Data associated with Hughes et al. (submitted)

A turbulence data reduction scheme for autonomous and expendable profiling floats

Submitted to *Ocean Sci.*

During field tests in 2019, we deployed two FCS floats (units 4002 and 4003). Between the two, approximately 650 profiles were taken. For each profile, we are archiving five configurations of the data. We are providing these in both Matlab `.mat` format and NetCDF format. The content is the same, but the formatting differs.

For the Matlab files, each of the five configurations is contained within a struct within the same Matlab `.mat` file. (An example of how to read in this file in Python is given at the end of this document.) For the NetCDF files, the variable names are prepended with the names of the structs. For example `avg.S1` in the Matlab file corresponds to `avg_S1` in the NetCDF file.

To facilitate downloading from Zenodo, the files are grouped into days (May 13, 14, 15, 16, and 17) and direction (UP and DN).

Struct name	Type of data
`data`	The raw data in voltage units
`blk`	The calibrated data in 512-element blocks
`blkr`	The raw voltage data in 512-element blocks
`phi`	Spectra in physical units (associated with segments in `blk`)
`avg, avgr`	The binned data including derived turbulence quantities

In the Matlab files, `blk`, `phi`, `avg`, and `avgr` contain readmes with full details about the individual quantities that were measured or derived. In the NetCDF files, these details are attributes associated with each variable.

The structs `avg` and `avgr` are similar, but `avg` was derived with a standard processing approach, whereas `avgr` was derived using the data reduction scheme

1

described in the paper. The number of depths bins may differ between the two due to differences in the calculation of $W_{\min}$ and hence the identification of the end of the profile. Further, `avgr` (but not `avg`) includes bad values of ϵ and χ in the lowest tens of meters of upward profiles when the ballast pump is running.[1]

Both `blk` and `blkr` include shear data after deskiping. The despiking routines, however, are not the same. See Appendix E of the associated paper for details.

Each Matlab file also includes the struct `head`, which is a header file that includes calibration coefficients.

Each filename is of the form `UD_YYYYMMDDhhmmss_00X.ext` where `UD` is `UP` or `DN`, `YYYYMMDDhhmmss` is the time stamp when the raw file was first created, `00X` is `001` for unit 4002 and `002` for unit 4003, and `.ext` is either `.mat` or `.nc`.

Sea surface height spectra are contained within a substruct of `avgr` within the `DN` profiles in the Matlab files.

Some files, especially early ones, may not contain any values in `avg` or `avg`. These can be ignored.

---

[1]The data reduction scheme does not include a method to identify when the ballast pump is on within individual profiles. Therefore, we have left the bad values in 'avgr'. However, the bad values are easily identified in post-processing by comparing successive casts.

To open a given file in Python and load the data into dicts, use the following:

```python
from numpy import squeeze
from hdf5storage import loadmat


f = loadmat(<filename.mat>)


def fields(struct):
    return struct.dtype.names


def read_array(quantity, struct):
    return squeeze(f[struct][quantity][0][0])


data = dict()
for quantity in fields(f['data']):
    data[quantity] = read_array(quantity, 'data')


cal = dict()
for quantity in fields(f['cal']):
    cal[quantity] = read_array(quantity, 'cal')


avg = dict()
for quantity in fields(f['avg']):
    avg[quantity] = read_array(quantity, 'avg')


avgr = dict()
for quantity in fields(f['avgr']):
    avgr[quantity] = read_array(quantity, 'avgr')
```