

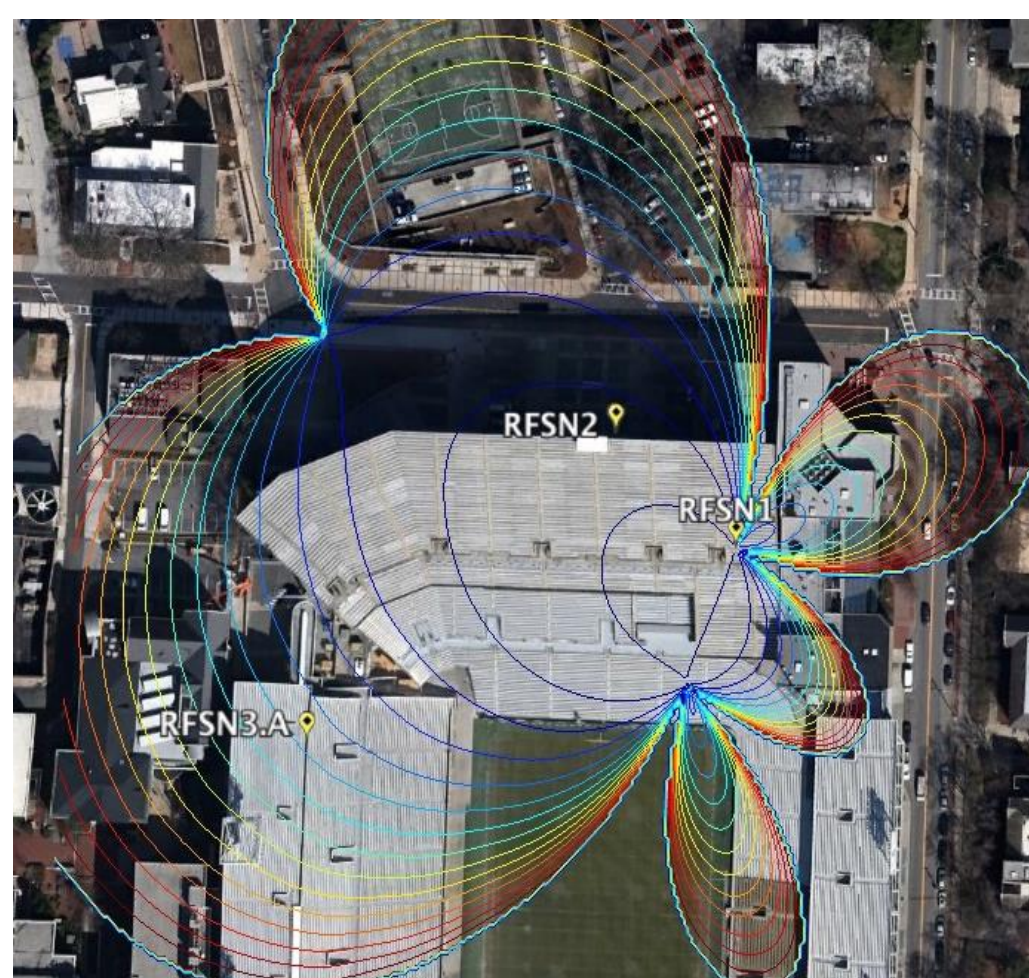


Tools for High Sample Rate Recording and Post-Processing: gr-analysis

Orin Lincoln and Paul Garver

Why gr-analysis?

- We work with **large data files** from football games; the recordings may be several hours long.
- We also have multiple recordings of the same environment from a different location, therefore **precise timing information** must be preserved throughout data processing.
- We needed a convenient, integrated set of tools to simplify post-processing of large amounts of data.
- Tools provided in gr-analysis consist of C code, Python and MATLAB scripts, and they leverage **metadata** to complete their task.



High Sample Rate Capture

- Over extended periods of time (> 30s), the built-in GR blocks fail to perform well at high sampling rates.
- Even without metadata recording, the built-in tools for capturing data fall behind at high rates.
- Capabilities of uhd_rx_cfile and specrec while recording data and metadata at various rates:

Sample rate (Mps)	uhd_rx_cfile	Specrec
5	✓	✓
10	✓	✓
15	*	✓
20	X	✓
25	X	✓
30	X	*

*works without metadata

Metadata (data about data)

- GNU Radio defines a convention for metadata headers
- Each piece of metadata is a key-value pair in a dictionary (i.e. “rx_rate” -> “10e6”)
- Headers could be inline (stored within the same file as data) or detached (stored in external file)
- Each header is expected to contain some standard items (sample rate, receive time, etc.), but can also contain extra key-value pairs

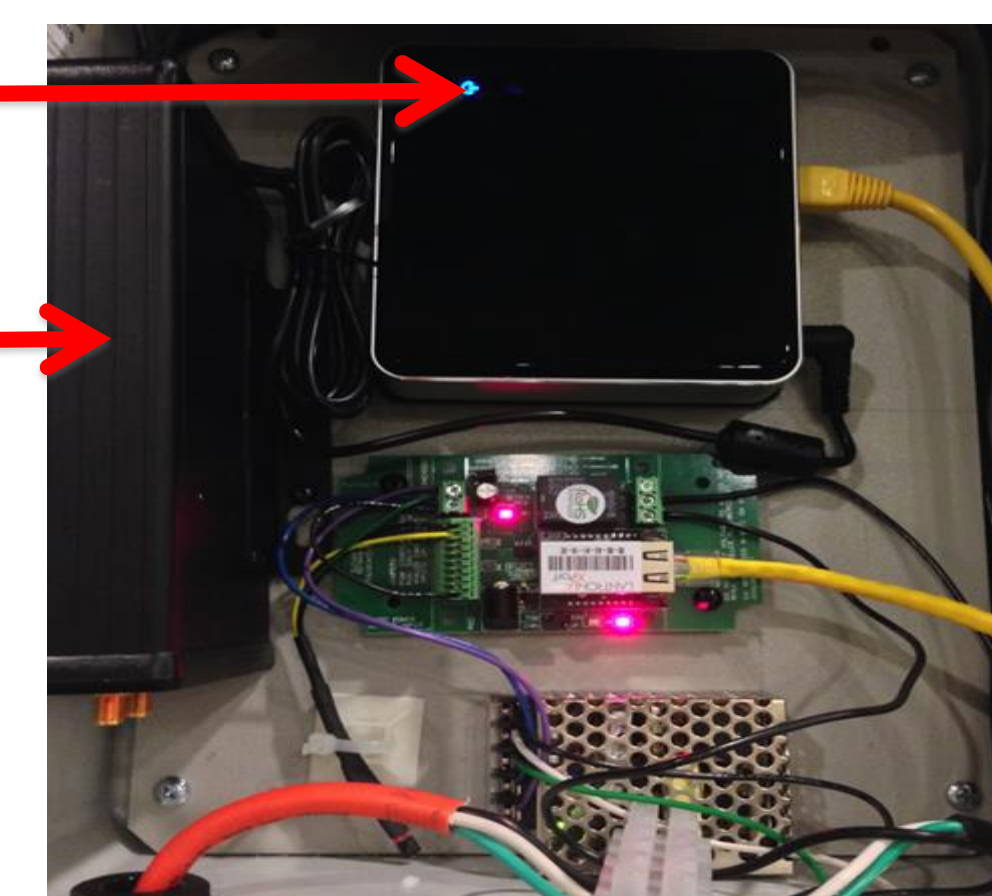
Hardware setup



Spectrum-sensing node and antenna are deployed in the GT football stadium.

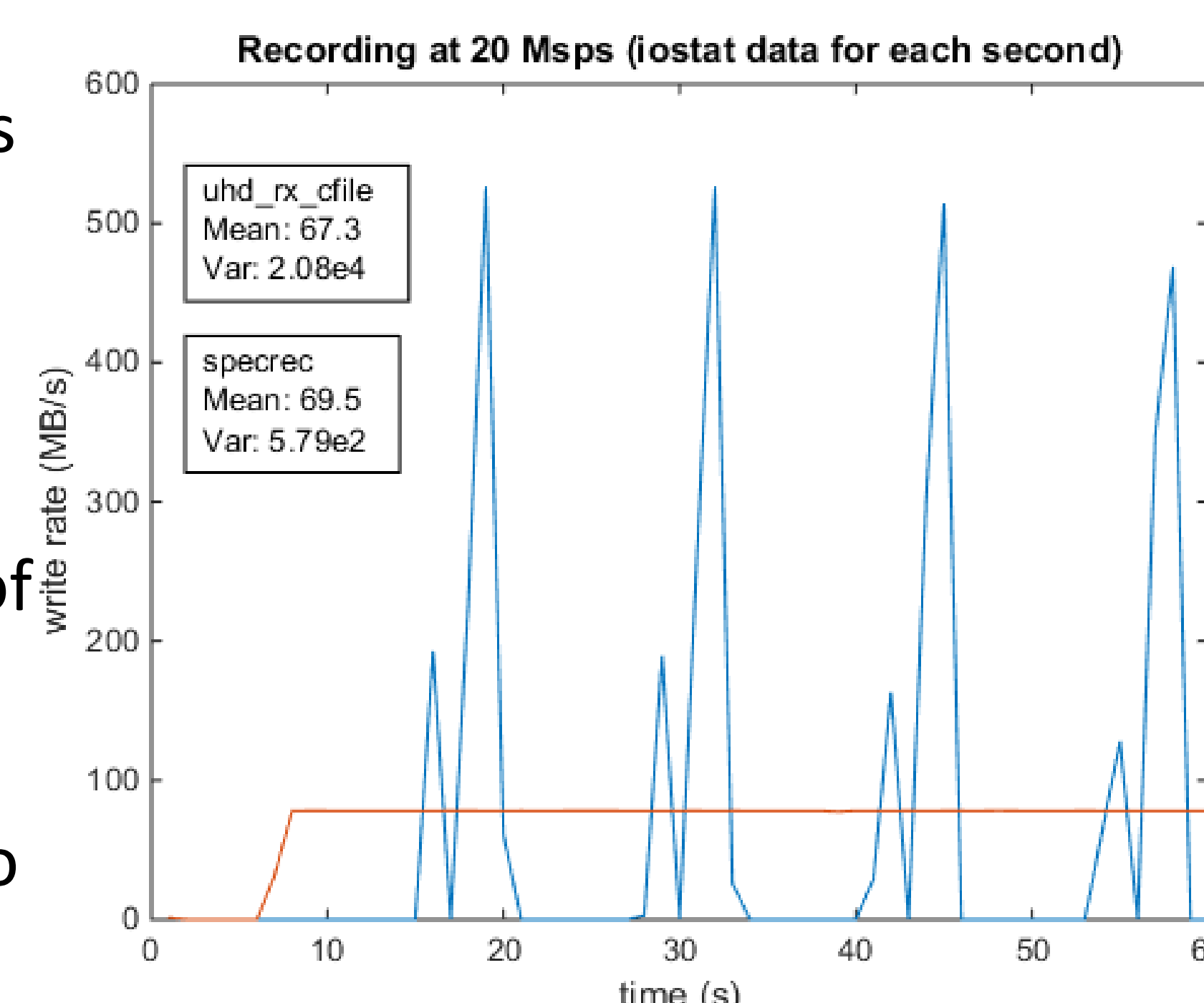
Intel NUC PC

USRP B200 with GPSDO



specrec

- gr-analysis’s specrec tool is capable of recording data and metadata at much higher rates.
- Uses producer-consumer model to take advantage of concurrency
- Also uses Linux-specific sync_file_range call to force page-aligned writes
- Storage media specifications are often unsustainable over long periods of time; maximum write speed may be less than advertised.
- uhd_rx_cfile experiences writing in bursts, which may exceed storage specifications, leading to overflows.
- specrec writes continuously, with a lower maximum rate than uhd_rx_cfile demands



fileman (to manipulate files)

- Reads header to truncates data files to slices based on time or sample number
- Interpolates timestamped data
- Converts between data formats (fc32 and sc16). For hardware reasons, may want to record data as sc16, but most tools prefer data as fc32.
- Ex: A data recording (fc32 format) for 5 mins at 25 Msps is around 60 GB.
- Convenient to take smaller slices of the file to process individually

metadata_to_csv

- Converts metadata from GNU Radio header format into a CSV table of metadata values
- Allows for easy import of metadata into MATLAB/Octave
- Can specify extra key-value pairs (in addition to standard) which are needed for CSV output

mkheader

- Some of our data was recorded without GNU Radio metadata headers
- mkheader allows generation of headers after the data has been recorded, in order to take advantage of other tools which leverage metadata
- Can also generate “fake” headers for other programs which require headers to operate (i.e. fileman)