8GB")&&(DeviceName!="Tesla K20Xm) '
export RIFT_AVOID_HOSTS=`cat
↪ /home/richard.oshaughnessy/igwn_feedback/rift_avoid_hosts.txt | tr '\n' , | head -c
↪ -1`
export NUMBA_CACHE_DIR=/tmp
```

```
export RIFT_GETENV=LD_LIBRARY_PATH,PATH,PYTHONPATH,*RIFT*,LIBRARY_PATH
export RIFT_GETENV_OSG=NUMBA_CACHE_DIR,*RIFT*
```

NOTE: The steps up until this point only need to be performed the first time you set up your RIFT directory and environment, unless you wish to create a fresh install.

Some of these variables are for using the OSG. You may also need to add additional variables for accessing and using external waveforms (such as `TEOBResumS` or numerical relativity waveforms), but check with someone in the group for more information. Setting your path specifies a particular set of directories where your executable files are located. RIFT uses these paths to locate the appropriate scripts, since many of the relate to or import each other.

You must source your environment and your setup script each time you wish to do any RIFT-related tasks. If you wish, this can be done in one step by adding the following lines to the top of your `setup.sh` script:

```
1  #!/bin/bash
2  conda activate YOUR_ENV
```

and then source your setup script each time you login as usual with `[source setup.sh]`, to activate your environment *and* set your paths.