

# Processing Pulsar Data

## CDMT (Stokes I) Method

The GPUs in `ucc4` are typically used to reduce the majority of the pulsar data produced by the station into mode 5, 8x channelised, 16x downsampled Sigproc Stokes I filterbanks. This is achieved through using a `python` script, `cdmtProc.py`, to produce a `bash` script with the correct flags needed to call `CDMT` with maximum performance, while not overutilising the GPU's VRAM.

The script needs to be provided a path to the recording folder, and optionally an extension to add to the file name. The resulting `bash` script can then be executed. Once data is analysed, the `compress.sh` and `transfer.sh` scripts can be used to compress the raw data using `zstandard`, then transfer it to the DIAS archive.

```
obs@ucc4:~$ tmux new
obs@ucc4:~$ dckrgpu

root@b02cf5b93fbc:/home/user# cd /mnt/ucc4_data2/data/David/
root@b02cf5b93fbc:/mnt/ucc4_data2/data/David# mkdir rrat_2023_05_30
root@b02cf5b93fbc:/mnt/ucc4_data2/data/David# cd rrat_2023_05_30
root@b02cf5b93fbc:/mnt/ucc4_data2/data/David/rrat_2023_05_30# cp ../cdmtProc.py ./
root@b02cf5b93fbc:/mnt/ucc4_data2/data/David/rrat_2023_05_30# python3 cdmtProc.py -i
/mnt/ucc1_recording2/data/rrats/2023_05_30/ --extra 1
root@b02cf5b93fbc:/mnt/ucc4_data2/data/David/rrat_2023_05_30# ll
total 92
drwxr-xr-x  2 root root 4096 May 30 15:48 ./
drwxr-xr-x 129 1000 1000 69632 May 30 15:47 ../
-rw-r--r--  1 root root 14501 May 30 15:47 cdmtProc.py
-rw-r--r--  1 root root   886 May 30 15:48 cdmtProc_1.sh
root@b02cf5b93fbc:/mnt/ucc4_data2/data/David/rrat_2023_05_30# bash cdmtProc_1.sh

<data is processed>

root@b02cf5b93fbc:/mnt/ucc4_data2/data/David/rrat_2023_05_30# bash ../compress.sh; bash ../transfer.sh
```

## Single Pulse, Periodic Emission Searches

Two scripts in the `/mnt/ucc4_data2/data/David/` directory can be used to setup single pulse searches.

The `generateHelmdall.py` script will open a number of matplotlib interfaces with bandpasses for RFI flagging before writing a list of narrow and wide DM searches for single pulses data. When flagging the data, each click on the bandpass will flag a subband worth of data (8 channels), and a range of subbands can be flagged by clicking somewhere, moving the mouse to the end of the range, then pressing "a" with the window active. The output script will be in the `cands` subdirectory of the input folder.

The candidate files are then used to generate plots of each pulse that fits certain criteria (typically  $\text{SNR} > 7.5$ ,  $\text{width} < 2 \times 10$  samples). This process is automated with the `frb_v4.py` and `generateCands.py` scripts.

```
dmckenna@ucc2:/mnt/ucc2_data1/data/dmckenna/working_cands$ mkdir rrat_2023_05_30
dmckenna@ucc2:/mnt/ucc2_data1/data/dmckenna/working_cands$ cd rrat_2023_05_30
dmckenna@ucc2:/mnt/ucc2_data1/data/dmckenna/working_cands/rrat_2023_05_30$ cp ./frb_v4.py .
dmckenna@ucc2:/mnt/ucc2_data1/data/dmckenna/working_cands/rrat_2023_05_30$ for i in {1..4}; do
python3.8 ./generateCands.py -i /mnt/ucc4_data2/data/David/rrat_2023_05_30/ -o proc_ucc4_${i}.sh; bash
proc_ucc4_${i}.sh; sleep 3600; done
```

The `generatePrepolds.py` script will generate a list of `rfind` and `prepold` commands for both set targets, and targets within 1 degree of the original pointing. The output script will be in the `olds` subdirectory of the input folder.

Single pulse candidates and folded profiles can then be copied locally, with folded profiles converted from `ps` files to `png` using the `grabFolds.sh` script.

## udpPacketManager/digifil (Stokes IQUV)

### Method

For full Stokes IQUV data, we use `udpPacketManager` to convert the raw CEP data into a `DADA` data block, which is then processed with `digifil`. The script for automating this need to be modified to contain a dispersion measure for coherent dedispersion, and a downsampling fact for the data. Afterwards, the script can be run, pointed at a specific directory, and it will handle the rest for you.

```
obs@ucc1-~$ tmux new
obs@ucc1-~$ bash /mnt/ucc1_recording2/data/rrats/4pol/enter_docker.sh

root@106316e58e36:/home/user# cd /mnt/ucc1_recording2/data/rrats/2023_05_30/
root@106316e58e36:/mnt/ucc1_recording2/data/rrats/2023_05_30# mkdir tmp_processing
root@106316e58e36:/mnt/ucc1_recording2/data/rrats/2023_05_30# cd tmp_processing
```

```
root@106316e58e36:/mnt/ucc1_recording2/data/rrats/2023_05_30/tmp_processing# cp ../4pol/* ./
root@106316e58e36:/mnt/ucc1_recording2/data/rrats/2023_05_30/tmp_processing# bash 4pol_generate.sh

# Alternatively,
root@106316e58e36:/home/user# cd /mnt/ucc1_recording2/data/rrats/prcoessing/
root@106316e58e36:/mnt/ucc1_recording2/data/rrats/processing# mkdir 2023_05_30
root@106316e58e36:/mnt/ucc1_recording2/data/rrats/processing# cd 2023_05_30
root@106316e58e36:/mnt/ucc1_recording2/data/rrats/processing/2023_05_30# cp ../4pol/* ./
root@106316e58e36:/mnt/ucc1_recording2/data/rrats/processing/2023_05_30# bash 4pol_generate.sh
/mnt/ucc1_recording2/data/rrats/2023_05_30/
```

---

Revision #7

Created 30 May 2023 15:26:32 by David

Updated 1 June 2023 10:21:43 by David