NTAS 14 Recovery Initial Data Processing

NTAS 14 was deployed from R/V Endeavor (cruise EN 549) on 12/13/2014 18:27 UTC (anchor drop) and recovered from R/V Endeavor (cruise EN 573) on 2/4/2016 14:08 UTC (anchor released). Deployment duration was 418 days.

I. Overall processing procedure

Data files are in folder *NTAS14\_Recovery*. From there folder *data* contain datafiles and processing codes, and *Figures* contain plots of initial data evaluation for each sensor. On Buoy2 data server, the folder intermediate *NTAS14\_Recovery* is omitted.

In *data*, excel spreadsheet **NTAS14\_Recovery.xls** contains information about recovered instrumentation. Other spreadsheet are derived from it, such as **ntas14\_ss\_insts.csv** which contain lists of subsurface sensor type, serial number and depth, and **N14modhts.csv** which contain lists of surface sensor type, serial number and depth.

Suite of Matlab codes sets metadata, combines with data from each instrument type, and plots data for evaluation. Some codes are also placed in *data/Utilities*.

In *data*, **ntas14meta.m** sets metadata for NTAS14 deployment (time of anchor drop and release, anchor position, magnetic variation value and source, water depth, buoy waterline, etc).

II. Subsurface data processing

II.1. Nortek Aquadopp Current Meters

Three Aquadopps were deployed, listed in Table II.1.a. All had similar setups (measurement interval = 20 minutes; average interval 180 seconds; blanking distance = 1.01 m (0.35 m for inductive sensor AQD 5973); power level = HIGH-); all also sampled at 1 Hz, once a day (diagnostic mode), for 100 seconds (50 seconds for AQD 5973). AQD5973 and AQD432 failed early (see details further down).

Table II.1.a. Nortek Aquadopps data recovered from NTAS 14 and data files output from initial processing.

|  |  |  |  |
| --- | --- | --- | --- |
|  | **AQD 9467** | **AQD 5973** | **AQD 432** |
| **Depth (m)** | 5.7 | 13 | 18 |
| **Raw file** | NTAS14\_AQD\_9467.dat | NTAS14\_AQD\_5973.dat | NTAS14\_AQD\_432.dat |
| **Output file** | NTAS14\_AQD\_9467.mat | NTAS14\_AQD\_5973.mat | NTAS14\_AQD\_432.mat |
| **Observed pressure (dbar)** | 6.3 | 14.4 | 19 |

II.1.a. Processing Sequence

From folder *NTAS14\_Recovery/data/nortek*:

i) In Matlab run **do\_nortek.m.**

ii) In Matlab run **plot\_nortek\_ntas14.m**.

iii) In Matlab run **do\_nortek\_diags.m.**

iv) In Matlab run **plot\_nortek\_diag\_ntas14.m**

Processing steps:

i) calls **ntas14meta.m** to grab mooring metadata. Also calls **get\_ss\_info.m**, which inputs serial numbers and depths of all Aquadopp sensors listed in **ntas14\_ss\_insts.csv** (list ofsubsurface instruments). Calls **get\_aqd.m** to ingest input raw file and output into Matlab (file names are in Table II.1.a).

ii) ingests mat data files created in previous step and plots time-series of velocity components, sensor angles, signal amplitudes, battery voltage, pressure and temperature for all 3 Aquadopps. Figures are saved in *NTAS14\_Recovery/Figures/nortek*.

iii) similar to step i) but deals with diagnostic files. These files have same data as regular data files but collected samples at a fast rate (1Hz for 100 seconds (50 seconds for AQD5973)) once a day.

iv) plots diagnostic data for all 3 Aquadopps. Figures are saved in *NTAS14\_Recovery/Figures/nortek*.

II.1.b. Sensor Evaluation

- AQD9467 has full dataset. AQD5973 and AQD432 records are too short. AQD5973 last record is on 11/9/2015 21:59. AQD432 last record is on 7/2/2015 17:40.

- Battery voltage on AQD5973 dropped from ~11V in early September 2015 to ~8V in November 2015. Battery voltage on AQD432 remained near 11V until July 2015.

II.1.c. Data Evaluation

- Signal amplitude was noisy on beam 2 (July-august 2015) and beam 3 (March-May 2015) on AQD5973. Signal amplitude was noisier and lower on beam 2 (June-September 2015) on AQD9467.

- Signal amplitude ranged between 60 and 80 counts on AQD9467 and AQD5973. Signal was lower, around 50 counts on AQD432.

- Sensor heading on AQD9467 is very stable near 120°. Heading on AQD5973 and AQD432 are much more noisy but probably close to 0° on average; these two sensors also tended to do full clockwise rotations (see periods January, May and July to September 2015).

- Sensor heading and current direction are well correlated and similar for AQD 5973 and AQD 432. AQD 5973 had a vane and, if mounted properly, the offset between the two angles should be near 180 (not 0 as observed here). AQD 9467 was on the compliance section.

- Pitch and roll are all less than 5° or less.

- Pressures (dbar) were 6.3-6.4 (AQD9467), 14.4-14.5 (AQD5973), 19.0-19.1 (AQD432). Corresponding mean depths are 6.26, 14.31 and 18.88 m.

- Temperatures on all 3 Aquadopps look good throughout their respective duration, ranging between 26 and 29 Celsius with annual cycle, include a daily cycle (decreasing with depth), and are close to each other.

Diagnostic data (1 Hz) shows:

- velocity components near-periodic (4-5 seconds) variations (~ 0.5 m s-1).

- heading can vary by ~50° per second on AQD5973 and AQD432. Sensor heading on is also not well correlated between these two instruments; there can be very large discrepancies (despite the sensors being only 5 m apart), which are not related to signal amplitude but rather pressure variations (not shown).

- pitch and roll can go up to 10° but typically remain close to 5°.

- noise level is between 16 and 19 counts.

- signal to noise ratio (SNR) was between 3 and 6 (AQD 9467), 4 and 6 (AQD 5973), 2 and 3.5 (AQD 432). SNR dropped on AQD 9467 and AQD 5973 for some beams during periods mentioned previously.

- standard deviation of 1 Hz current speed and pressure are linearly related (not shown).

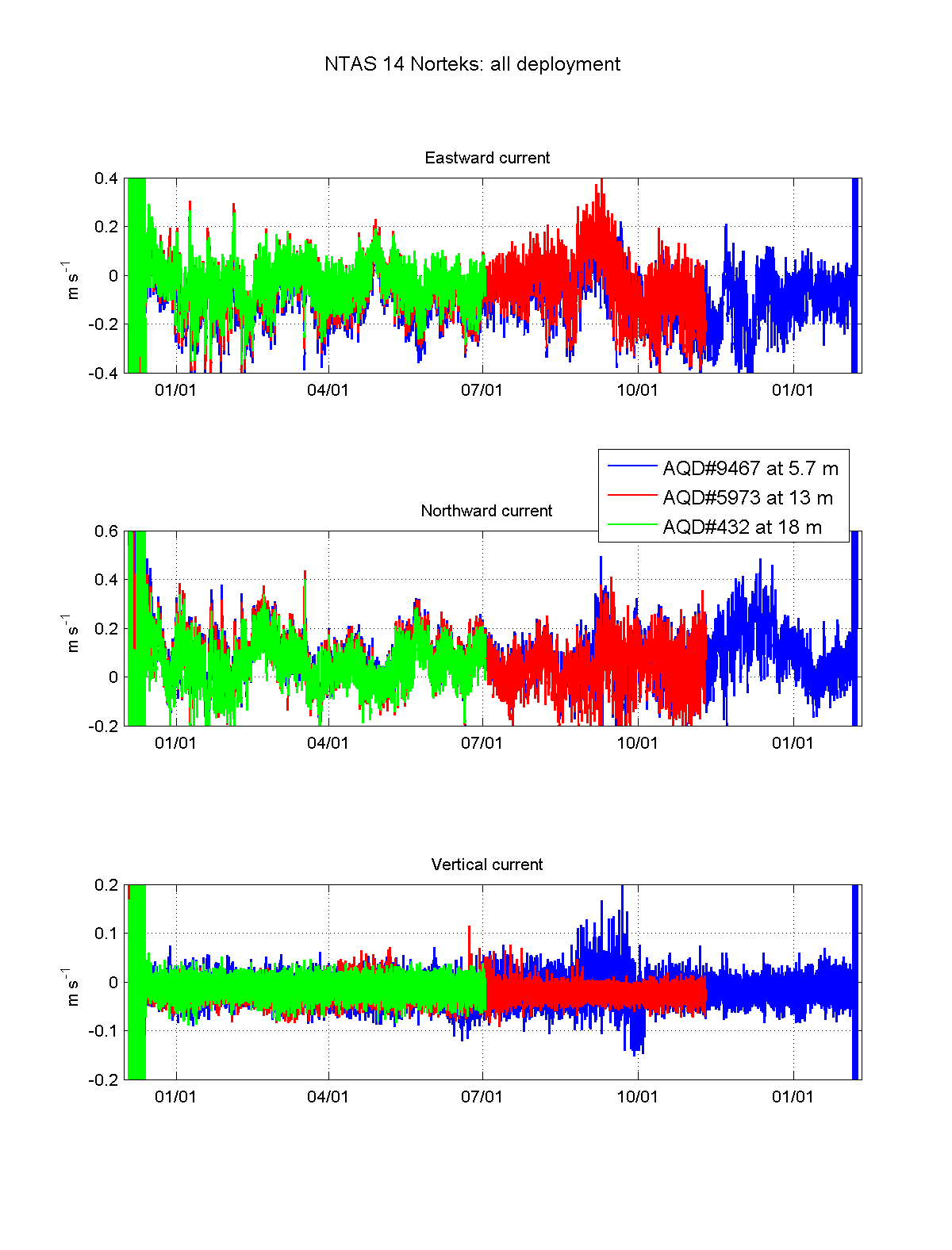


Figure II.1.1. NTAS 14 Aquadopps: velocity components during all deployment.

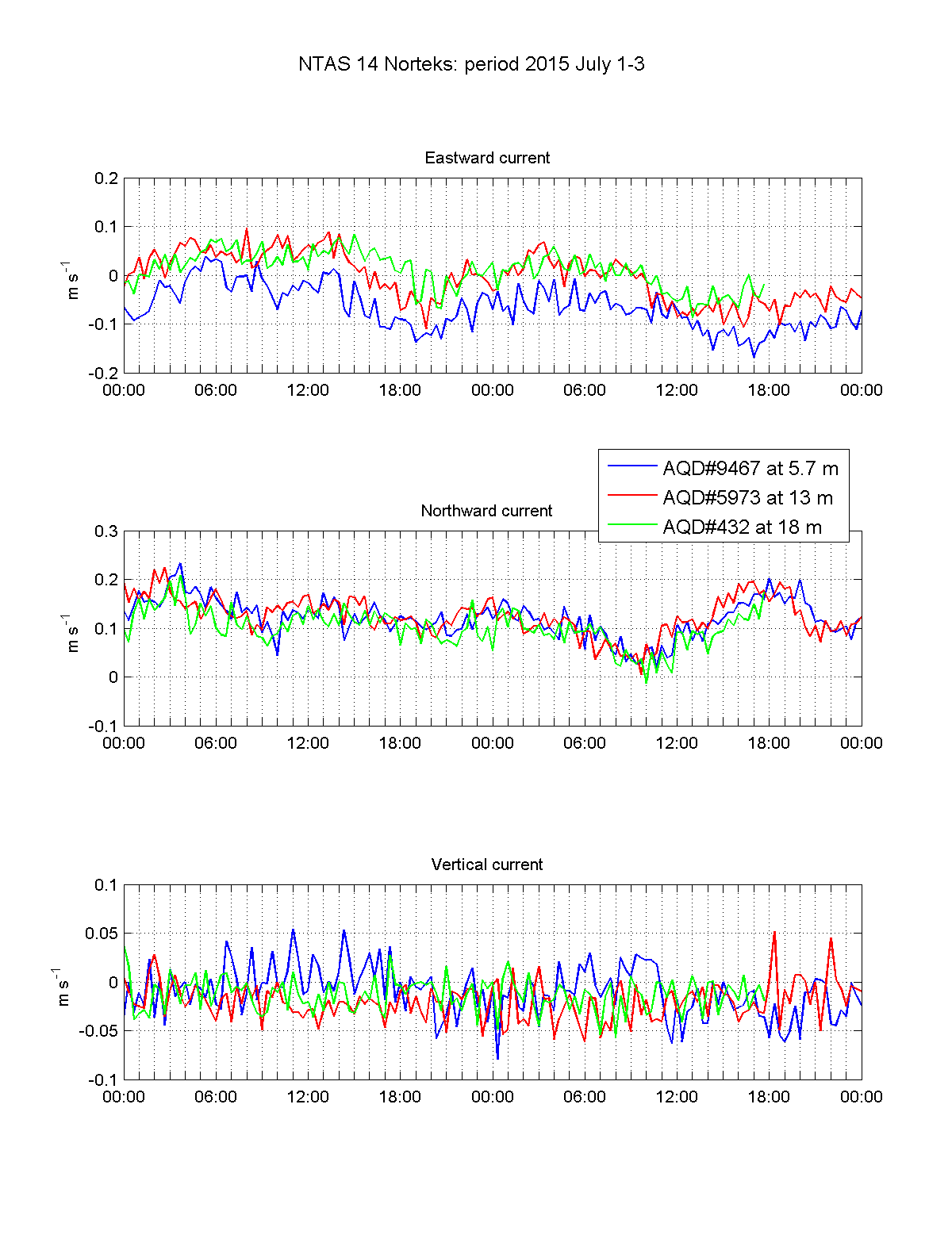


Figure II.1.2. NTAS 14 Aquadopps: velocity components during July 1-3 2015.

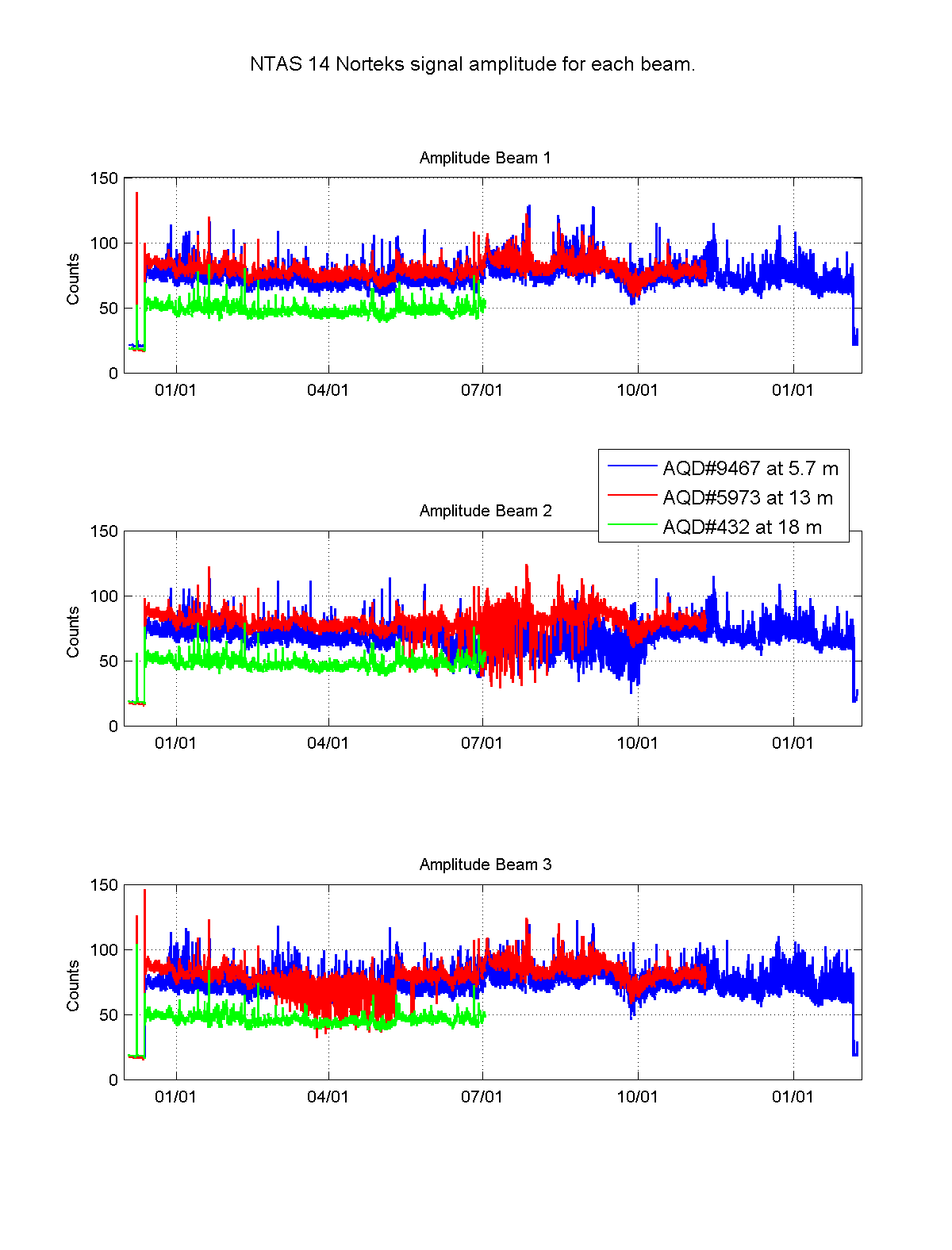


Figure II.1.3. NTAS 14 Aquadopps: signal amplitude during all deployment.

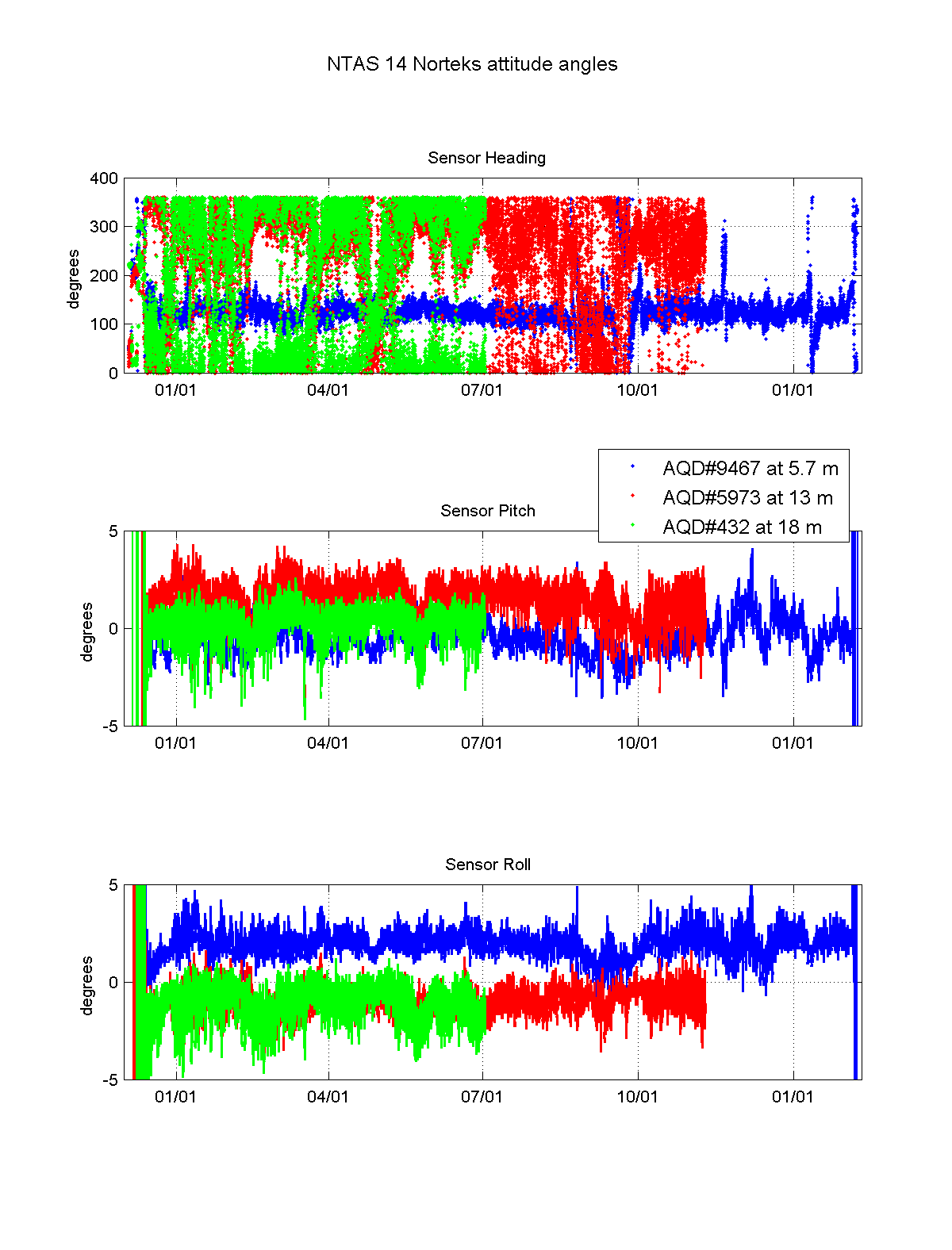


Figure II.1.4. NTAS 14 Aquadopps: sensor attitude during all deployment.

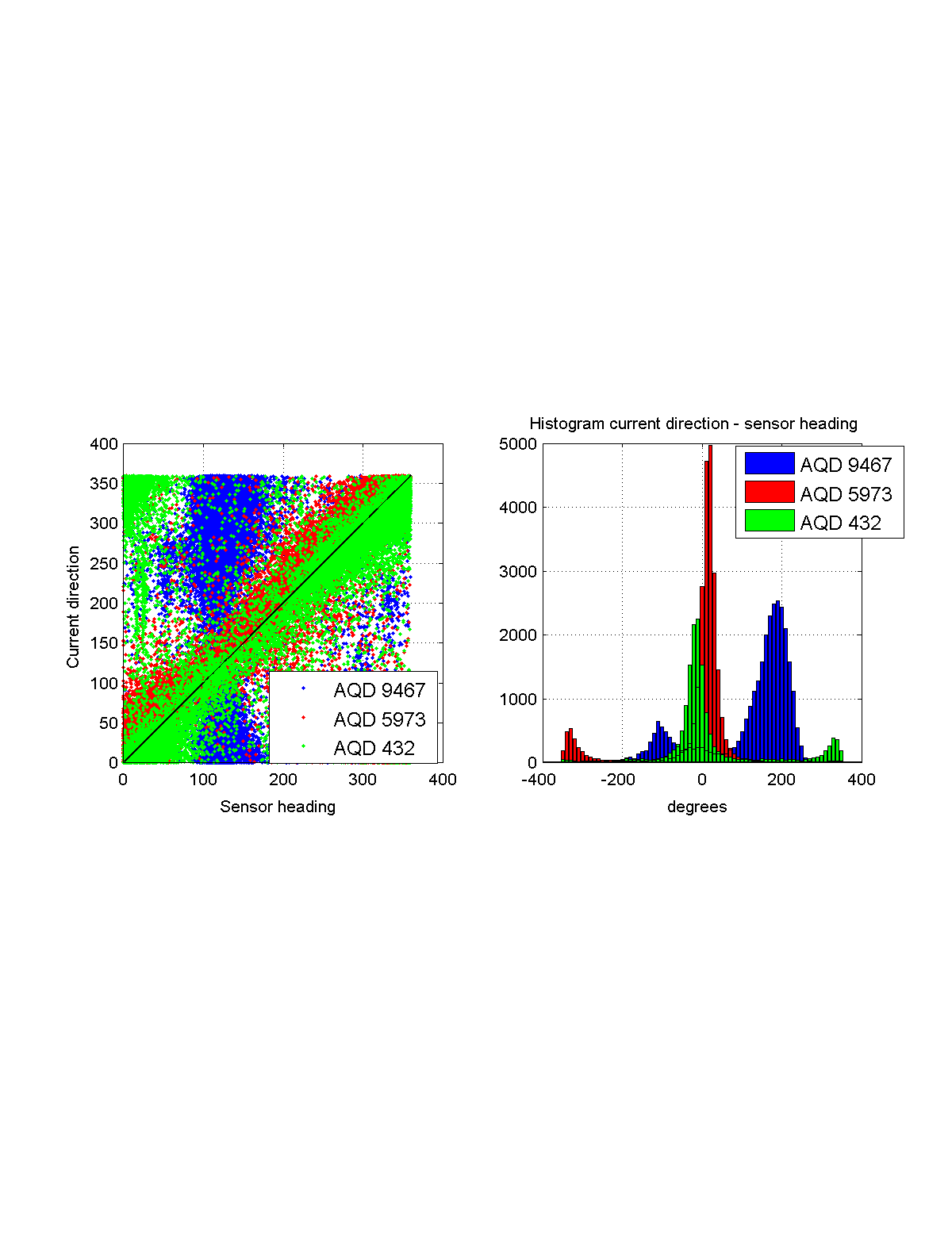


Figure II.1.5. NTAS 14 Aquadopps: sensor heading vs current direction.

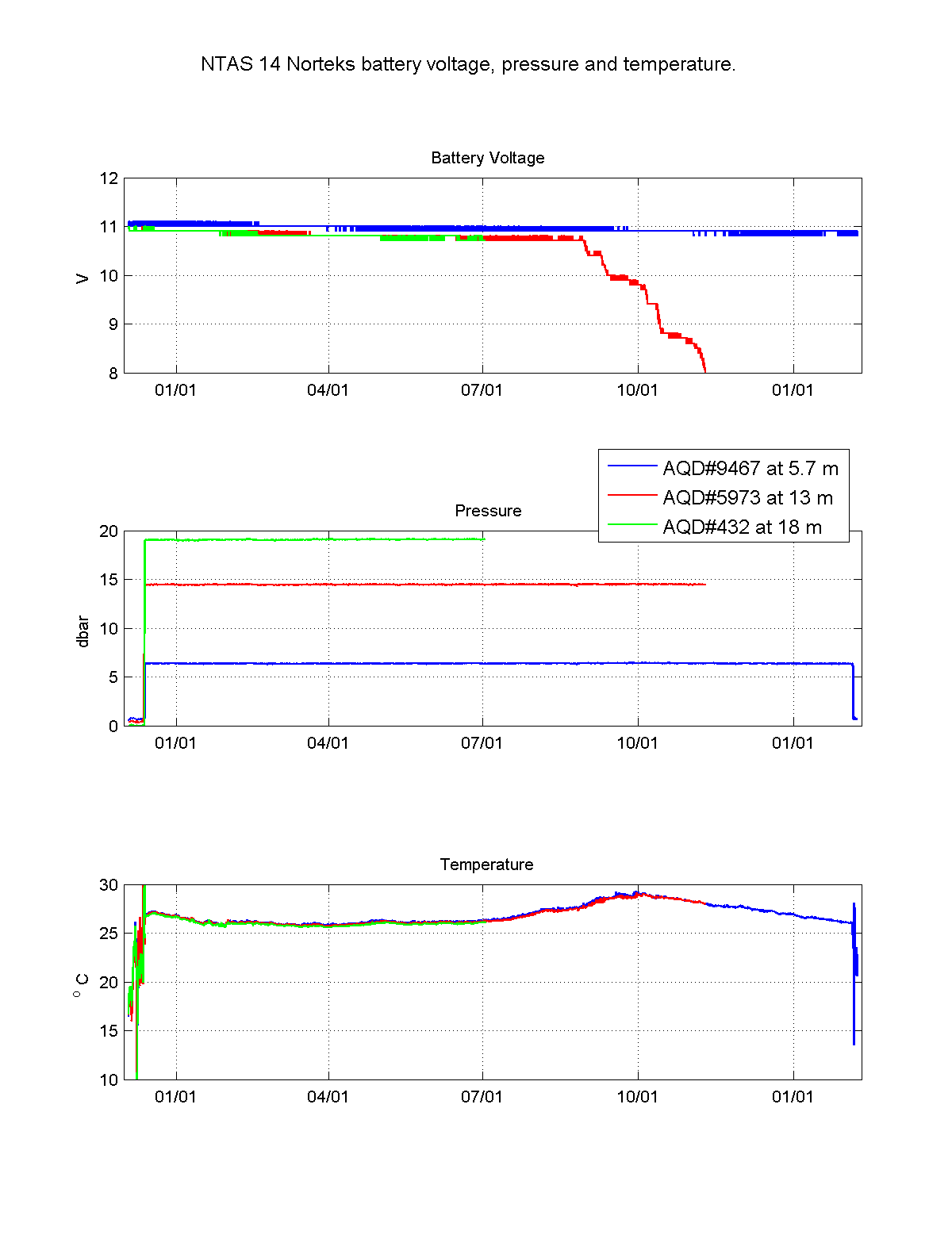


Figure II.1.6. NTAS 14 Aquadopps: battery voltage, temperature and pressure during all deployment.

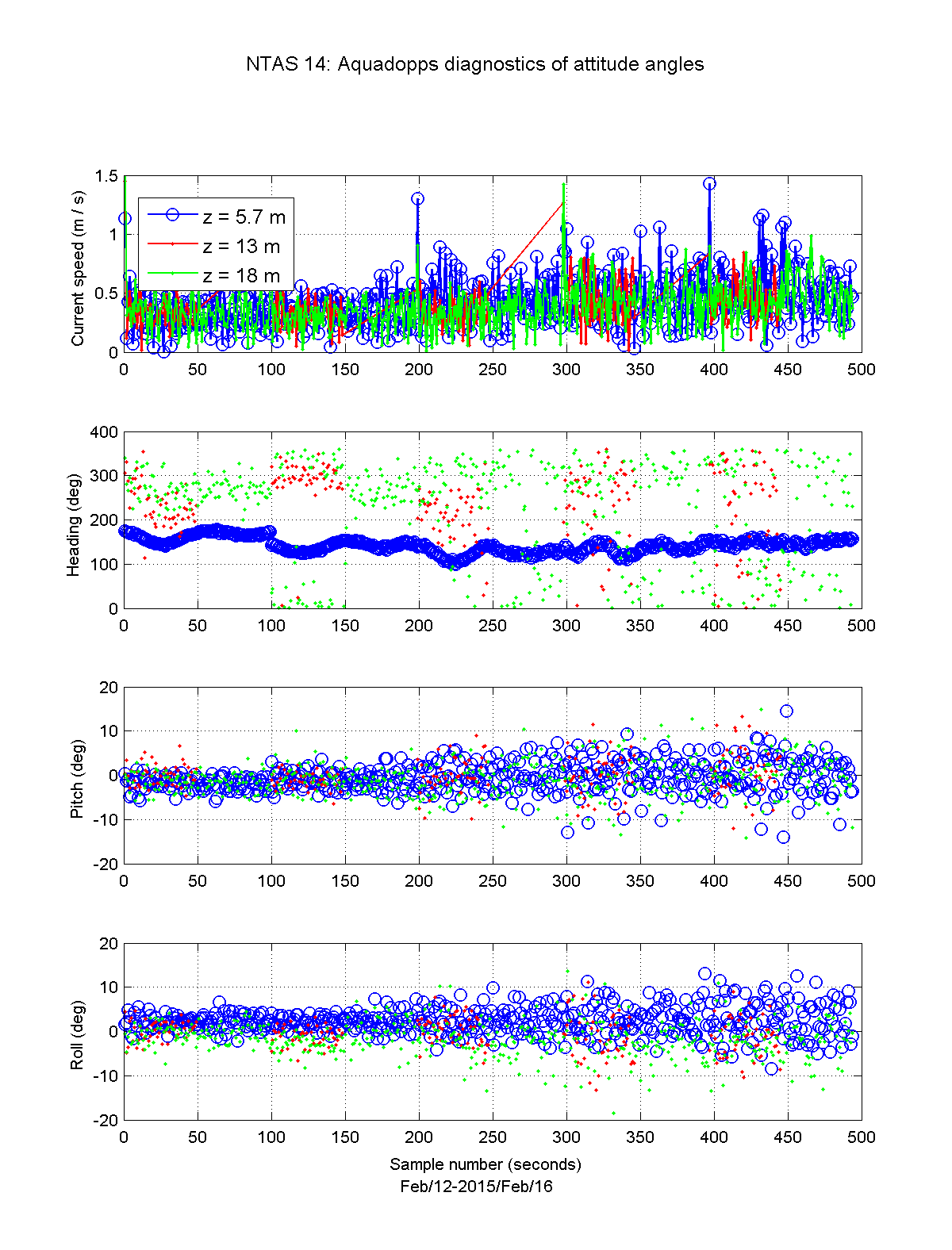


Figure II.1.7. NTAS 14 Aquadopps diagnostic (1 Hz) data: horizontal speed, sensor heading, pitch and roll; period February 12-16 2015.

II.2. RDI

The RDI ADCP was deployed at 85 m depth looking up. Setup was time per ensemble = 1 hour; pings per ensemble = 180; number of cells = 25; cell size 4 m; blanking distance = 3 m.

Table II.2.a. RDI ADCP recovered data from NTAS 14 and data files output from initial processing.

|  |  |  |
| --- | --- | --- |
|  | **RDI 2125** | **Notes** |
| **Depth (m)** | 85 |  |
| **Raw file** | NTA13000.000 | Misleading name may be a carryover from NTAS13 |
| **Output file** | NTAS14\_RDI\_2125\_raw.mat |  |
| **Output file with metadata** | n14\_rdi\_2125.mat |  |

II.2.a. Processing Sequence

From folder *NTAS14\_Recovery/data/rdi*:

i) Edit **do\_adcp.m**: file names for one input file and two output files (same as in Table II.2.a).

ii) In Matlab, run **do\_adcp.m**

Running **do\_adcp.m** does:

- call **ntas14meta.m** to grab mooring metadata.

- runs **upkadcp.m** which load input file and output RDI data.

- adds metadata (in meta variable) such as instrument serial number and depth

- calls **proc\_rdi.m** which renames variables and save into final Matlab output file, along with meta variable.

II.2.a. Data evaluation

**N14\_plot\_rdi.m** plots the RDI data and saves the figures in *NTAS14\_Recovery/Figures/rdi*.

- instrument recorded during the whole duration of the deployment.

- speed data looks realistic for the whole deployment in all bins.

- speed data in bins between 0 and 67 m have a combined percentage of good 3-beams and 4-beams solutions better than 90%.

- Intensity from beams 1 through 3 falls off from ~110-120 counts in first bin to ~ 50-60 counts in bins 68 to 72 m above instrument.

- Correlation in beam 2 is about 10% lower than in other beams.

- the time average error velocity (difference between 2 solutions for vertical velocity) is less than 0.5 mm s-1 from second bin to bin at 55 m.

- ensemble averages of pitch and roll were typically less than 4°. Heading values span all directions but the dominant mode is near 60°. Heading values indicate slow clockwise rotations of the instrument occurring over 1 to 2 days in September 2015.

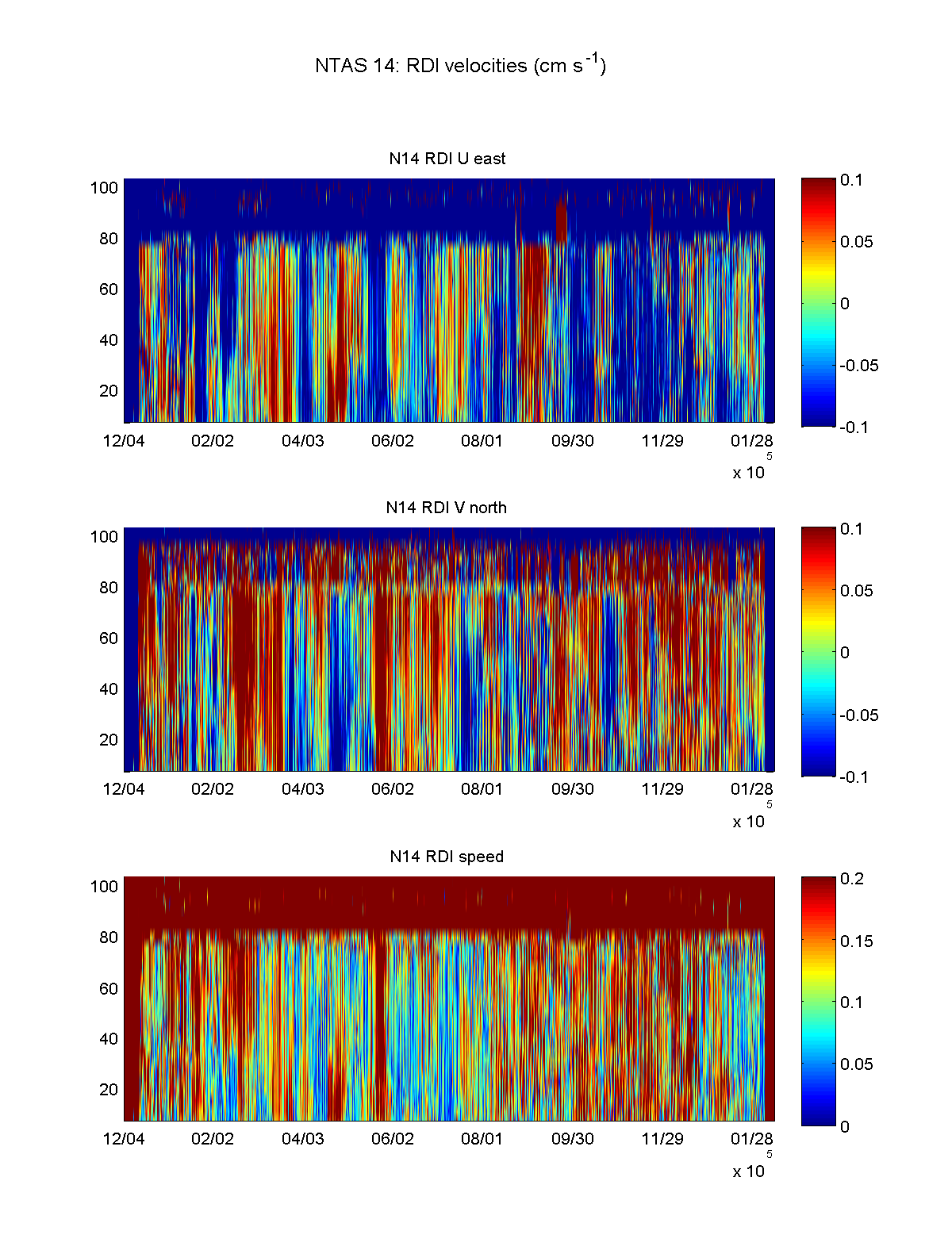


Figure II.2.1. NTAS 14: time-series RDI velocities in cm s-1. X-axis is time(mm/dd) and Y-axis is distance from transducer (m).

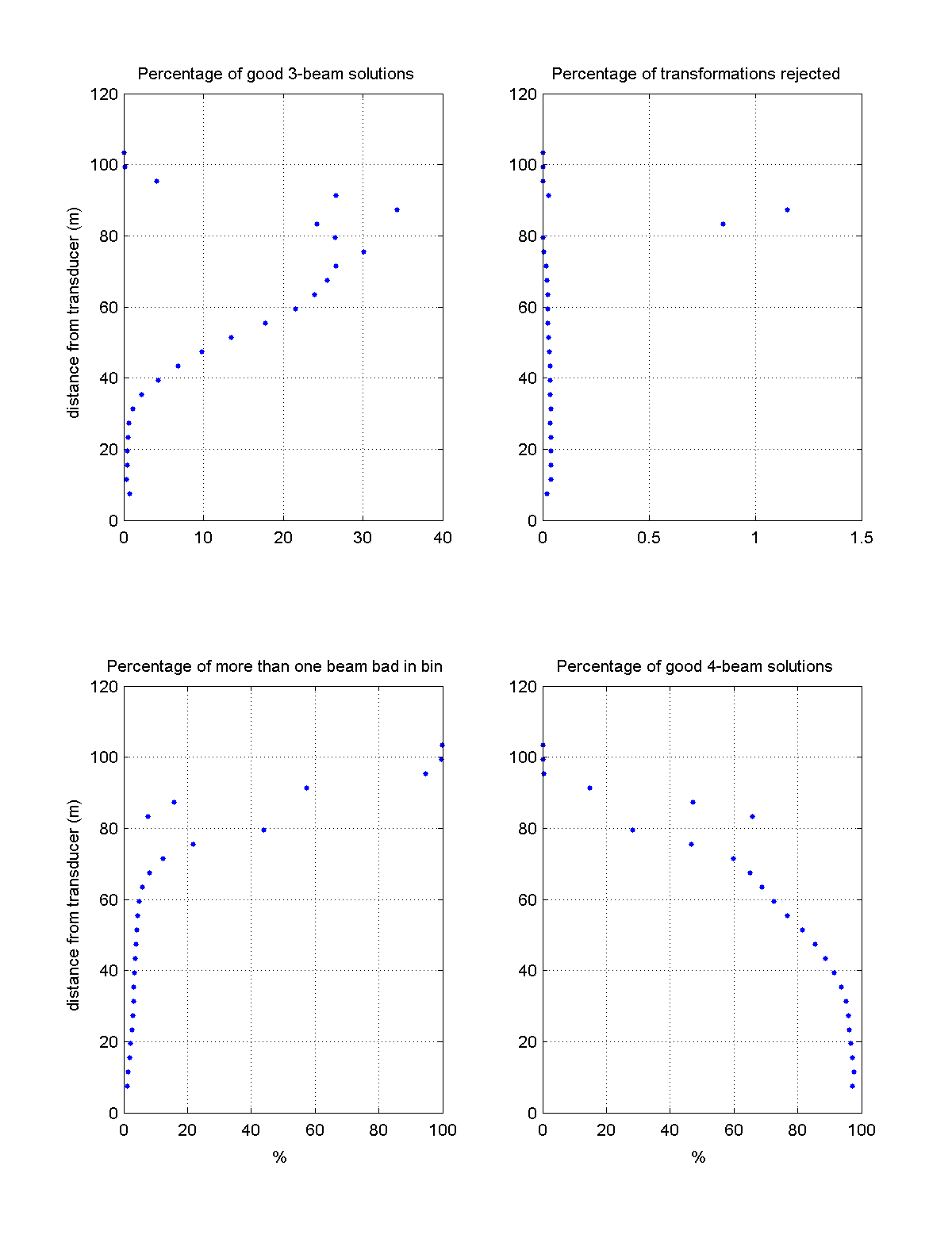


Figure II.2.2. NTAS 14: percentage of good data collected by RDI ADCP.

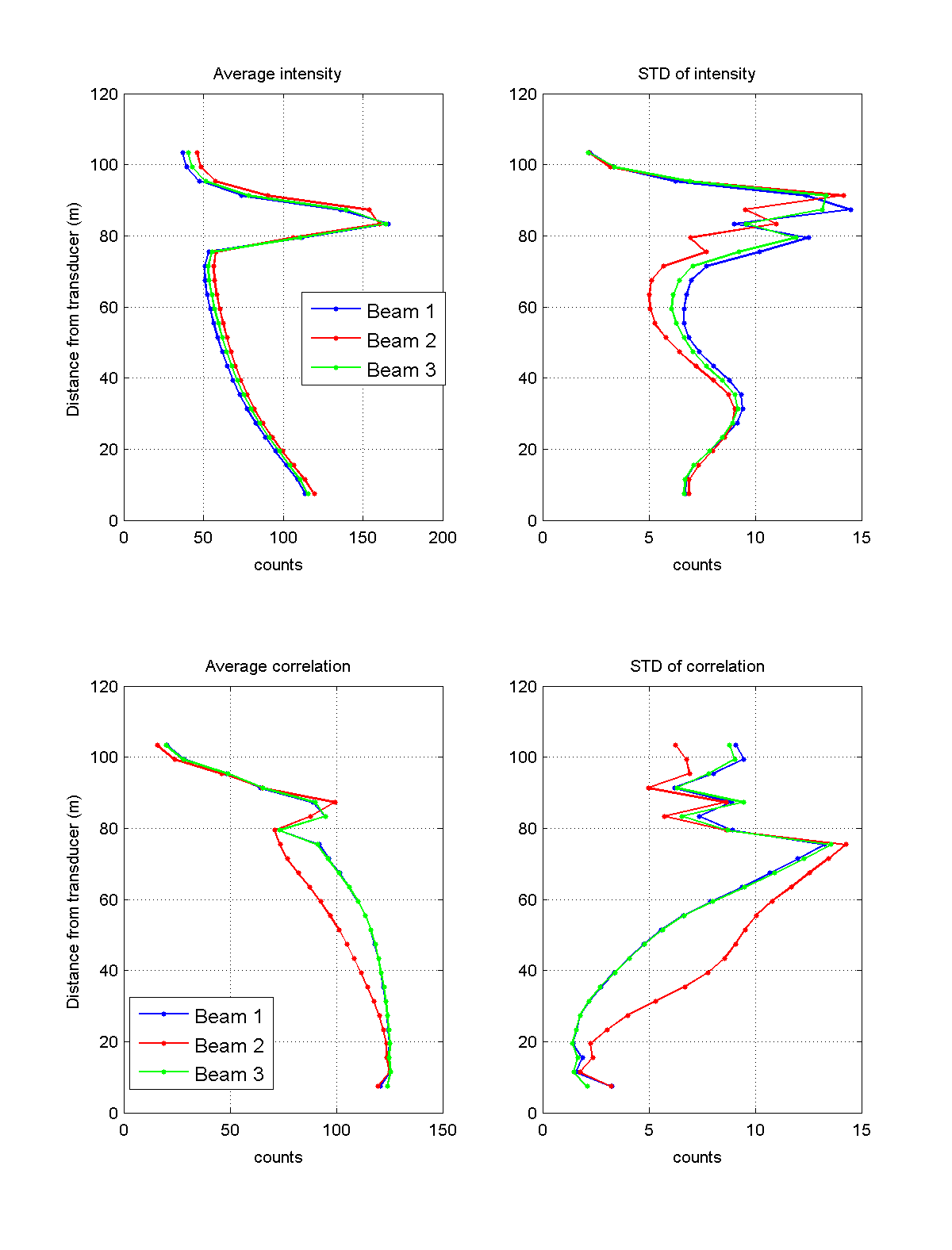


Figure II.2.3. NTAS 14: time average and standard deviation of intensity and correlation for each beam of the RDI ADCP. X-axis is number of counts and Y-axis is distance from transducer (m).

II.3. Seabird 39

There is currently one data file missing (SBE39#631 as data could not be downloaded from this instrument).

Data file from SBE39#539 is missing about last 2 weeks of data (January/February 2016) and instead has some erroneous time stamps at the end (year 2116).

Data from SBE39#4465 was initially saved with erroneous name (sbe37IM\_0684\_ntas14.asc). The name was changed to sbe39\_4465\_ntas14.asc. Meanwhile, sbe37IM\_0684\_ntas14a.asc which contains data from SBE37IM#684 was relabeled sbe37\_0684\_ntas14.asc.

Back at WHOI, data was downloaded from SBE39#4465 a second time under the name 20160317\_n14\_sbe39-4465\_data\_EGH.asc. This file was renamed sbe39\_4465\_ntas14.asc (other one was first saved in folder “old”) and is the file that was used for processing: data was the same in both files but header was cleaner in last download.

II.3.a. Data processing

Some of the original file names were edited (removing “sn” from name) so that all file names had same format (sbe39\_XXXX\_ntas14.asc, where XXXX is 4-digits string containing serial number, pre-padded with 0 if necessary). Processing sequence was:

i) In Matlab, run **dosbe39s.m** in *NTAS14\_Recovery/data/sbe39*

ii) In Matlab, run **plotsbe39s.m**

Step i) calls **ntas14meta.m**, then **get\_ss\_info.m** to grab information (serial numbers, depths and type of instrument) in **ntas14\_ss\_insts.csv**. It then goes through all available SBE39 data files (sbe39\_XXXX\_ntas14.asc) and reads them using **get\_sbe\_39.m** on these ascii files. Meta data and data are combined and saved in Matlab files (ntas14\_sbe39\_XXXX.mat). SBE39AT is also part of this processing.

Step ii) opens mat files created in previous step and plots time-series of all SBE39s temperatures, with offset added on each successive sensor record to avoid overlapping curves. SBE39AT is also plotted. The plot is saved in the folder *../../Figures/sbe39*.

CHANGES MADE ON 7/22/2016:

a) changed depth value for SBE39#539 to 4.2 m (down from 5 m) in ntas14\_ss\_insts.csv

b) updated recovery Excel spreadsheet with new depth for SBE39#539. New file named NTAS14\_Recovery20160722.xlsx

c) changed value in depth variable in Matlab file ntas14\_sbe39\_0539.mat and saved manually

d) modified plotsbe39s.m to add decimal digit in legend and rerun to create updated figure reflecting new depth for SBE39#539.

e) updated this document.

f) all mentioned changes saved on Buoy2 data server.

II.3.b. Data evaluation

All data records available (SBE39s #631) look realistic. SBE39#539 record ends about 2 weeks before actual time of recovery.

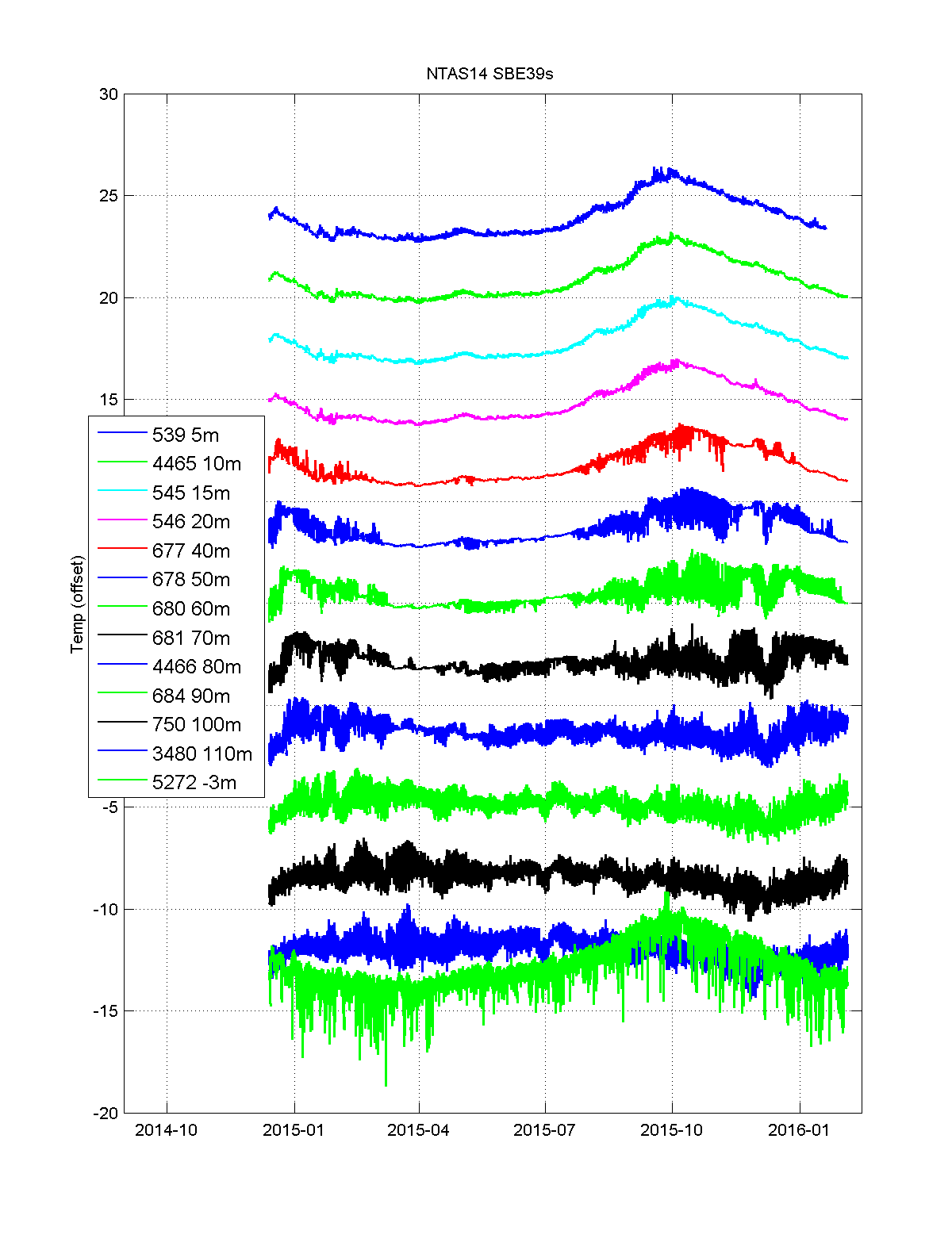


Figure II.3.1. NTAS 14 SBE 39s temperature records. Records from SBE39s #631 are missing.

II.4. Seabird 37

All 7 sbe37s (3 IMs, 2 SSTs, 2 bottom sensors) collected data for the whole duration of NTAS 14. Data file names are listed in Table II.4.1.

Table II.4.1. Filenames for initial data processing of NTAS 14 SBE37s.

|  |  |  |  |
| --- | --- | --- | --- |
| **SN** | **Original file name** | **Renamed input file** | **Output file** |
| 669 | sbe37IM\_sn0669\_ntas14.asc | sbe37\_669\_ntas14.asc | ntas14\_sbe37\_669.mat |
| 684 | sbe37IM\_sn0684\_ntas14a.asc | sbe37\_684\_ntas14.asc | ntas14\_sbe37\_684.mat |
| 686 | sbe37IM\_sn0686\_ntas14.asc | sbe37\_686\_ntas14.asc | ntas14\_sbe37\_686.mat |
| 1836 | sbe37SST\_sn1836\_ntas14.asc | sbe37\_1836\_ntas14.asc | ntas14\_sbe37\_1836.mat |
| 3605 | sbe37SST\_sn3605\_ntas14.asc | sbe37\_3605\_ntas14.asc | ntas14\_sbe37\_3605.mat |
| 11392 | SBE37SM-RS232\_03711392\_2016\_02\_06.hex | sbe37\_11392\_ntas14.cnv | ntas14\_sbe37\_11392.mat |
| 11393 | SBE37SM-RS232\_03711393\_2016\_02\_06.hex | sbe37\_11393\_ntas14.cnv | ntas14\_sbe37\_11393.mat |

II.4.1 Data processing

-sbe37IM\_sn0669\_ntas14.asc was edited by removing 2 lines (containing garbled entries) between the \***END\*** header statement and the data. New file saved as sbe37\_669\_ntas14.asc.

-sbe37IM\_sn0684\_ntas14.asc had wrong name as it contained data from SBE39IM#4465. It was renamed sbe39\_4465\_ntas14.asc and moved into folder sbe39.

- sbe37IM\_sn0684\_ntas14a.asc was renamed sbe37\_684\_ntas14.asc. Two lines with garbled entries were also removed between \***END**\* statement and data stream.

- sbe37IM\_sn0686\_ntas14.asc was renamed sbe37\_686\_ntas14.asc.

sbe37SST\_sn1836\_ntas14.asc was renamed sbe37\_1836\_ntas14.asc.

sbe37SST\_sn3605\_ntas14.asc was renamed sbe37\_3605\_ntas14.asc.

-SBE37SM-RS232\_03711392\_2016\_02\_06.cnv was renamed sbe37\_11392\_ntas14.cnv.

-SBE37SM-RS232\_03711393\_2016\_02\_06.cnv was renamed sbe37\_11393\_ntas14.cnv

The 7 data files from the SBE37s were processed using these steps in folder *NTAS14\_Recovery/data/sbe37*:

i) In Matlab, run **dosbe37s.m**

ii) In Matlab, run **plotsbe37s.m**

Step i) calls **ntas14meta.m** to set meta data for the NTAS 14 mooring, it then calls the function **get\_ss\_info.m** with argument file **ntas14\_ss\_insts.csv** to get the list of sensors types, serial numbers and depths. For each sensor, it then looks for the input data file (filename format is sbe37\_XXXXX\_ntas14.YYY; Xs denote the serial number which can be 3 to 5 digits long; YYY is extension type which either “asc” or “cnv”). Depending on extension of the file, either **get\_sbe\_37.m** (“.asc” files) or **get\_sbe37\_cnv.m** (“.cnv” files) is called to read the input file and the data it contains. The data and metadata are then saved in Matlab files (file name format is ntas14\_sbe37\_XXXXX.mat). Step ii) plots the data from the .mat files as time-series and save the figures in the folder *../../Figures/sbe37*.

II.4.2 Data evaluation

Time-series of temperature and salinity from the sbe37 dataset are shown in figures II.4.1 and II.4.2. Salinity from SBE37#1836 dropped ~4 psu, for a few hours on two occasions (2015/6/7 and 2015/8/18-19).

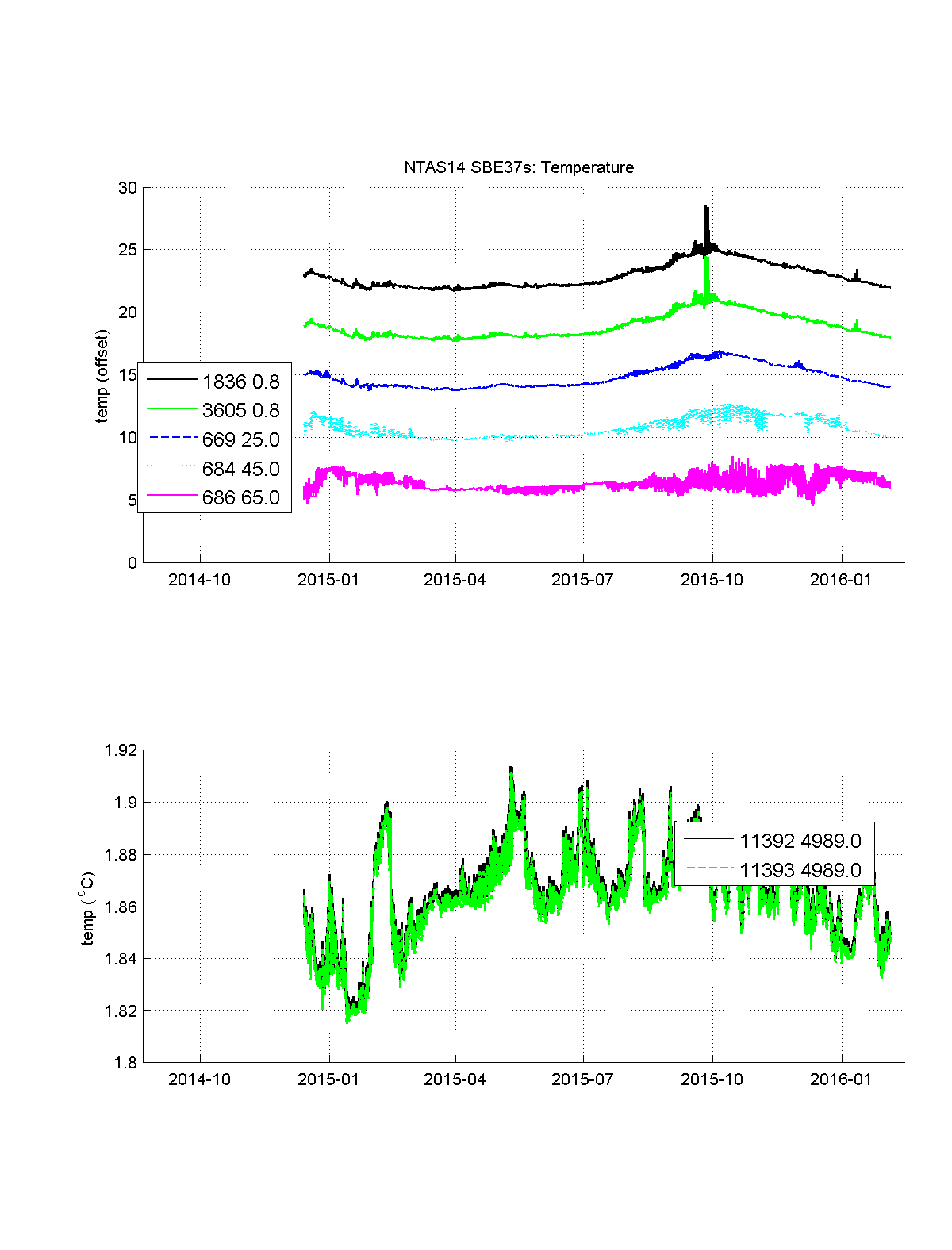


Figure II.4.1. NTAS 14 SBE 37s: temperature record, with offset added for readability. Serial numbers and corresponding depths are shown in legend.

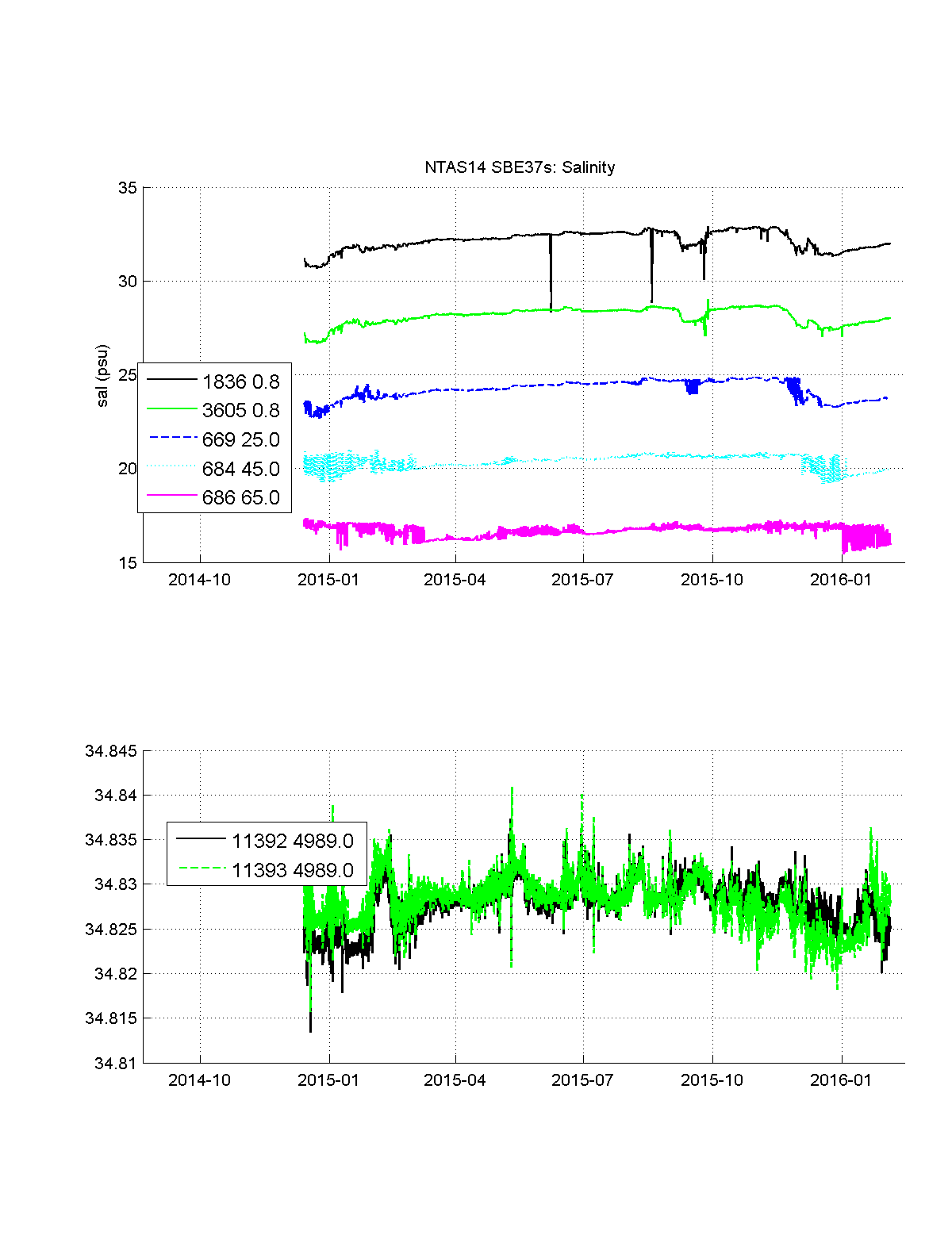


Figure II.4.2. NTAS 14 SBE 37s: salinity record, with offset added for readability.

Serial numbers and corresponding depths are shown in legend.

II.5. Starmon ODDI sensors

There were 6 Starmon ODDI temperature sensors deployed on the mooring line of NTAS 14. All sensors recorded for the whole NTAS14 duration.

Table II.5.1. Names of Starmon ODDIs data files downloaded from instruments, edited in shell and used for input in initial data processing, and finally output in Matlab format.

|  |  |  |  |
| --- | --- | --- | --- |
| **SN** | **Original file name** | **Renamed input file** | **Output file** |
| 3167 | 8T3167.DAT | oddi\_T3167.DAT | ntas14\_starmon\_3167.mat |
| 3168 | 9T3168.DAT | oddi\_T3168.DAT | ntas14\_starmon\_3168.mat |
| 3169 | 7T3169.DAT | oddi\_T3169.DAT | ntas14\_starmon\_3169.mat |
| 3170 | 7T3170.DAT | oddi\_T3170.DAT | ntas14\_starmon\_3170.mat |
| 3171 | 7T3171.DAT | oddi\_T3171.DAT | ntas14\_starmon\_3171.mat |
| 3791 | 5T3791.DAT | oddi\_T3791.DAT | ntas14\_starmon\_3791.mat |

II.5.1.Data processing

The files are located in the folder *NTAS14\_Recovery/data/starmon*. Table II.5.1 lists the file names containing data as downloaded from the recovered instruments and used for input in the initial data processing; the table also lists the output file names in Matlab.

i) The filenames and their contents were batch edited using the shell script **oddiname\_edit.sh**. Original .DAT file names are changed to names with format “oddi\_TXXXX.DAT), as listed in third column of Table II.5.1.

ii) In Matlab, run **load\_raw\_starmons.m**.

iii) in Matlab, run **plot\_starmons.m**.

In step i), the raw file data files downloaded from the instruments are renamed (see Table II.5.1) and the data format edited (replace comment sign from # to % to match Matlab’s style, replace periods and columns in date and time stamps by spaces, replace commas by periods in decimal numbers). This produces the oddi\_TXXXX.DAT files which are used as input in step ii) in the Matlab script **load\_raw\_starmons.m** which calls **ntas14meta.m** to get the meta data for NTAS 14, uses **get\_ss\_info.m** to grab SN and depth for sensors listed in **ntas14\_ss\_insts.csv**. The data and metadata are combined for each sensor and saved in files with name format ntas14\_oddi\_XXXX.mat (XXXX is 4-digits serial number). In step iii), **plot\_starmons.m** plots the time-series of temperature for each Starmon ODDI sensor, adds an offset for readability and saves the figure in *../../Figures/starmon*.

II.5.2. Data evaluation

All sensors recorded for the whole NTAS14 duration.

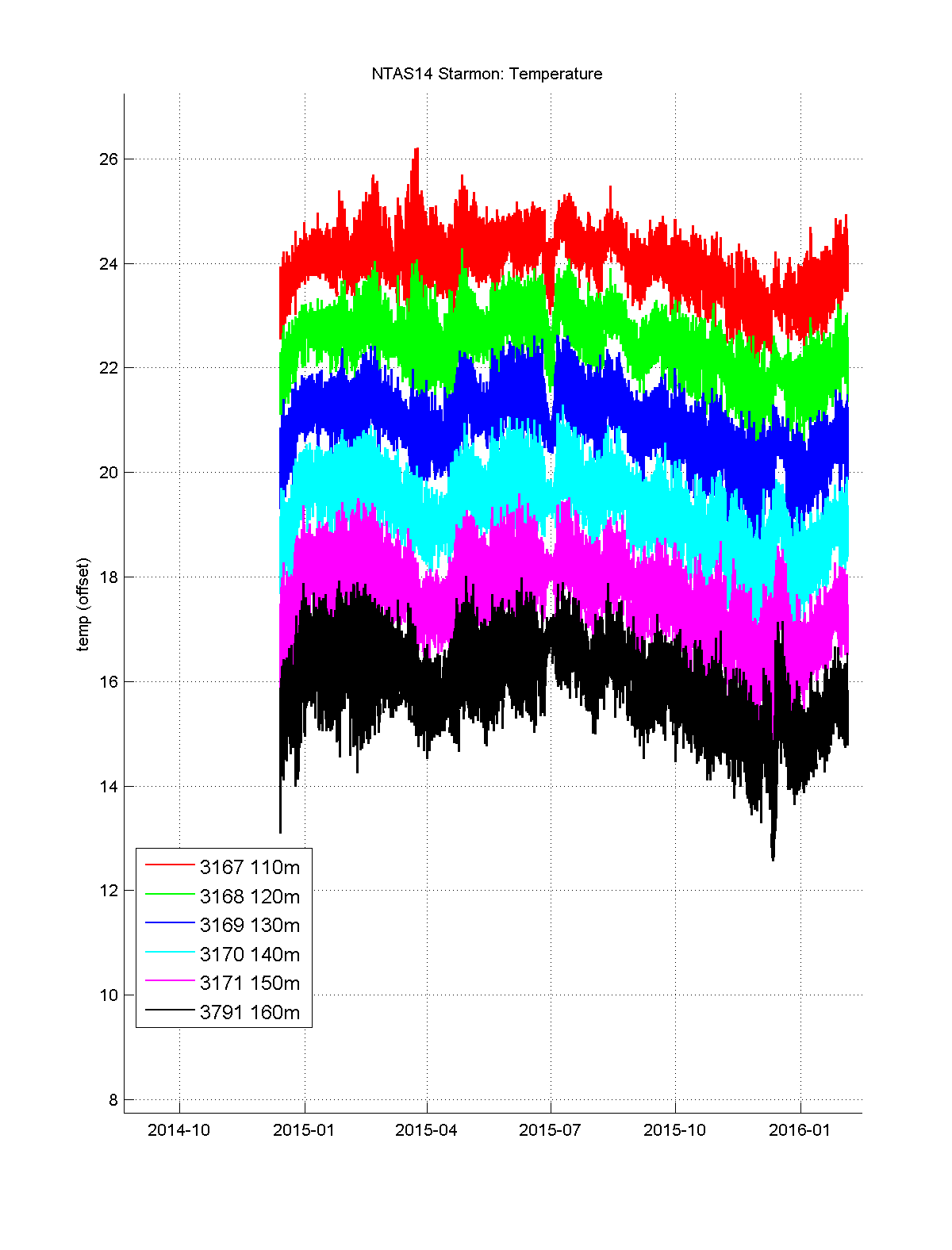


Figure II.5.1. NTAS 14: time-series of Starmon ODDIs temperature records with offset added for readability.

II.6. Brancker SOLO sensors.

There were 4 SOLO temperature sensors deployed on NTAS 14 mooring line. Two of these sensors have an incomplete record and stopped before the end of the deployment. Data files (raw and Matlab) are listed in Table II.6.1.

Table II.6.1. Data file names for RBR SOLO sensors recovered from NTAS 14.

|  |  |  |
| --- | --- | --- |
| **SN** | **Input raw file** | **Output Matlab file** |
| 75556 | 075556\_20160210\_2327.rsk | ntas14\_solo\_75556.mat |
| 75557 | 075556\_20160210\_2242.rsk | ntas14\_solo\_75557.mat |
| 75558 | 075556\_20160210\_2307.rsk | ntas14\_solo\_75558.mat |
| 75559 | 075556\_20160210\_2318.rsk | ntas14\_solo\_75559.mat |

II.6.1. Data processing

Data files are located in folder *NTAS14\_Recovery/data/solo*.

i) In Matlab, run **dosolos.m**

ii) in Matlab, run **plot\_solos.m**

Step i) runs **ntas14meta.m** to create meta data for NTAS 14, then calls **get\_ss\_info.m** to collect sensor information (serial number and depth) in the list in **ntas14\_ss\_insts.csv**. It then reads data file (0XXXXX\_20160210\_hhhh.rsk, where XXXXX is 5-digits serial number and hhhh is time of download from instrument on 2016/2/10) for each sensor, using Matlab scripts located in folder RSKtools and called **RSKopen.m** and **RSKreaddata.m**. Once the data id read, it is combined with meta data and saved into a .mat file, as listed in Table II.6.1. In step ii), the .mat files produced in previous step are downloaded and the temperature time-series are plotted in the same graph with different color for each sensor. Resulting figure is saved in *../../Figures/solo*.

II.6.2 Data evaluation

Last valid data point was on 2015/5/11 06:55 UTC for SOLO 75557 (note that a few isolated data points were recorded after that date), and 2015/12/13 06:34 UTC on SOLO 75558. All SOLOs had a few short gaps (1 sample missing) spread out in their datasets.

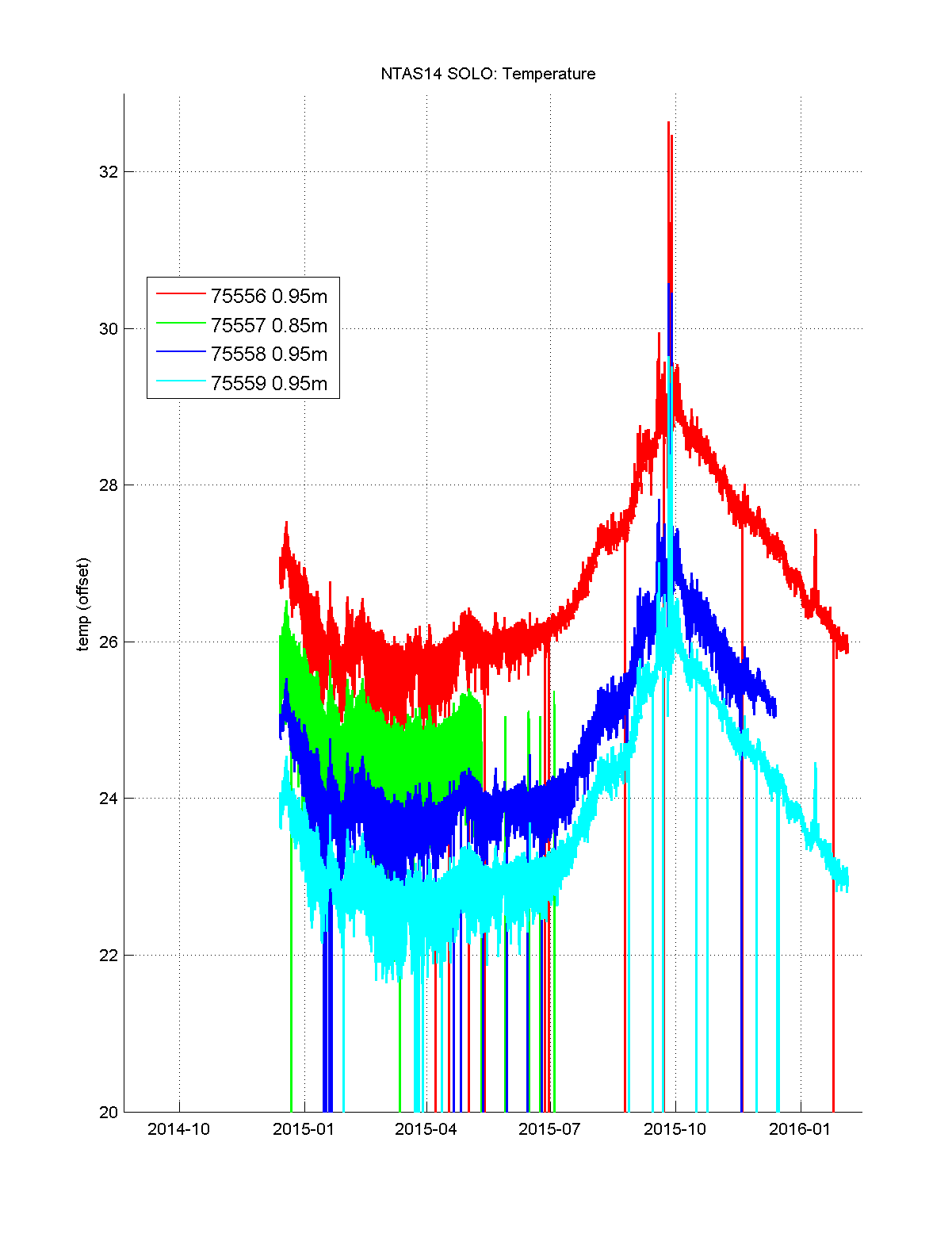


Figure II.6.1. NTAS 14 SOLO temperature time-series, with offset added for readability.

II.7. RBR (Brancker) DUO

One DUO sensor was deployed on NTAS 14 mooring that measured temperature and conductivity. The sensor recorded for the whole duration of the deployment. There are some isolated outliers in conductivity (jumps from one sample to the next), especially around May 2015. Data files are listed in Table II.7.1.

Table II.7.1. Data file for DUO sensor recovered from NTAS 14.

|  |  |  |
| --- | --- | --- |
| **SN** | **Raw input file** | **Matlab output file** |
| 61568 | 061568\_20160309\_1630.rsk | ntas14\_duo\_62568.mat |

II.7.1 Data processing

Data files listed in Table II.7.1 are located in folder *NTAS14\_Recovery/data/duo*.

i) in Matlab run **doduos.m**.

ii) in Matlab, run **plot\_duos.m**.

Step i) runs **ntas14meta.m** to create meta data for NTAS 14, then calls **get\_ss\_info.m** to collect sensor information (serial number and depth) in the list in **ntas14\_ss\_insts.csv**. It then reads data file 061568\_20160309\_1630.rsk using Matlab scripts located in folder RSKtools and called **RSKopen.m** and **RSKreaddata.m**. Once the data is read, it is combined with meta data and saved into a .mat file, as listed in Table II.7.1. In step ii), the .mat file produced in previous step is downloaded and the temperature and salinity time-series are plotted and graphics are saved in *../../Figures/duo*.

II.7.2 Data evaluation

DUO sensor recorded data for the whole deployment. Salinity had a few short-lived jumps in salinity around in early May 2015.

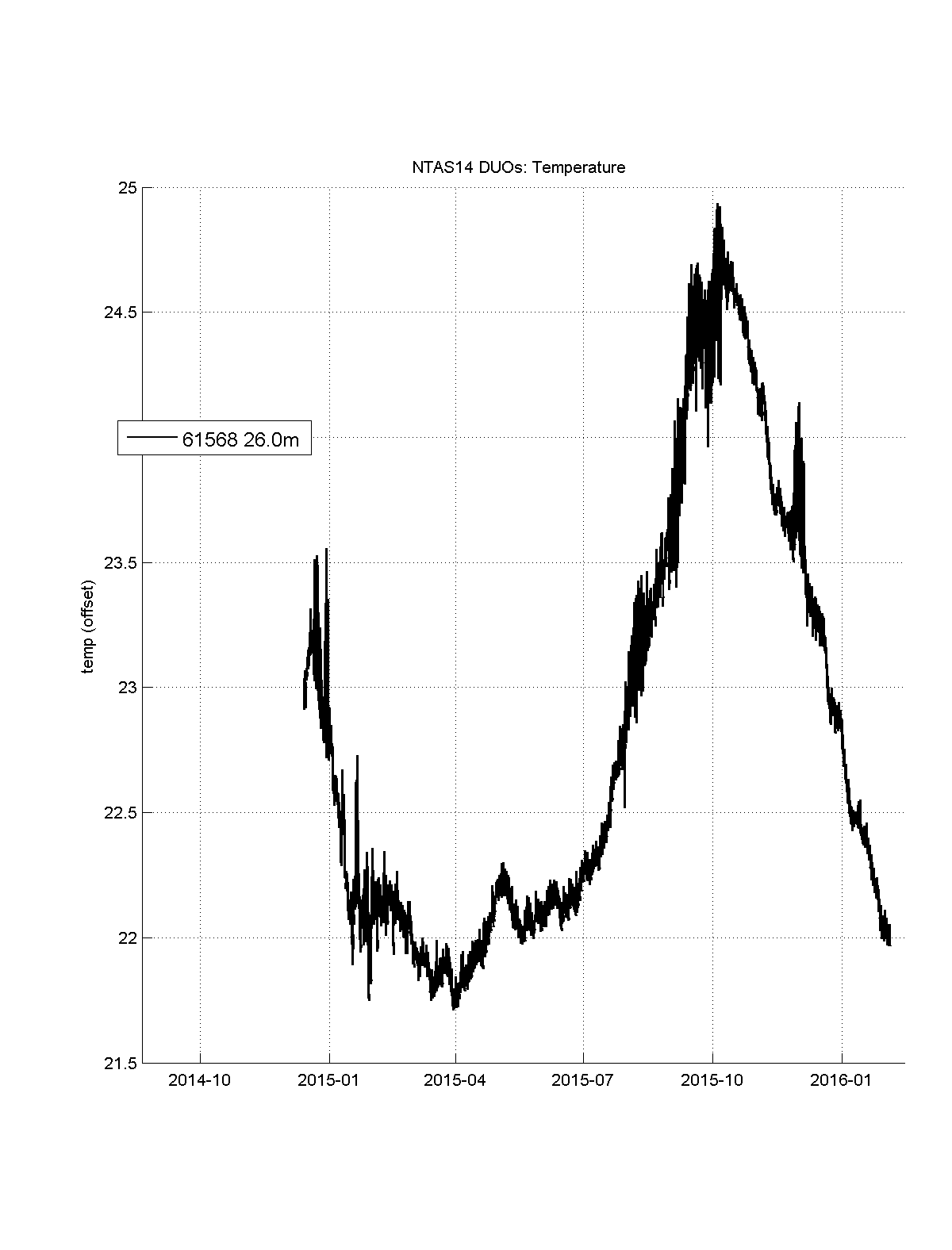


Figure II.7.1. NTAS 14: RBR DUO temperature record.

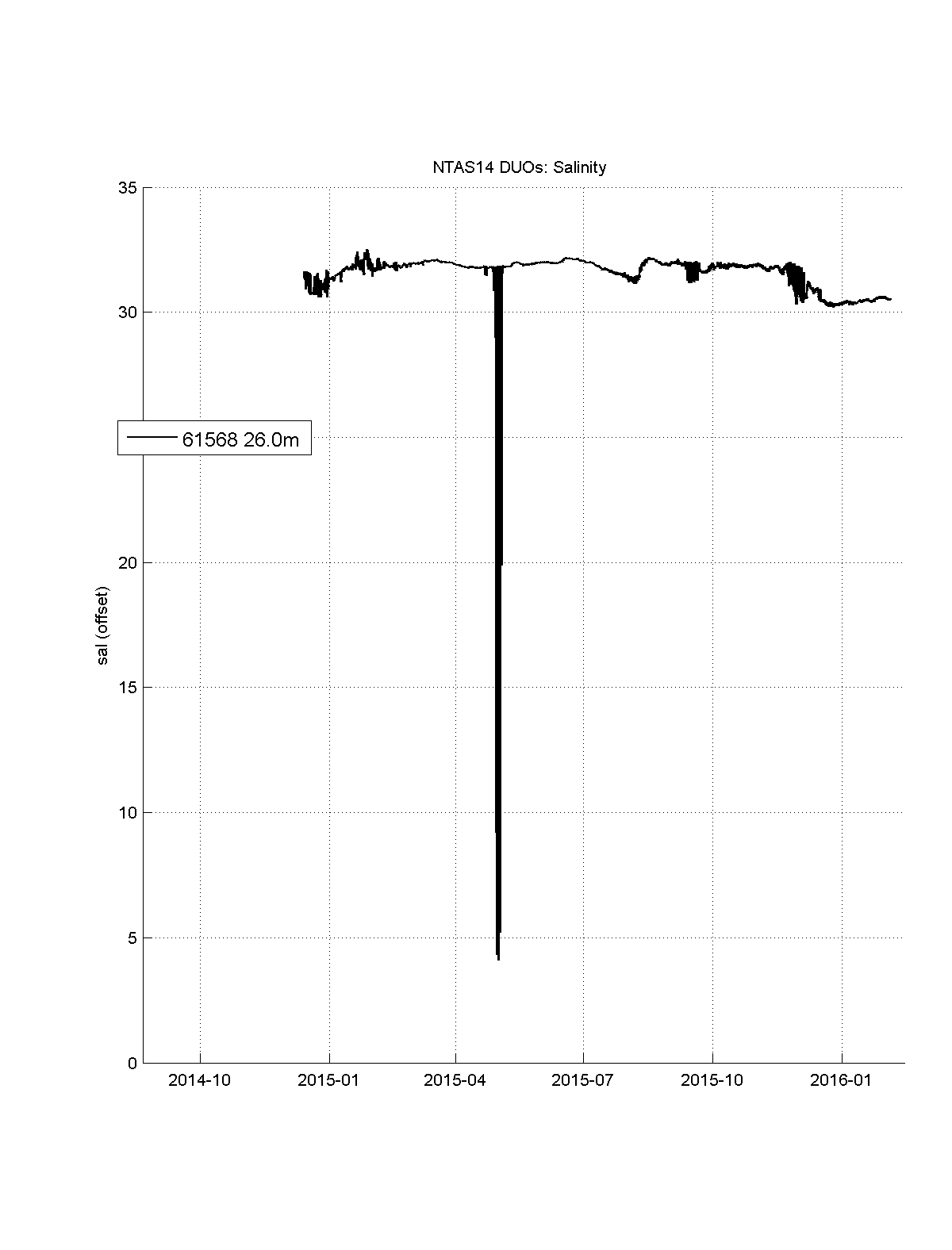


Figure II.7.2. NTAS 14: RBR DUO salinity record.

III. Surface sensors

III.1 ASIMET loggers

Loggers 16 and 12 operated for the whole duration of NTAS 14 deployment. BPR data from logger 16 failed prematurely in December 2015 and had many missing data and spikes in the summer and fall 2015. Module data (BPR#217) is more complete although it also misses the end of the deployment. A leak on this BPR sensor was detected at recovery. PRC#214 on logger 16 did not record good data during second half of year 2015.

Data processing sequence:

i) in Matlab, run **do\_imets.m**

ii) in Matlab, run **plot1logger.m**

Step i) calls **ntas14meta.m** which creates meta data for NTAS 14, reads raw data files using **imet\_bin\_to\_asc.m**. It then calls **get\_mod\_meta.m** with **N14modhts.csv** which lists ASIMET modules serial number and heights. Finally it calls **imet\_asc\_to\_mat.m** to save the data into Matlab .mat files in the folder *NTAS14\_Recovery/data/asimet*, as listed in Table III.1.1.

Table III.1.1. File names for data in ASIMET loggers recovered from NTAS 14.

|  |  |  |
| --- | --- | --- |
| **SN** | **Raw file** | **Matlab output file** |
| 16 | L16.DAT | nt14\_lgr16.mat |
| 12 | L12.DAT | nt14\_lgr12.mat |

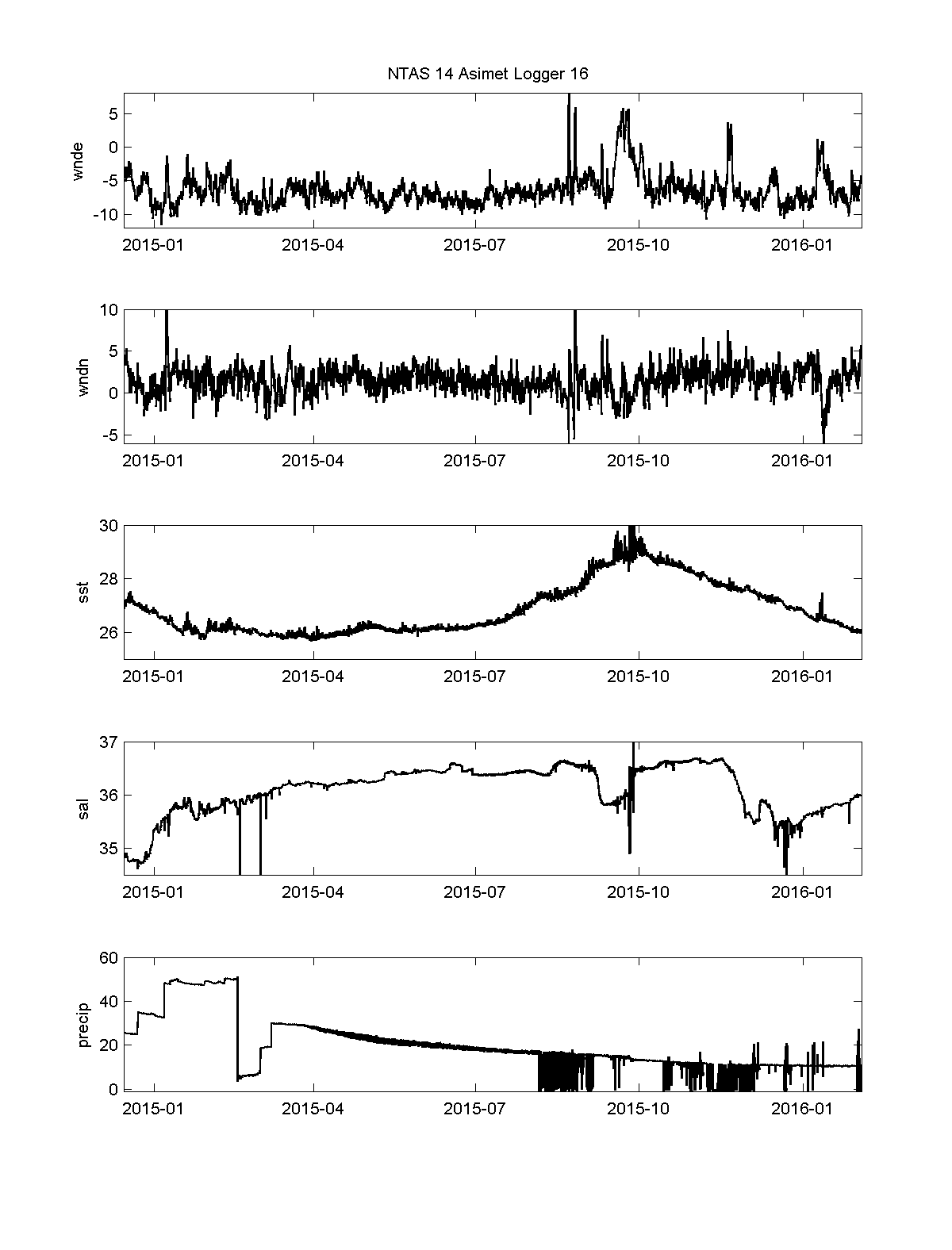


Figure III.1.1. Logger 16 data from NTAS 14: wind U (m/s), wind V (m/s), SST (°C), surface salinity (psu), precipitation (mm).

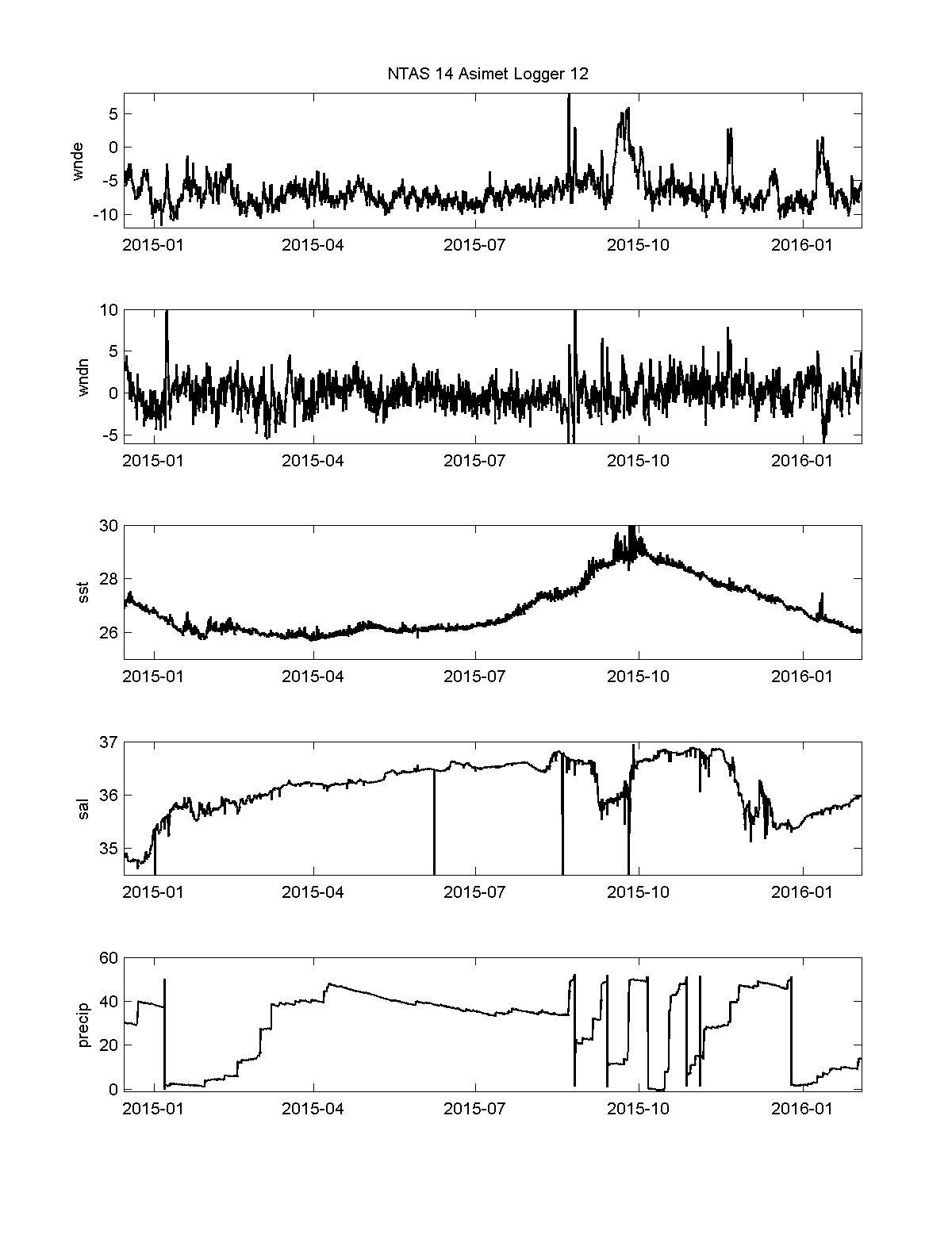


Figure III.1.2. Same as Fig. III.1.1 but for logger 12.

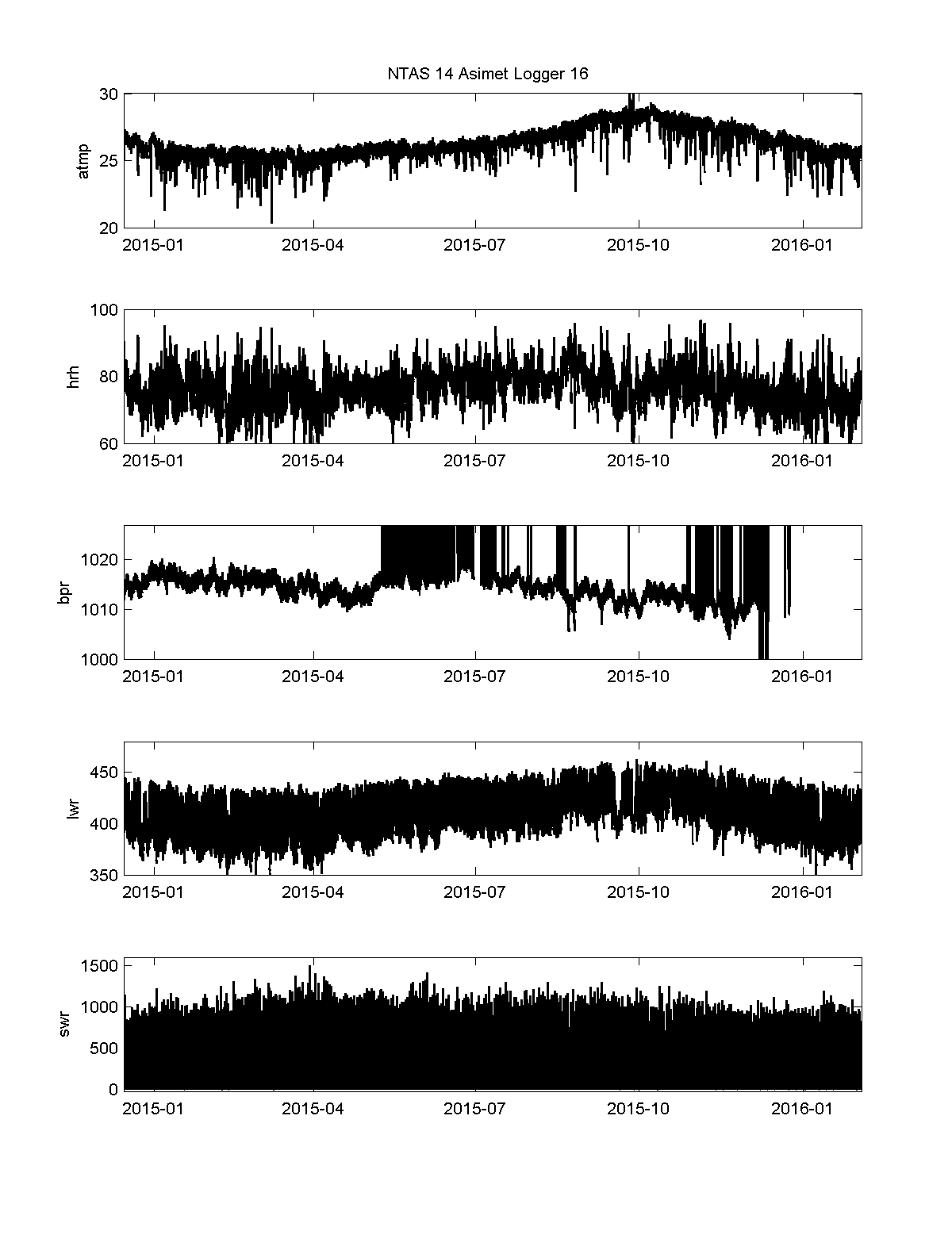


Figure III.1.3. Logger 16 data from NTAS 14: Air temperature (°C), relative humidity (%RH), barometric pressure (mbar), Longwave radiation (W/m2), Shortwave radiation (W/m2).

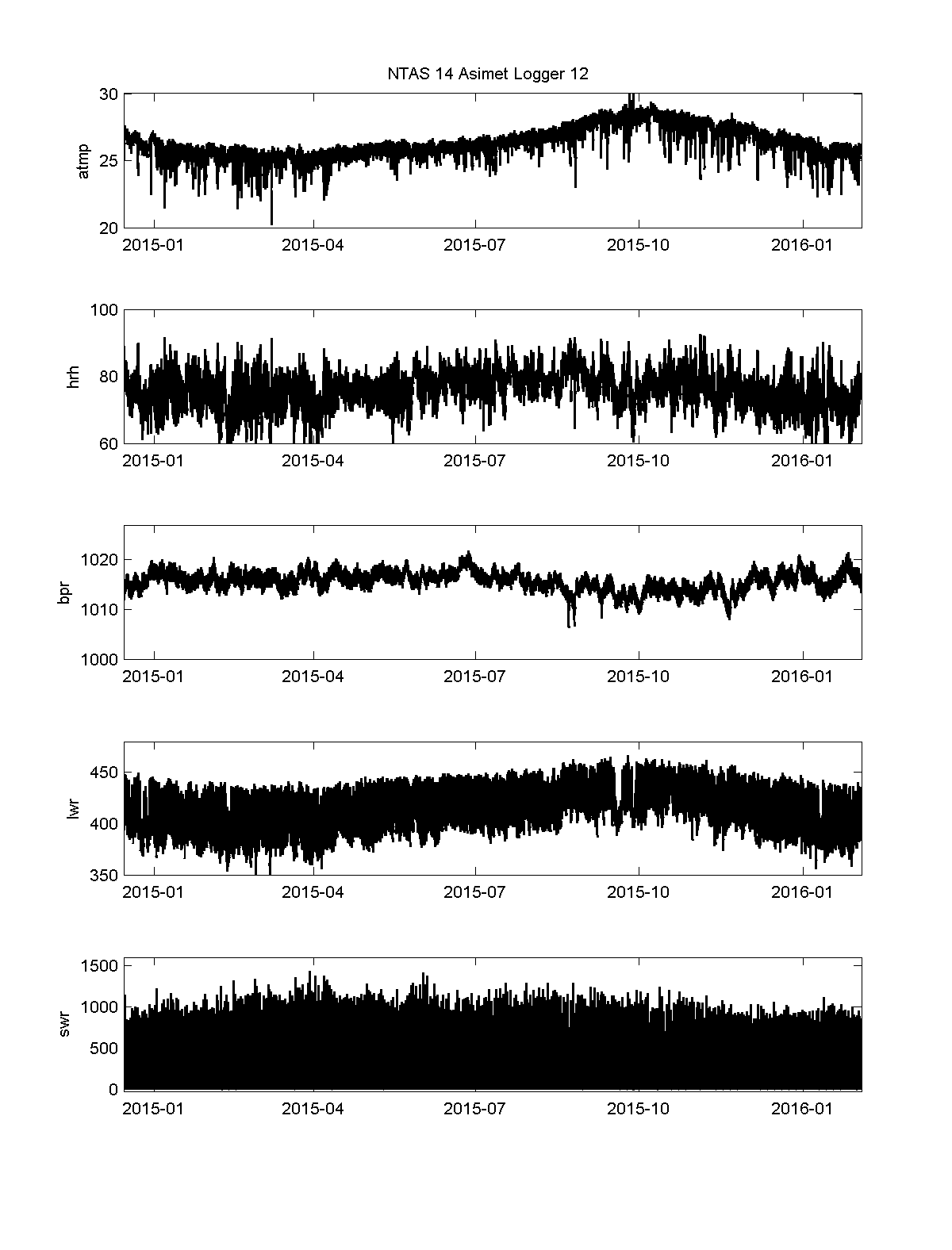


Figure III.1.4. Same as Fig. III.1.3 but for logger 12.

III.2 ASIMET modules

Data from ASIMET modules was also collected from module memory cards directly. BPR217 had more data than its counterpart on logger 16.

III.2.1. Data processing

Data files are in folder *NTAS14\_Recovery/data/asimet/modules*.

i) in Matlab run **do\_mods.m**

ii) in Matlab, run **do\_plot\_mods\_auto.m**

Step i) calls **ntas14meta.m** to create ntas14 meta data, then calls **get\_mod\_meta.m** on **N14modhts.csv**, then calls **load\_bin\_module.m** to read datasets from each ASIMET module; the data and metadata are then saved together for each sensor in files with name format: ntas14\_SSS\_XXX.mat (SSS is sensor type like ‘bpr’ for BPR barometric pressure; XXX is sensor’s serial number). Step ii) loads logger files and module files and plots both loggers and both modules data in same figure for each sensor. Figures are saved in *NTAS14\_Recovery/Figures/asimet/modules*.

III.3. Standalone LASCAR

There was one Lascar air temperature and humidity sensor deployed on the meteorological tower of NTAS 14. Data record from the Lascar is complete.

III.3.1 Data processing

Data files are in *NTAS14\_Recovery/data/lascar* and are listed in Table III.3.1.

i) In Matlab, run **do\_lascar.m**

ii) In Matlab, run **plot\_lascar.m**

Step i) calls **ntas14meta.m** to create platform meta data, then **loadlascar.m** to read the lascar data text file listed in Table III.3.1. Metadata for the instrument is then created and data and metadata are both saved in output Matlab file listed in Table III.3.1. Step ii) plots the data from the Matlab data file and saves the plots in folder ..*/../Figures/lascar*.

Table III.3.1. Lascar data file names recovered from NTAS 14.

|  |  |  |
| --- | --- | --- |
| **SN** | **Raw file** | **Matlab output file** |
| 1813 | NTAS14\_Lascar\_1813.txt | ntas14\_lascar\_1813.mat |

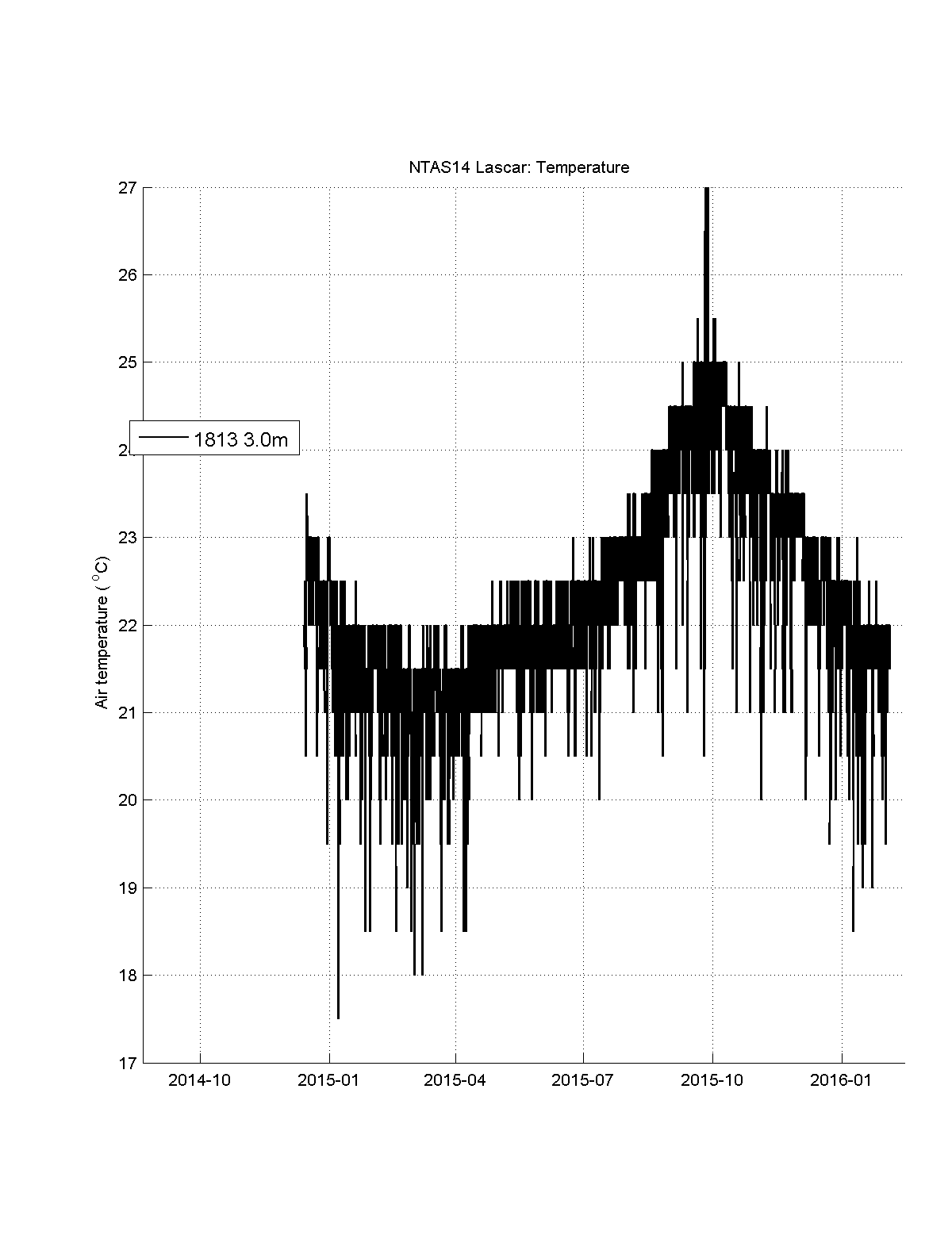


Figure III.3.1. NTAS 14 Lascar: air temperature record.

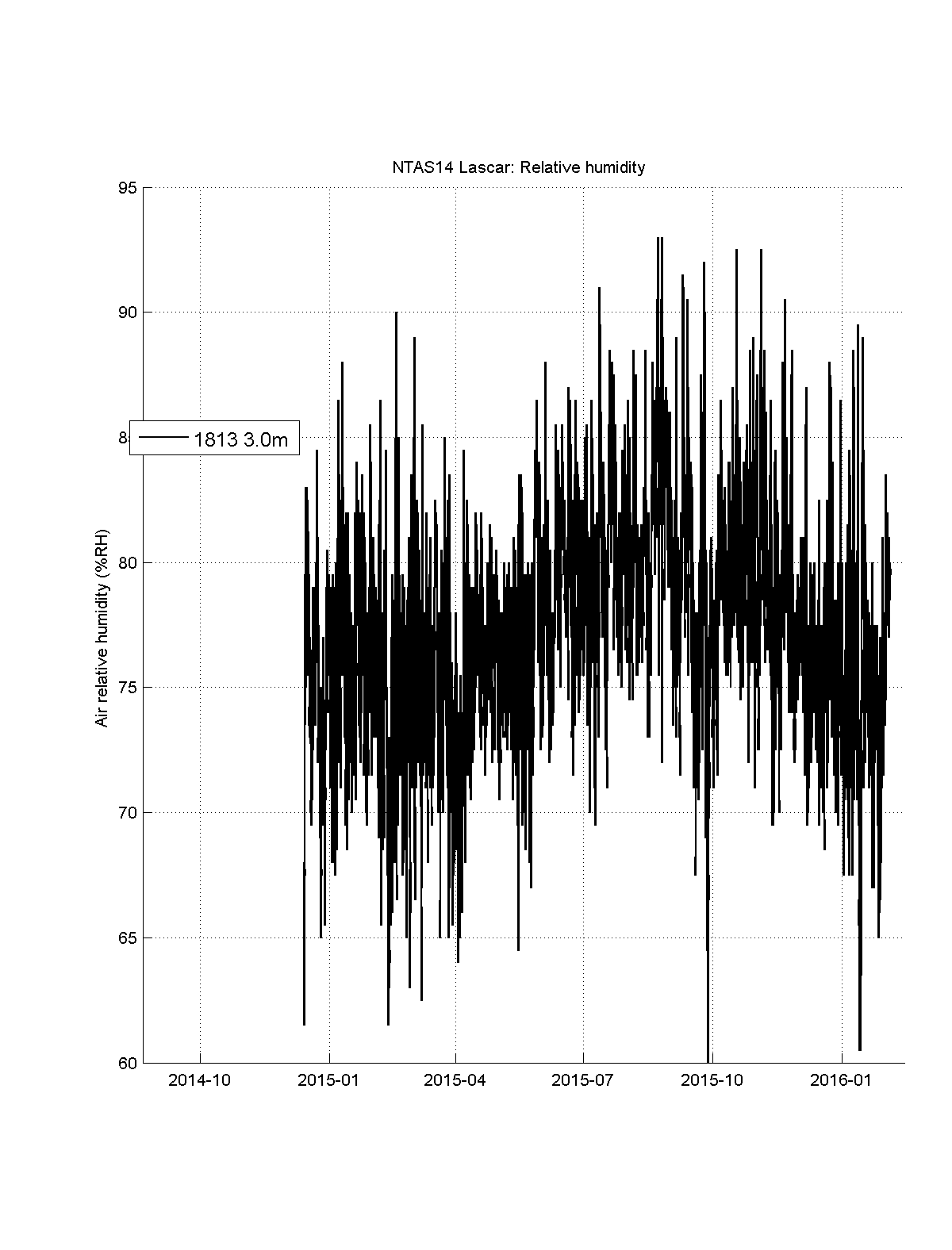


Figure III.3.2. NTAS 14 Lascar: air relative humidity record.

III.4. Vaisala WXT520

One Vaisala WXT520 was deployed on NTAS 14 buoy as a standalone unit and collected data for the whole duration of NTAS 14. Table III.4.1 lists the data file names, located in folder *NTAS14\_Recovery/wxt*.

Table III.4.1. Data file names for Vaisala WXT520 recovered from NTAS 14.

|  |  |  |
| --- | --- | --- |
| **SN** | **Raw file** | **Output Matlab file** |
| 5 | VWX005.DAT | ntas14\_wxt\_05.mat |

III.3.2 Data processing

i) in Matlab run **do\_load\_wxt520.m**

ii) in Matlab run **plot\_wxt.m**

Step i) calls **ntas14meta.m** to create meta data for platform Ntas14, then calls **load\_wxt520.m** which reads raw data file (see Table III.4.1) adds meta data for the WXT sensor and saves both data and metadata in output Matlab file (see Table III.4.1).

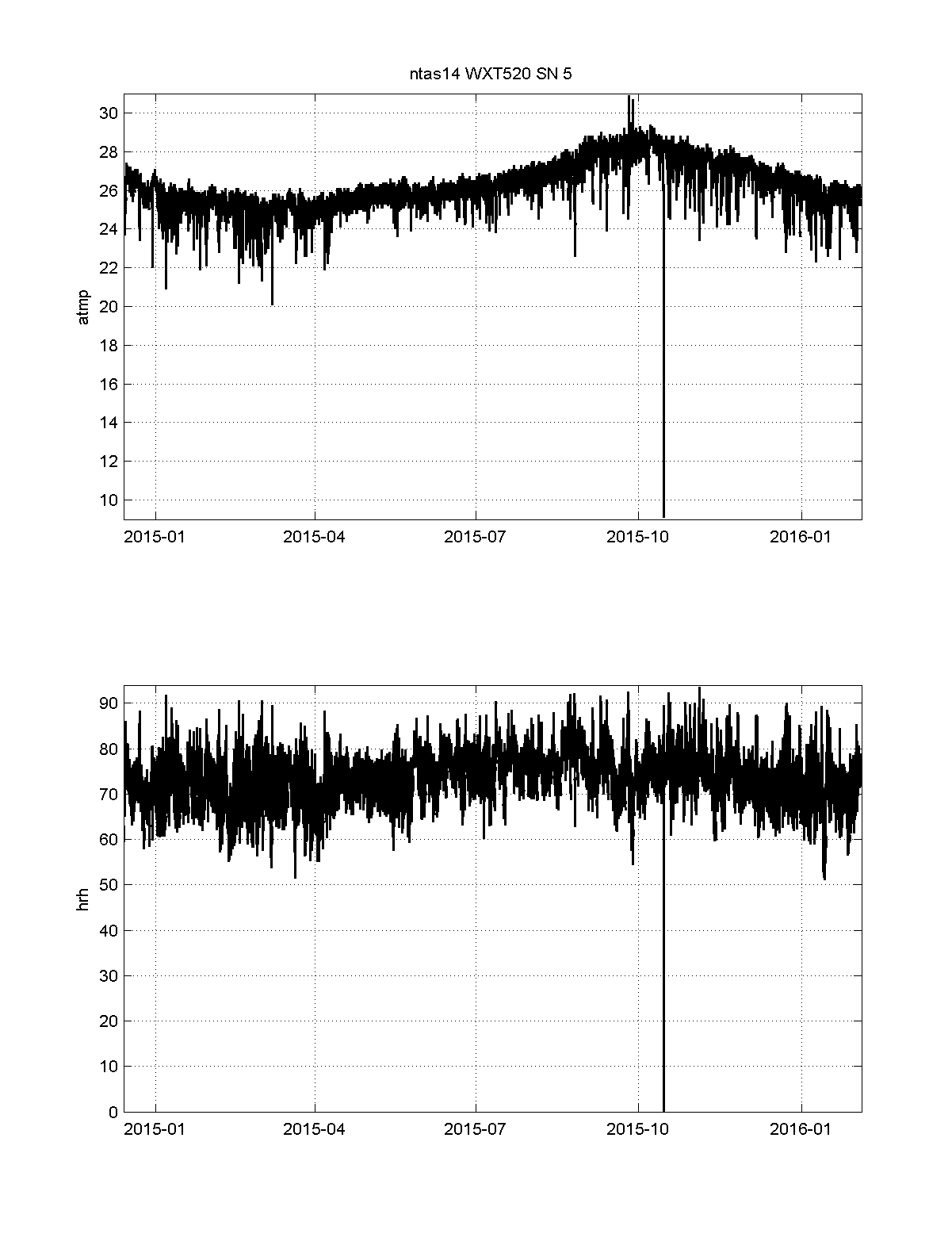


Figure III.4.1. NTAS 14 WXT: air temperature (°C) and relative humidity (%RH).

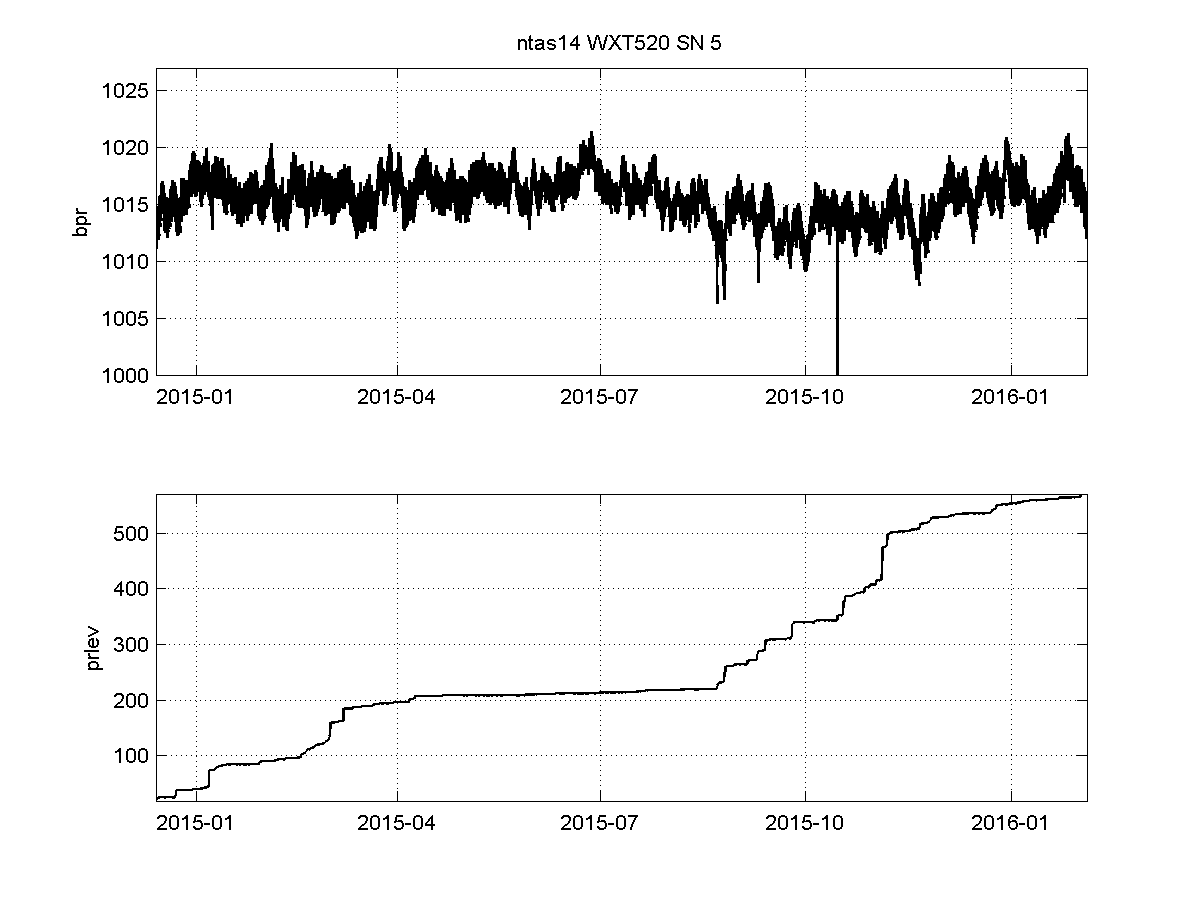


Figure III.4.2. NTAS 14 WXT: barometric pressure(mbar) and accumulated precipitation (mm).

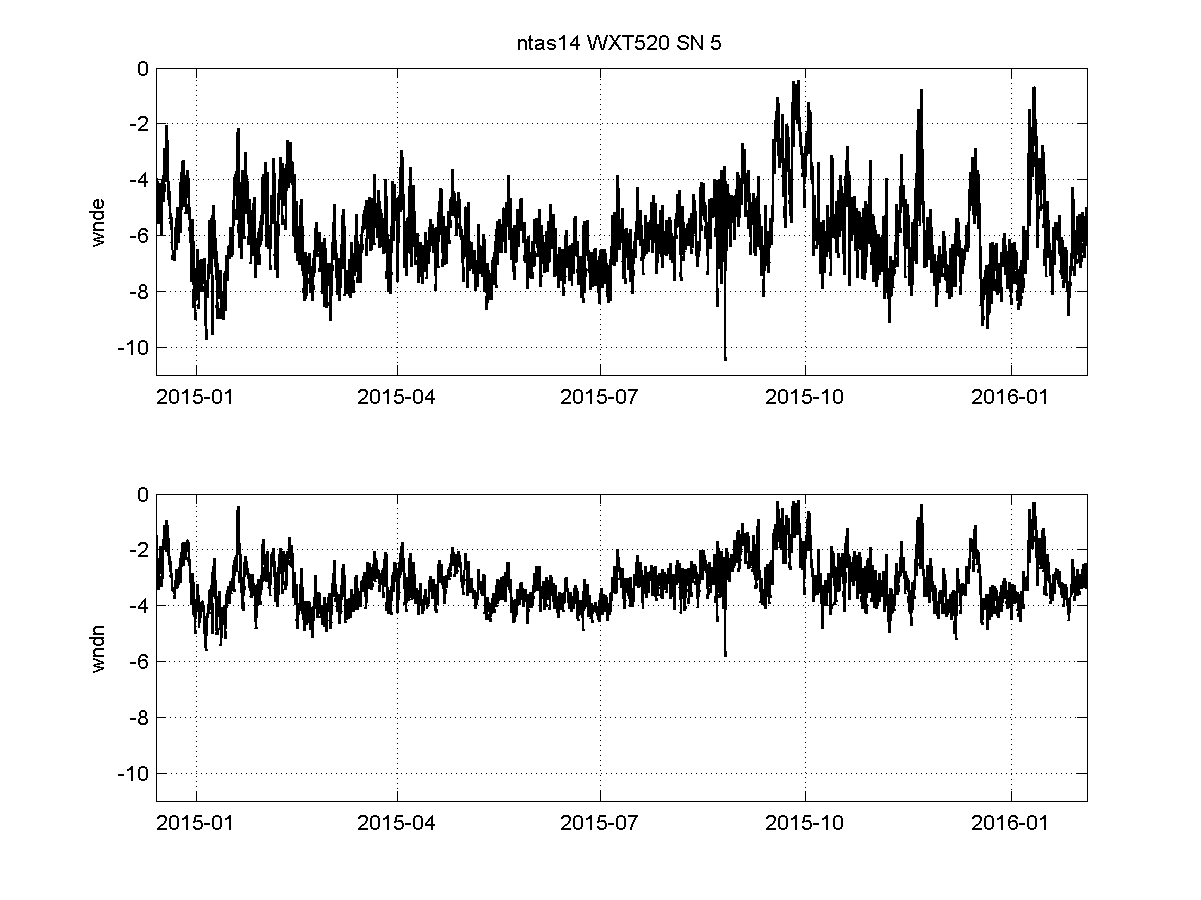


Figure III.4.3. NTAS 14 WXT: wind east and north (m/s).

III.5 Other sensors

III.5.1. XEOS MELO GPS sensor

The MELO (300034013207760) worked and transmitted telemetry data during NTAS 14. However, there was no data record to recover from the instrument’s memory at recovery.

III.5.2. WAMDAS wave package

WAMDAS operated normally during NTAS14. Data files were downloaded from memory card but no processing has been done at the time of this writing.