

Salinity Correction Toolbox

version 0.2.0

SOCIB DATA CENTER FACILITY

Manual -- DRAFT VERSION



Balearic Islands
Coastal Observing
and Forecasting
System



Document type:	DRAFT
Date:	2019-11-13

Description:	This document describes the development plans, installation process and requirements of the SOCIB Salinity Correction Toolbox (SCTB) version v0.2.0.
Authors:	C. Muñoz
Supervision:	J. G. Fernández, J. T. Allen
Involved Personnel:	DCF, GLF, RVF

DOCUMENT VERIFICATION LIST

Date:	Checked by (name)	SOCIB division	Ref.
2019-11-13	C. Muñoz	SOS	

DOCUMENT DISTRIBUTION LIST

Date:	Distribution to:
2019-11-13	public release

CHANGE RECORD

#	Date	Description	Author	Checked by
0.1	2019-11-13	DRAFT version document	C. Muñoz	

Index of contents

Overview	5
Related documentation	5
Requirements	6
Installation	7
Download and installation	7
Configuration settings	8
Main settings	8
Main paths	8
CTD correction paths	9
Glider correction paths	10
Execution	11
CTD correction Pack	11
Phase 1: Load Cruise Data	12
Phase 2: Preliminary Results	13
Phase 3: Default Salinity Correction	13
Phase 3a: Manual Salinity Correction	14
Phase 4: Apply Correction Results	15
Phase 5: Write L1_corr Data Files	15
Glider correction Pack	17
Normal Mode	17
Phase 1: Load CTD/Glider Data	18
Phase 2: Run Whitespace Maximisation	20

Phase 3: Export Metadata	23
Phase 4: Write L1_corr Data	23
External Glider Mode	26
Sections Mode	26
Profile Type Mode	28
Argo profiler correction Pack	29
Future Improvements	30
APPENDIX A: CTD Correction Pack Figures	31
Residuals insitu salinity	31
Theta/S diagrams corrected reference	32
Theta/S diagrams with and without corrections	33
Theta/S diagrams for all halfm corrected data available	33
APPENDIX B: Glider Correction Pack Figures	35
Cruise location map	36
Theta/S diagram halfm corrected vs glider test data	36
Theta/S diagram halfm corrected vs glider test data and glider test data corrected	37
APPENDIX C: Argo profiler Correction Pack Figures	37
Argo profiler Correction Pack	37

1. Overview

The salinity correction toolbox (SCTB) is a set of MATLAB/Python scripts and functions developed at SOCIB to perform scientific corrections on the data collected by RV SOCIB CTD and SOCIB glider fleet. SCTB identifies correction coefficients for salinity data taken from CTD rosette and glider.

2. Related documentation

Resource	Public link
SOCIB RV Facility Data Management Plan	https://repository.socib.es:8643/repository/entry/show?entryid=3f66c479-8355-4485-aa43-9c420db9d8d7
SOCIB Glider Facility Data Management Plan	https://repository.socib.es:8643/repository/entry/show?entryid=cc90e6c3-ba04-4131-810a-ecb0560428c9
Guidelines for the delayed mode scientific correction of glider data. WP 5 , Task 5.7, D5.15. Version 4.1.	http://dx.doi.org/10.25607/OPB-430
SOP_RVF_portasal8410-salinity-analysis	http://repository.socib.es/repository/entry/show?entryid=b82960e4-540f-4c22-be81-6b3eb9fdde1e

3. Requirements

- Linux (recommended) or Windows OS
- Matlab 2014, Matlab 2016 or Matlab 2018 .
- Python 2.7
- Django
- Matlab External libraries
 - SOCIB Glider Toolbox v1.0.0 - https://github.com/socib/glider_toolbox
 - GSW - <https://github.com/TEOS-10/GSW-Matlab>
 - m_map - <https://www.eoas.ubc.ca/~rich/map.html>
 - mexcdf -
<https://github.com/dcherian/tools/tree/master/netcdf/mexcdf/snctools>
 - seawater - http://www.cmar.csiro.au/datacentre/ext_docs/seawater.htm
- Python scripts:
 - salinity-corrections-writer
- Django project:
 - salinity-correction
- Matlab jar libraries:
 - Data base drivers 9.4 ([postgresql-9.4.1212.jre6.jar](#)). MATLAB interface with the PostgreSQL JDBC (Java Database Connectivity) driver. An Octave alternative would be needed. It may be installed from the distribution repositories as a package libpg-java (renamed to libpostgresql-jdbc- java in Debian repositories on March 2012).
 - NetCDF 4.2 library ([netcdfAll-4.2.jar](#) from Unidata)

4. Installation

4.1. Download and installation

- Clone repository at <https://github.com/socib/salinity-correction-toolbox>.
This creates a directory salinity-correction-toolbox-master with the following structure:
 - doc: contains general schematics and manuals
 - lib: contains the main binary files of the toolbox as follows:
 - ctd-salinity-correction-pack:
 - doc: contains specific schematics and manuals
 - res: contains specific resources as follows:
 - config: contains specific configuration file with paths.
 - img: contains specific images used for the GUI.
 - src: contains ctd corrections package code.
 - glider-salinity-correction-pack:
 - doc: contains specific schematics and manuals
 - res: contains specific resources as follows:
 - config: contains specific configuration file with paths.
 - img: contains specific images used for the GUI.
 - src: contains glider corrections package code.
 - res: contains general resources as follows:
 - config: contains general configuration file with main settings and paths.
 - db: mat files containing list of available NetCDF files in Thredds.
 - img: contains images used for the GUI.
 - ops:
 - README.md
 - medcoast.mat:
- Download and uncompress external libraries, binary files and matlab jars.
External libraries need to be allocated into a new folder ext/ resulting

4.2. Configuration settings

4.2.1. Main settings

- Edit general configuration file **set_main_paths.m**:
 - General configuration settings:
 - OPERATIONAL_MODE: case running application in operational mode in SOCIB server. 0 means NO, 1 means YES.
 - ONLINE_MODE: case working in the office network. 0 means NO, 1 means YES.
 - TEST_MODE: case testing the application. 1 means YES, and results will be stored in separate test directory. 0 means NO, and results will be stored in prod directory.
 - PORTABLE_MODE: case working in another device such as a laptop. 0 means NO, 1 means YES.
 - EXT_GLIDER: case an external glider dataset with different format needs to be analysed. 0 means NO, 1 means YES.
 - WHITESPACE_SECTIONS: case glider whitespace is gonna be adjusted by sections defined in idxToPProcess. 0 means NO, 1 means YES.
 - PROFILE_DIR_SECTIONS: case glider whitespace is gonna be adjusted by sections defined in profile_directions. 0 means NO, 1 means YES.

4.2.2. Main paths

- Toolbox Main Paths settings:
 - MainPath.mainToolbox
 - MainPath.code
 - MainPath.resources
 - MainPath.libraries
 - MainPath.db
- CTD salinity correction Pack Main Paths settings:
 - MainPath.main
 - MainPath.dataCtdL1Thredds
 - MainPath.dataCtd
 - MainPath.dataPortasal
 - MainPath.dataCTDIinsituCalHistory
- Glider salinity correction Pack Main Paths settings:
 - MainPath.main
 - MainPath.dataGliderL1Thredds
 - MainPath.metadataGliderCorrected
 - MainPath.dataPortasal

- MainPath.dataCTDIInsituCalHistory
- MainPath.codePack: path to the glider correction pack source code
- MainPath.resourcesPack: path to specific resources of the package like list of available files in thredds, images, etc

- Argo salinity correction Pack Main Paths settings:

4.2.3. CTD correction paths

- Edit specific configuration file **ctd_sc_set_ancillary_paths.m**:
 - Match connections to main paths:
 - Path.main
 - Path.dataCtdL1Thredds
 - Path.dataInsituSalinityRaw
 - Path.dataCtd
 - Path.dataCTDIInsituCalHistory
 - Set specific paths for storing output data and figures:
 - Path.fig
 - Path.dataOut
 - Check specific paths to package code:
 - Path.code
 - Path.libraries
 - Path.resources
 - Set specific paths through data directories:
 - Path.dataCtdBtl
 - Path.dataCtdBinAvgHalfm
 - Path.dataCtdBtlMat
 - Path.dataInsituSalinityConverted
 - Path.dataCorrectionFiles
 - Path.dataCorrectedMat
 - Path.dataCorrectedMat5mBinAvg
 - Path.dataCorrectedMatHalfmBinAvg
 - Path.dataCorrectedMatHalfmBinAvgAll
 - Path.dataCorrectedNc
 - Path.dataCorrectedNc5mBinAvg
 - Path.dataCorrectedNcHalfmBinAvg
 - Path.dataCorrectionCoefficients
 - Path.dataCorrectionCoefficientsMat

- Path.dataCorrectionCoefficientsNc

- Set specific paths through output figures directories:
 - Path.figsTSdiagsAllCruisesThreddsL1
 - Path.figsTSdiagsSingleCruisesThreddsL1
 - Path.figsTSdiagsSingleCruisesThreddsL1Zoom
 - Path.figsTSdiagsComparedCruisesThreddsL1
 - Path.figsResidualsInsituSalinity
 - Path.figsTSdiagsWithWithoutCorrections
 - Path.figsTSdiagsCorrectedReference
 - Path.figsTSdiagsCorrectedReferenceHalfmSingle
 - Path.figsTSdiagsCorrectedReferenceHalfmAll

4.2.4. Glider correction paths

- Edit specific configuration file **glider_sc_set_ancillary_paths.m**:

- Set specific paths for storing output data and figures:
 - GSCPPath.figs
 - GSCPPath.dataOut
- Set specific paths through data directories:
 - MainPath.dataCorrectionFiles
 - MainPath.dataCorrectedMat
 - MainPath.dataHalfmCtdCorrected
 - MainPath.dataCorrectedNc
 - MainPath.dataCorrectionCoefficients
 - MainPath.dataCorrectionCoefficientsMat
 - MainPath.dataCorrectionCoefficientsNc
- Set specific paths through output figures directories:
 - MainPath.outFigsCorrection
 - MainPath.figsTSdiagsCorrectedReference
 - MainPath.deploymentFigsTSdiagsCorrectedReference

- Edit specific configuration file **glider_load_sections_idx.m**:

This file will be only edited in case glider data needs to be corrected by sections. In such case a vector needs to be provided with the indices of the sections defined.

5. Execution

In this section we explain a few examples to guide users for a complete correction of CTD, Glider and Argo profilers data with SCTB. This is a practical description of the processing. The explanation of the processing scripts and function is in the [Processing](#) section. For more details and processing options refer to next sections. Example of figures that are produced by the SCTB are in Appendix A and B. As mentioned in the introduction of this document, users can find examples of produced netCDF at:

CTD L1_CORR:

http://thredds.socib.es/thredds/catalog/research_vessel/ctd/catalog.html

Glider L1_CORR:

<http://thredds.socib.es/thredds/catalog/auv/glider/catalog.html>

Follow the steps that are described below as guidelines to process data in the following modes:

- CTD correction
- Glider correction
- Argo profiler correction

These examples are explained in the context of linux users but windows users could perform the equivalent steps.

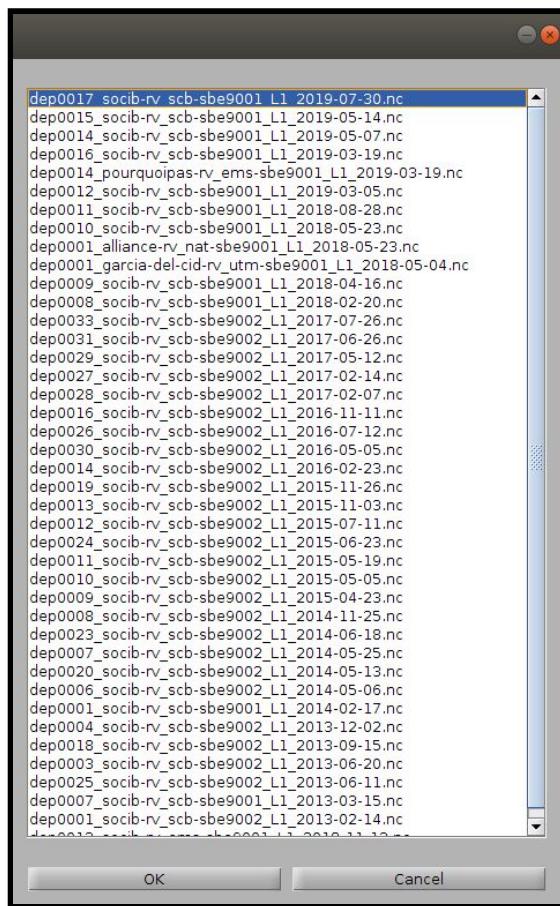
5.1. CTD correction Pack

- Open Matlab, set paths to the SCTB main directory.
- Execute function `ctd_salinity_correction_main`



5.1.1. *Phase 1: Load Cruise Data*

Phase 1 loads L1 netcdf data from CTD. This will also automatically load data from ascii bottle files and portasal samples analysis.



5.1.2. Phase 2: Preliminary Results

Phase 2 generates Theta/S diagrams of figures of all cruises available.

5.1.3. Phase 3: Default Salinity Correction

Phase 3 automatically performs the correction considering the default settings.

For the selected cruise, matches up the in situ bottle salinity sample with the corresponding .btl data considering all the samples available from the beginning.

This may be executed considering only one CTD sensor available or two CTD sensors available:

- With two CTD sensors available, this phase will:
 - Identify outliers where the difference between the two ctd sensors (Sal00 and Sal11) is larger than 0.01.
 - Remove outliers that deviate more than 2 std's from the mean.

NOTE: This setting can be altered manually in function *Stage_7a_mbtl_in situ_SAL_matchup* through the variable XX.

- Remove outliers that deviate more than 1 std's from the mean.

NOTE: This setting can be altered manually in function *Stage_7a_mbtl_in situ_SAL_matchup* through the variable XX2.

- With one CTD sensor available, this phase will:
 - Remove outliers that deviate more than 2 std's from the mean.

NOTE: This setting can be altered manually in function *Stage_7b_mbtl_in situ_SAL_matchup_unique_Sensor* through the variable XX.

 - Remove outliers that deviate more than 1 std's from the mean.

NOTE: This setting can be altered manually in function *Stage_7b_mbtl_in situ_SAL_matchup_unique_Sensor* through the variable XX2.

After the corrections are applied, the following details are provided in the output figures:

- Sensor00, Sensor11 coefficients.
- Mean residuals for Sensor00, Sensor11 and standard deviations from the means.

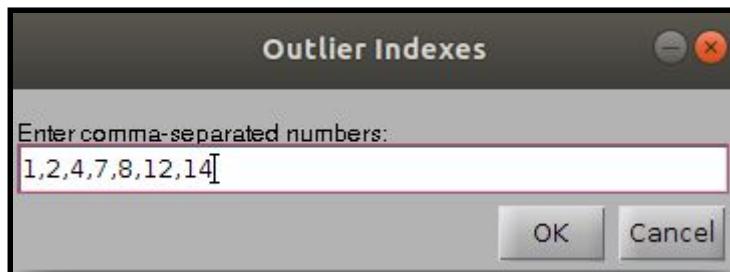
Sensor00 Coeff = 1.000181592143150
mean resid Sensor00: -2.9834e-05 ± 0.0023175
Sensor11 Coeff = 1.000066257223724
mean resid Sensor11: -3.7293e-05 ± 0.0025149

NOTE: The figures created in this phase are available in [APPENDIX A](#).

5.1.4. Phase 3a: Manual Salinity Correction

Phase 3a manually performs the correction considering the default settings. A prompt appears to include those samples that need to be removed in order to avoid them being considered for calculating the final correction coefficients.

NOTE: Numbers of samples need to be separated by commas as indicated in the figure below.



After this point, the same conditions apply in this case than in Phase 3.

5.1.5. *Phase 4: Apply Correction Results*

Phase 4 performs the following actions, once the user agrees with the corrections:

- Create NetCDF metadata file containing the corrections information.
- Store halm corrections in **allDATA_halfm_corrected.m** file.
- Create NetCDF data file applying the corrections over the halfm ascii CTD data files.
- Create Theta/s diagram with all halfm data available in **allDATA_halfm_corrected.m** file.

NOTE: The figures created in this phase are available in [APPENDIX A](#).

5.1.6. *Phase 5: Write L1_corr Data Files*

Phase 5 applies the corrections over the existing CTD NetCDF files available in Thredds by duplicating the file, applying the corrections on both conductivity and salinity, and also creating new variables corrected into the new duplicated file that will be labeled under L1_corr processing level.

This process is performed through the **Python script salinity-corrections-writer**.

The corrections performed over this script are also included into SOCIB database schema called corrections.

The information concerning such corrections may be consulted through the admin interface of a **Django application called salinity-correction-metadata** (see figure below).

The first screenshot shows a list of CTD Salinity Corrections. It includes columns for Deployment (checkbox), CTD SALINITY CORRECTION APPROVED RESULTS (checkbox), INSTRUMENT (dropdown), and CRUISE (text). Four entries are listed:

Action:	Deployment	CTD SALINITY CORRECTION APPROVED RESULTS	Instrument	Cruise
<input type="checkbox"/>	SOCIB_ENL_CANALES_NOV2018_EMS-SBE9001	<input checked="" type="checkbox"/>	EMS-SBE9001	SOCIB_ENL_Canales_Nov2018
<input type="checkbox"/>	SOCIB_ENL_CANALES_AGO2018_SCB-SBE9001	<input checked="" type="checkbox"/>	SCB-SBE9001	SOCIB_ENL_Canales_Ago2018
<input type="checkbox"/>	SOCIB_ENL_CANALES_JUL2019_SCB-SBE9001	<input checked="" type="checkbox"/>	SCB-SBE9001	SOCIB_ENL_Canales_Jul2019

The second screenshot shows the detailed edit form for a CTD Salinity Correction object. It contains fields for deployment, date last service, interval service, sensor correction coefficients, residual salinity differences, Ctd salinity correction sensor, flag, outliers removed, comments, and a checkbox for 'Ctd salinity correction approved results'. It also lists date added, created by, date modified, and updated by.

Fields automatically filled are the following:

- Ctd salinity correction deployment
- Sensor 01 correction coefficient
- Sensor 01 residual salinity differences mean
- Sensor 01 residual salinity differences std
- Sensor 02 correction coefficient
- Sensor 02 residual salinity differences mean
- Sensor 02 residual salinity differences std
- Ctd salinity correction sensor 02
- Date added
- Created by
- Date modified
- Updated by

5.2. Glider correction Pack

5.2.1. Normal Mode

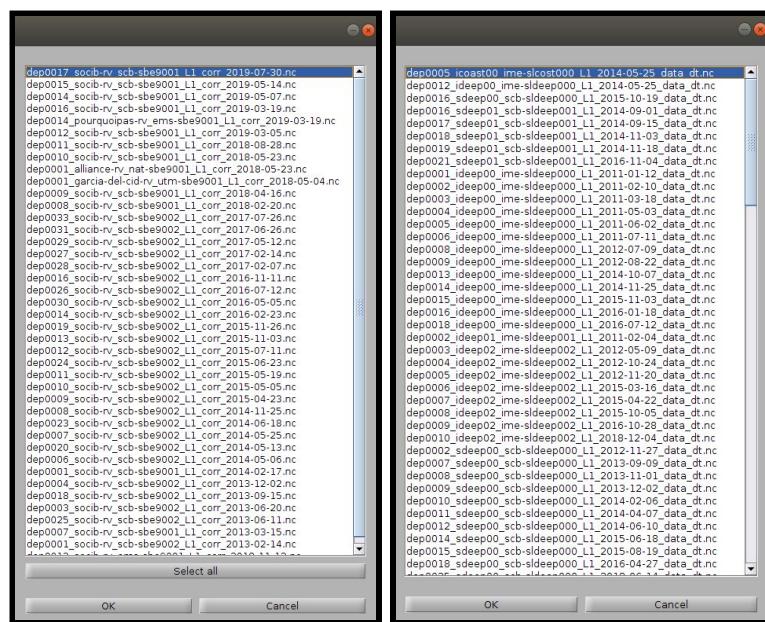
- Open Matlab, set paths to the SCTB main directory.
- Execute function `glider_salinity_correction_main`



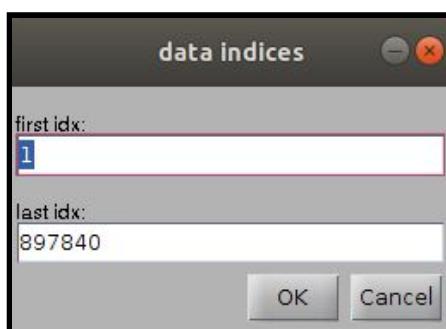
5.2.1.1. Phase 1: Load CTD/Glider Data

Phase 1 loads L1_corr data from **allDATA_halfm_corrected.m** CTD and Glider NetCDF dataset that needs to be corrected.

- Two different prompts will appear to select the datasets. Multi CTD datasets are available to be selected.



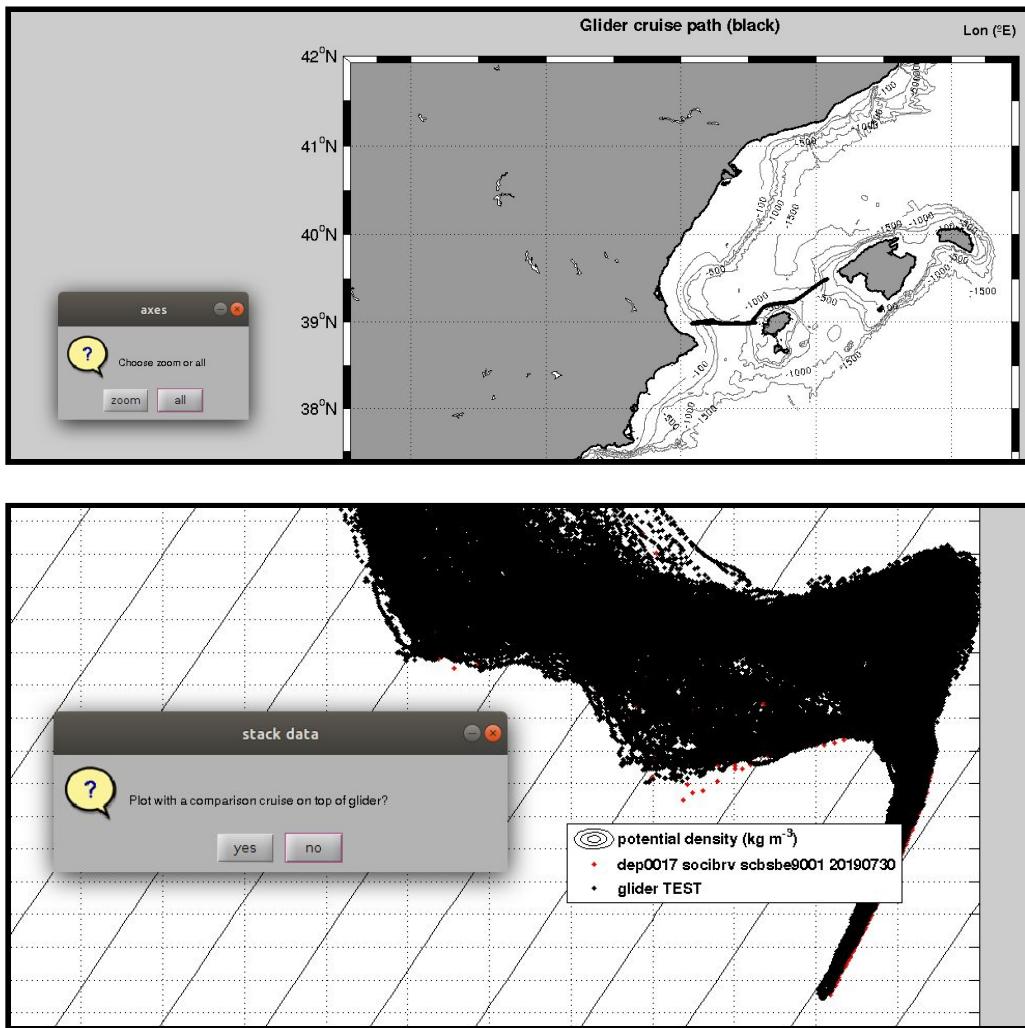
- Once data is selected, a new prompt will appear with the indices ranging the whole dataset by default. Here the user may select a specific period within the dataset in order to force the whitespace maximisation to such specific time period.

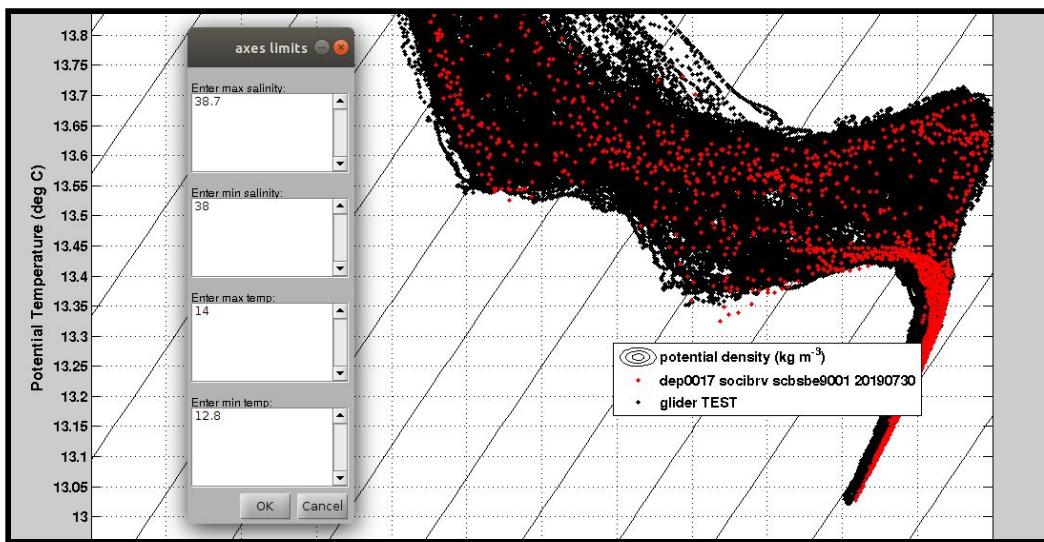


- After this, a series of figures will appear showing the spatial distribution of the selected datasets and a Theta/S plot with both distributions too in order that the user is able to decide whether this is the proper comparison to perform.

NOTE: The figures created in this phase are available in [APPENDIX B](#).

Figures may be adjusted by selecting zoom or general distribution:





- After loading the data and creating the preliminary figures, some additional settings may be adjusted, such as adding a keyword to the figure names and removing specific data from Mallorca channel in case the datasets belong to the canales cruises.

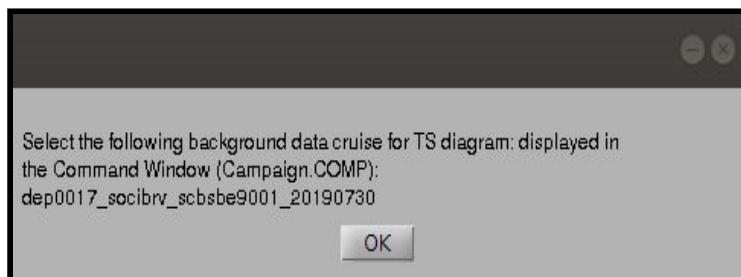


5.2.1.2. Phase 2: Run Whitespace Maximisation

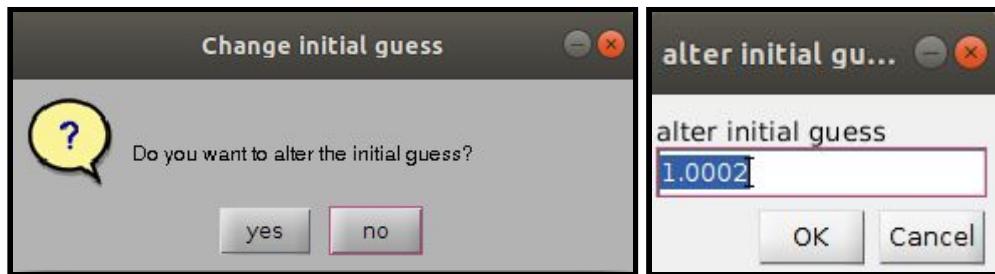
Once the data is loaded, Phase 2 performs the whitespace maximisation.

- Axes for the whitespace may be altered by the user before starting the process. This is a subjective activity and it will depend on the type of datasets to compare. In general, the user needs to consider some limitations of the process, eg, salinitymax may be adjusted as a significant higher value than the real (eg 39 psu) one in order to facilitate the process of comparison. Temperature limits may be setted depending again on the distributions that are going to be compared and the similitudes between them. Generally, ranges between 12.8 - 13.8 deg celsius are the most common.

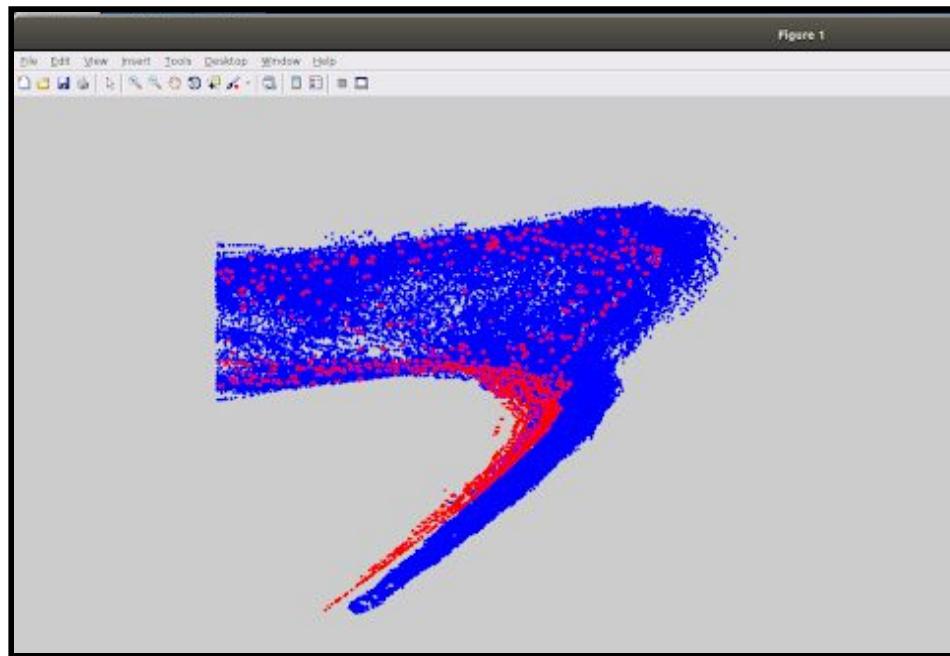




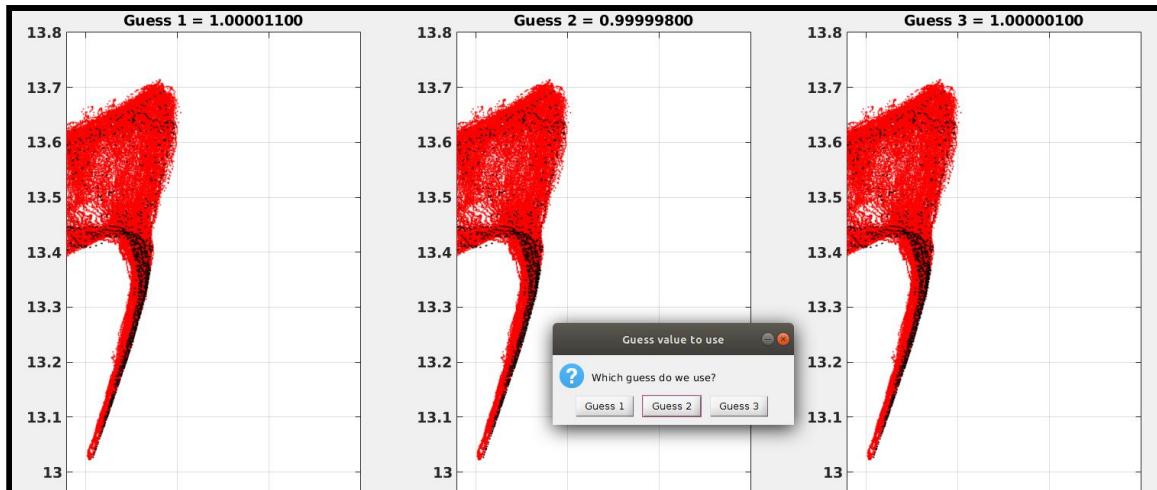
- The whitespace maximisation process will use 3 different default initial guesses (0.9999, 1.0000, 1.0001), but they can be also altered by the user before starting each of the three iterations.



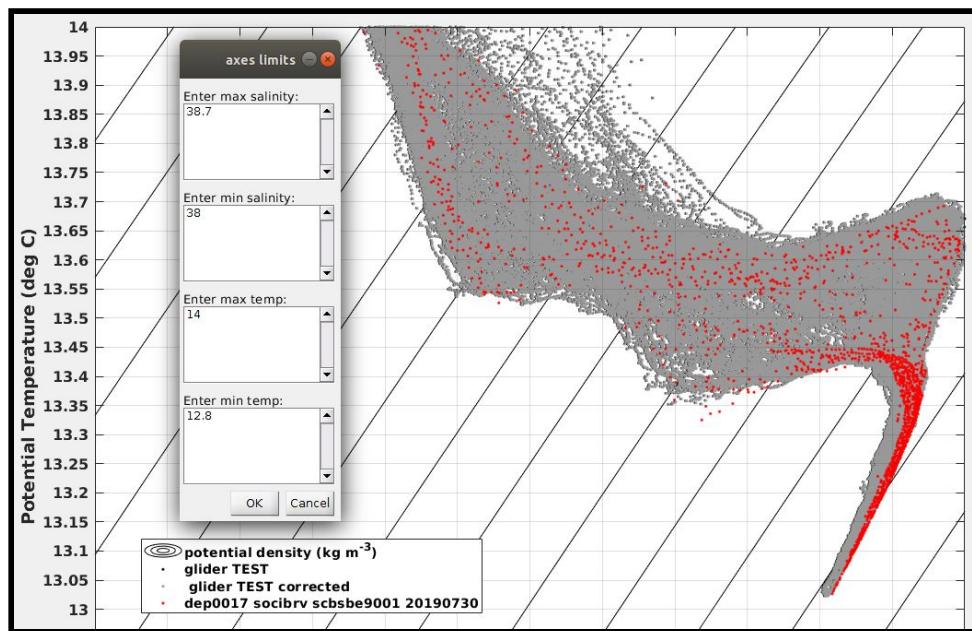
- When the process is running the user will see comparison figures being created each second where the test T/S distribution will appear in different positions in order to maximise the whitespace within the figure.



- Once the three iterations are completed using 3 different initial guesses, the user will be advised to select the one that better suits the particular case.



- After selecting the best solution, a final figure will appear presenting the initial test distribution and the final corrected distribution, including the CTD reference data too.

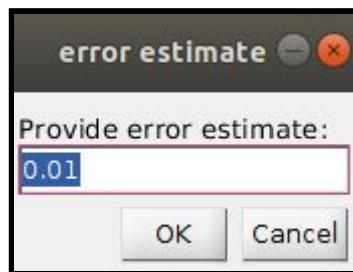


NOTE: The figures created in this phase are available in [APPENDIX B](#).

5.2.1.3. Phase 3: Export Metadata

In Phase 3, the user accepts the results and thus, the metadata file containing the correction coefficient and ancillary information is created.

- Before exporting the metadata file, the user will be asked to introduce the value for the confidence level expected with the performed correction. Default is setted as 0.01.



5.2.1.4. Phase 4: Write L1_corr Data

Phase 4 applies the corrections over the existing L1 Glider NetCDF files available in Thredds by duplicating the file, applying the corrections on both conductivity and

salinity, and also creating new variables corrected into the new duplicated file that will be labeled under L1_corr processing level.

This process is performed through the Python script salinity-corrections-writer.

The corrections performed over this script are also included into SOCIB database schema called corrections.

The information concerning such corrections may be consulted through the admin interface of a Django application called salinity-correction (see figure below).

Django administration

Home · Salinity_Correction · Glider Salinity Corrections

Select Glider Salinity Correction to change

Search: [] Go 0 of 46 selected

Action:	GLIDER SALINITY CORRECTION APPROVED RESULTS	DEPLOYMENT NAME	DEPLOYMENT CODE	INSTRUMENT
<input type="checkbox"/>	SOCIB_ENL_Canales_Jul2014_sdeep01		0014	SCB-SLDEEP001
<input checked="" type="checkbox"/>	SOCIB_ENL_CANALES_MAY2019_IDEEP02_GFMR0090		0012	IME-SLDEEP002
<input checked="" type="checkbox"/>	IMEDEA_TEST_PRE-CALYPSO2019_DEC2018_IDEEP02_GFMR0083_CANALES		0010	IME-SLDEEP002
<input checked="" type="checkbox"/>	SOCIB_ENL_CANALES_FEB2019_SDEEP04_GFMR0085		0013	SCB-SLDEEP004
<input checked="" type="checkbox"/>	SOCIB_ENL_CANALES_JUN2018_SDEEP00_GFMR0079		0026	SCB-SLDEEP000
<input checked="" type="checkbox"/>	SOCIB_ENL_CANALES_APR2019_SDEEP04_GFMR0088		0014	SCB-SLDEEP004

Django administration

Home · Salinity_Correction · Glider Salinity Corrections · GliderSalinityCorrection object

Change Glider Salinity Correction

Glider salinity correction deployment: SOCIB_ENL_CANALES_JUN2018_SDEEP00_GFMR0079

Date last service prior to mission: Today |

Note: You are 1 hour ahead of server time.

Interval service to mission:

Glider salinity correction sensor 01: -----

Sensor 01 correction coefficient: 1.000091

Residual salinity differences std background data: 0.005758 given temperature and given pres

Salinity error estimate: 0.01

Glider salinity correction background data: SOCIB_ENL_CANALES_AGO2018_SCB-SBE9001

Theta-sal whitespace for correction potential temperature and practical salinity ranges: Salinity: 38.48 to 38.8 psu, Temperature: 12.

Comments:

Glider salinity correction approved results

Fields automatically filled are the following:

- Glider salinity correction deployment
- Sensor 01 correction coefficient
- Residual salinity differences std background data
- Salinity error estimate
- Glider salinity correction background data
- Theta-sal whitespace for correction potential temperature and practical salinity ranges
- Date added
- Created by
- Date modified
- Updated by

5.2.2. External Glider Mode

This functionality was created just for attending a specific requirement from an external project (CALYPSO). This is currently a proof of concept inserted in the tool.

- Set EXT_GLIDER = 1;
- Read external glider files. Currently mat files under Scripps format is available. Manually edit **phase_1_load_data.m** lines 24-26, 45-55, 135-143, 360-364

5.2.3. Sections Mode

- In MATLAB SCTB:
 - Set WHITESPACE_SECTIONS = 1;
 - Edit **glider_load_sections_idx.m** file as follows, by specifying in a commented line the glider file that needs to be edited and creating below a vector containing the indices that contain the sections to be processed:

```
%% dep0002_sdeep05_scb-sdeep005_L1_2017-11-16 sections
idxToProcess = [1, 245201, 376501, 615101, 772869];
```

- Run SCTB Phases 1, 2 and 3 as described in Normal Mode.

NOTE: Phase 2 will run automatically as many times as indices provided in the sections indices list

- In Python salinity-corrections-writer:

- Edit **config.py** file including the coefficients obtained and the indices list as follows:

```
multi_coeff_A = True
```

```
coeff_A_list = [1.000339000000000, 1.000139000000000, 1.000130999999999,  
1.000101000000000]
```

```
idx_list = [1, 245201, 376501, 615101, 772869]
```

- In MATLAB SCTB run SCTB Phase 4 as described in Normal Mode.
- In Django salinity-correction-metadata, fill comments metadata in salinity-correction DB:



- Manually edit L1_corr variable attributes:

conductivity_corr: Array of 64 bit Reals [time = 0..772869]

time: []

```
comment: Salinity calibration reference: Allen, J.T.; Fuda,J-L.;Perivoliotis, L.; Munoz-Mas, C.; Alou, E. and Reeve, K. (2018) Guidelines for the delayed mode scientific correction of glider data. WP 5, Task 5.7, D5.15. Version 4.1. Palma de Mallorca, Spain, SOCIB - Balearic Islands Coastal Observing and Forecasting System for JERICO-NEXT, 20pp. (JERICO-NEXT-WP5-D5.15-140818-V4.1). DOI: 10.25607/OPB-430
_fillValue: NaN
theta-sal_whitespace_for_correction: Salinity: 38.48 to 38.8 psu, Temperature: 12.8 to 13.8 deg C
correction_coefficient_A: 1.000339000000000, 1.000139000000000, 1.000130999999999, 1.000101000000000
sources: sci_water_cond
summary_method_error_estimate: error estimate is based on the range of salinity values of the comparison cruises at about 13 deg C (i.e. at the tail end of the deepest values on the Theta-S diagram)
background_data_used_for_correction: Background comparison Cruises used: 1) dep0008_socib-rv_scb-sbe9001_L1_corr_2018-02-20, 2) dep0033_socib-rv_scb-sbe9002_L1_corr_2017-07-26.
coordinates: time depth latitude longitude
observation_type: corrected_measurements
long_name: water conductivity
standard_name: sea_water_conductivity
summary_method_report: For further details, refer to report...TBC
conductivity_thermal_corr_used: NO, unavailable
units: S m-1
calibration_equation: COND_CORR=A*COND_01
glider_report: http://www.socib.es/?seccion=gliderPage&facility=gliderReports
summary_method: whitespace area maximisation of a Theta-S diagram comparison, between glider data and other nearby (in time and space) cruises was employed
```

salinity_Corr: Array of 64 bit Reals [time = 0..772869]

time: []

```
comment: Salinity calibration reference: Allen, J.T.; Fuda,J-L.;Perivoliotis, L.; Munoz-Mas, C.; Alou, E. and Reeve, K. (2018) Guidelines for the delayed mode scientific correction of glider data. WP 5, Task 5.7, D5.15. Version 4.1. Palma de Mallorca, Spain, SOCIB - Balearic Islands Coastal Observing and Forecasting System for JERICO-NEXT, 20pp. (JERICO-NEXT-WP5-D5.15-140818-V4.1). DOI: 10.25607/OPB-430. Corrections performed over 4 different sections identified in indices list idx_list = [1, 245201, 376501, 615101, 772869]
_fillValue: NaN
sources: conductivity temperature pressure
background_data_used_for_correction: Background comparison Cruises used: 1) dep0008_socib-rv_scb-sbe9001_L1_corr_2018-02-20, 2) dep0033_socib-rv_scb-sbe9002_L1_corr_2017-07-26.
coordinates: time depth latitude longitude
observation_type: corrected_derived_from_conductivity_corr
long_name: water salinity
standard_name: sea_water_salinity
residual_salinity_differences_std_background_data: 0.002233 given temperature and given pressure
units: PSU
salinity_error_estimate: 0.01
method: sw_salt
summary_details: Refer to meta.conductivity_corr.attributes
```

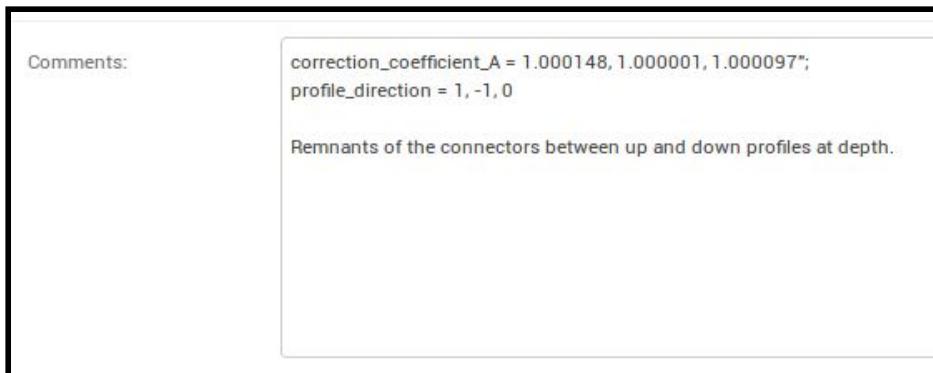
5.2.4. Profile Type Mode

- In MATLAB SCTB:
 - Set PROFILE_DIR_SECTIONS = 1;
 - Run SCTB Phases 1, 2 and 3 as described in Normal Mode.

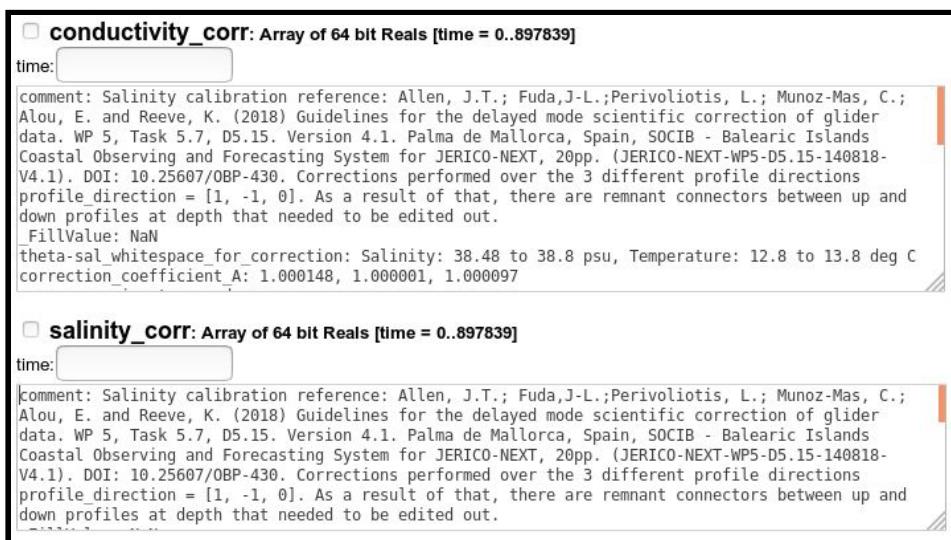
NOTE: Phase 2 will run automatically three times, considering profile indexes [1, 0, -1].
- In Python salinity-corrections-writer:
 - Edit **config.py** file including the coefficients obtained and the indices list as follows:

```
multi_coeff_A = True
coeff_A_list = [1.000267, 1.000199, 1.000201]
idx_list = [1, -1, 0]
```

- In MATLAB SCTB run SCTB Phase 4 as described in Normal Mode.
- In Django salinity-correction-metadata, fill comments metadata in salinity-correction DB:



- Manually edit L1_corr variable attributes:



5.3. Argo profiler correction Pack

6. Future Improvements

Code refactoring -> move to python and simplify

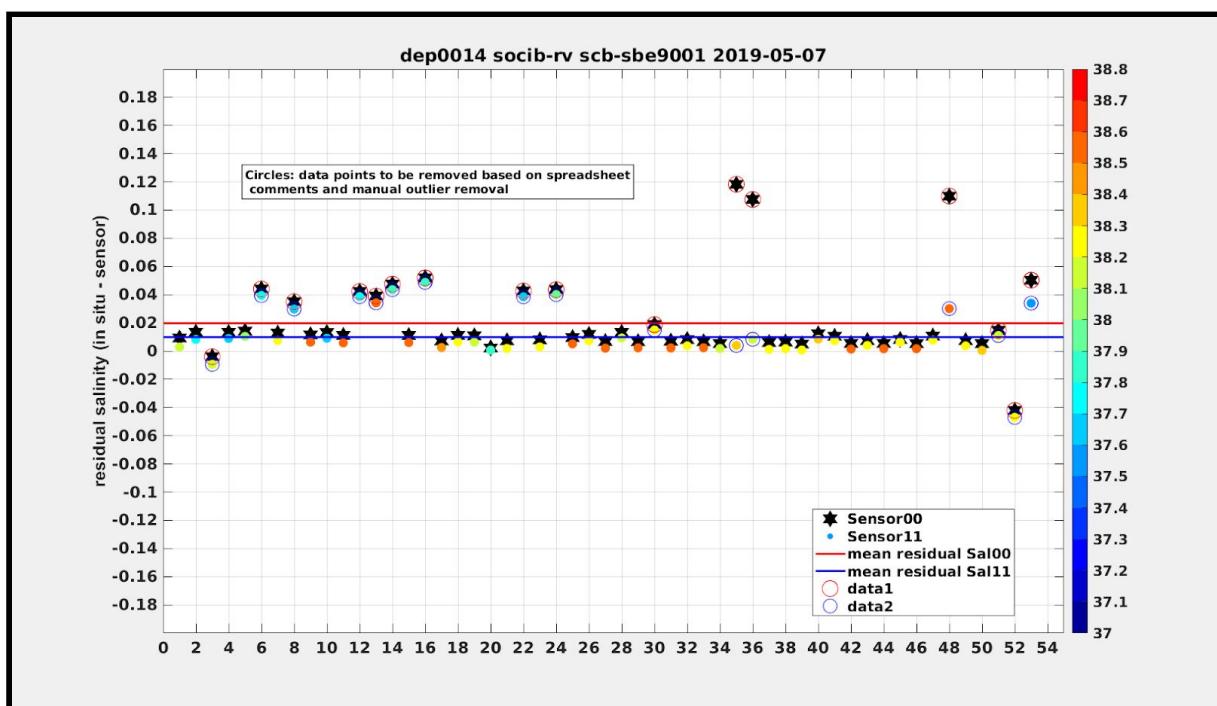
outputs -> ensure figures are being stored, avoid creation of metadata files and use DB instead to write but also to read

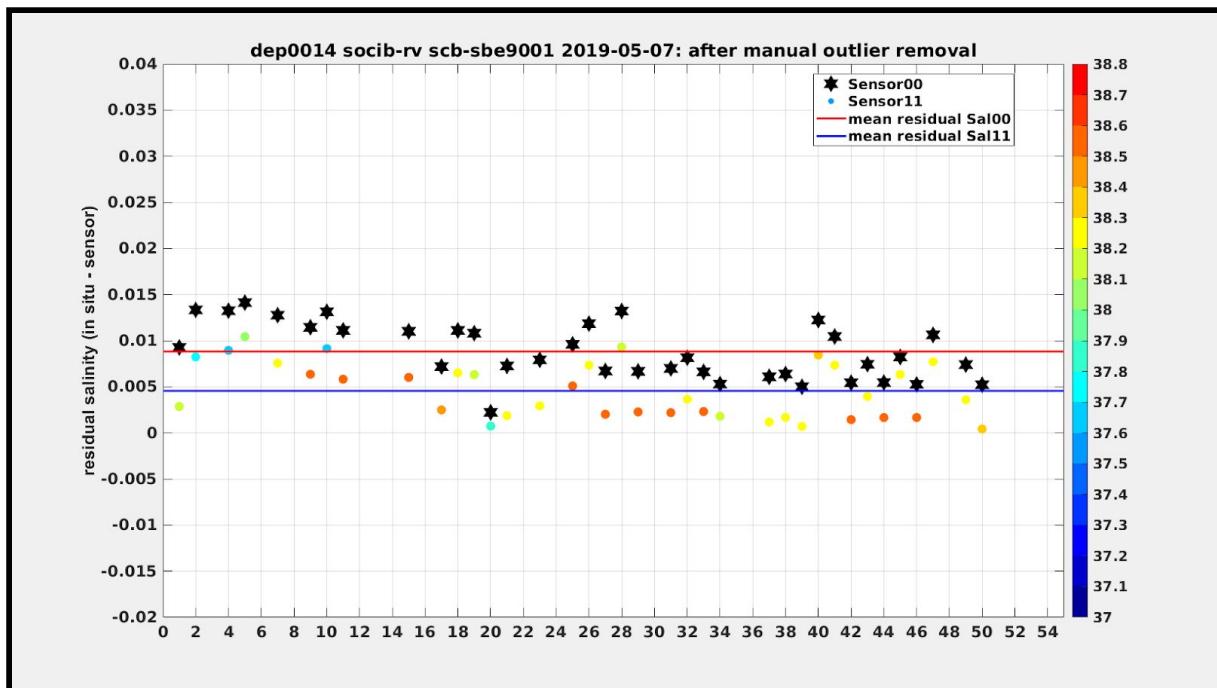
Modules integration -> toolbox, writer and django

configuration -> unify conf file

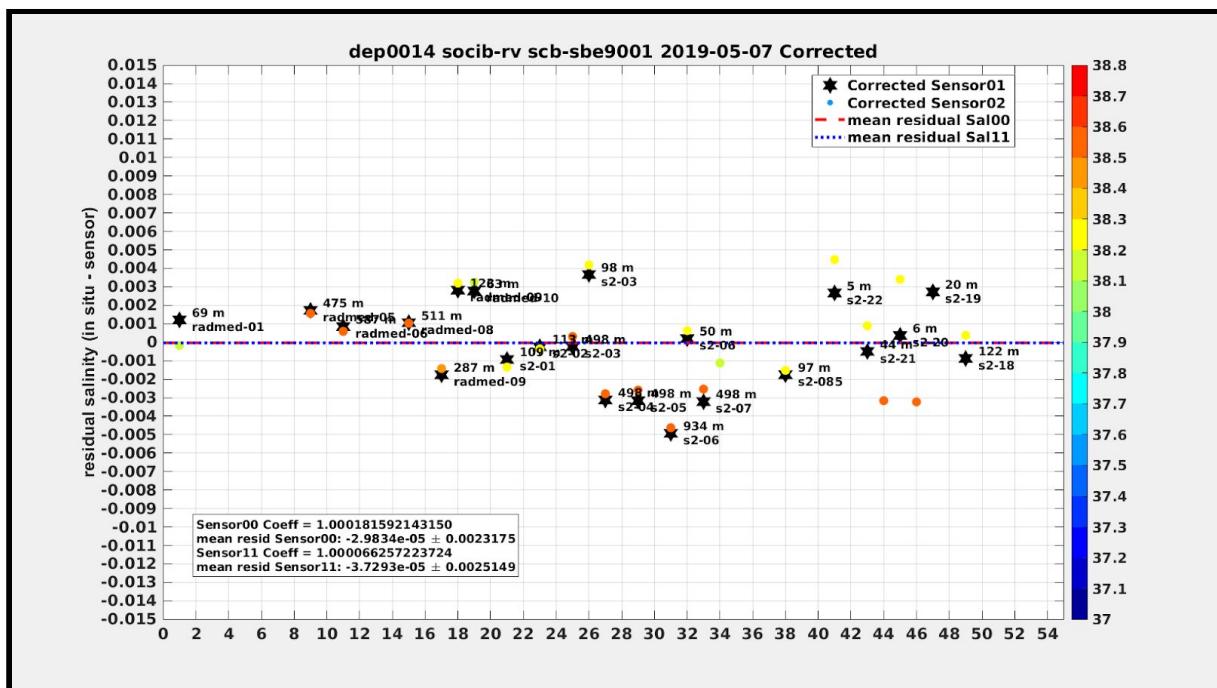
7. APPENDIX A: CTD Correction Pack Figures

7.1. Residuals insitu salinity

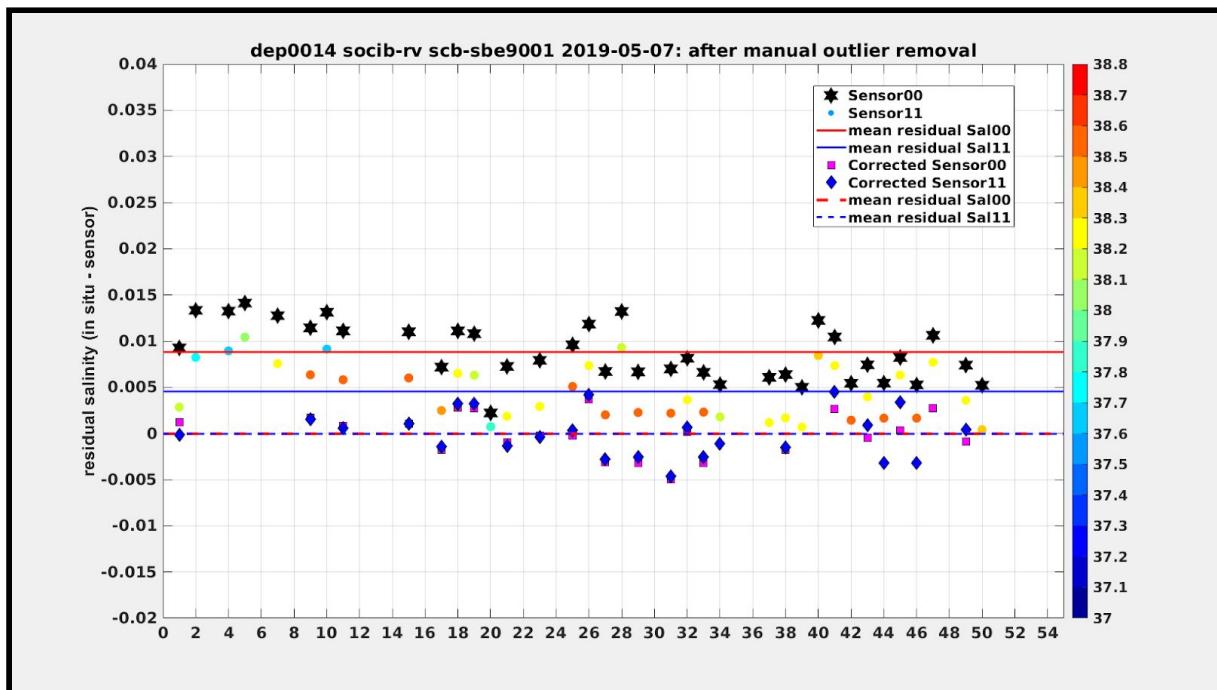




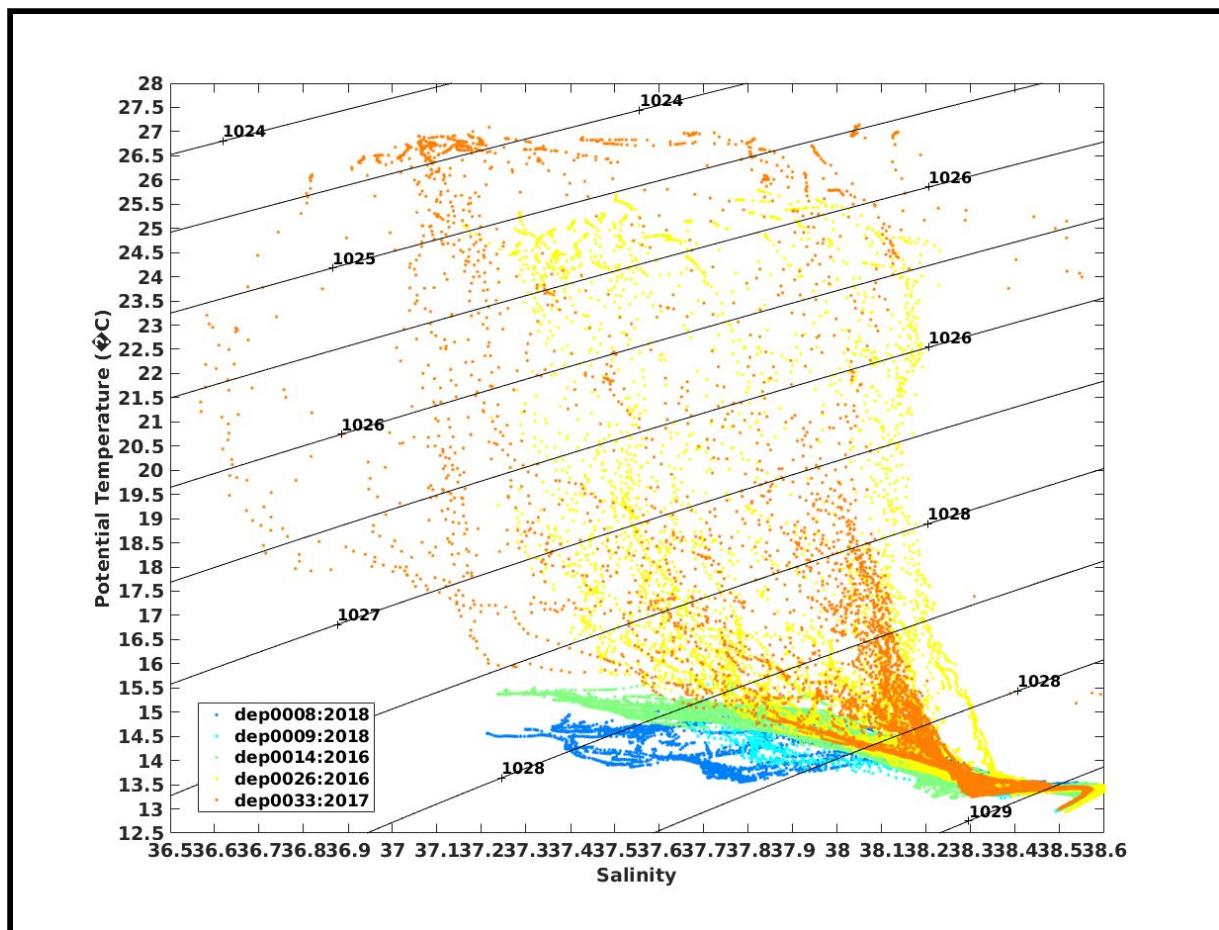
7.2. Theta/S diagrams corrected reference

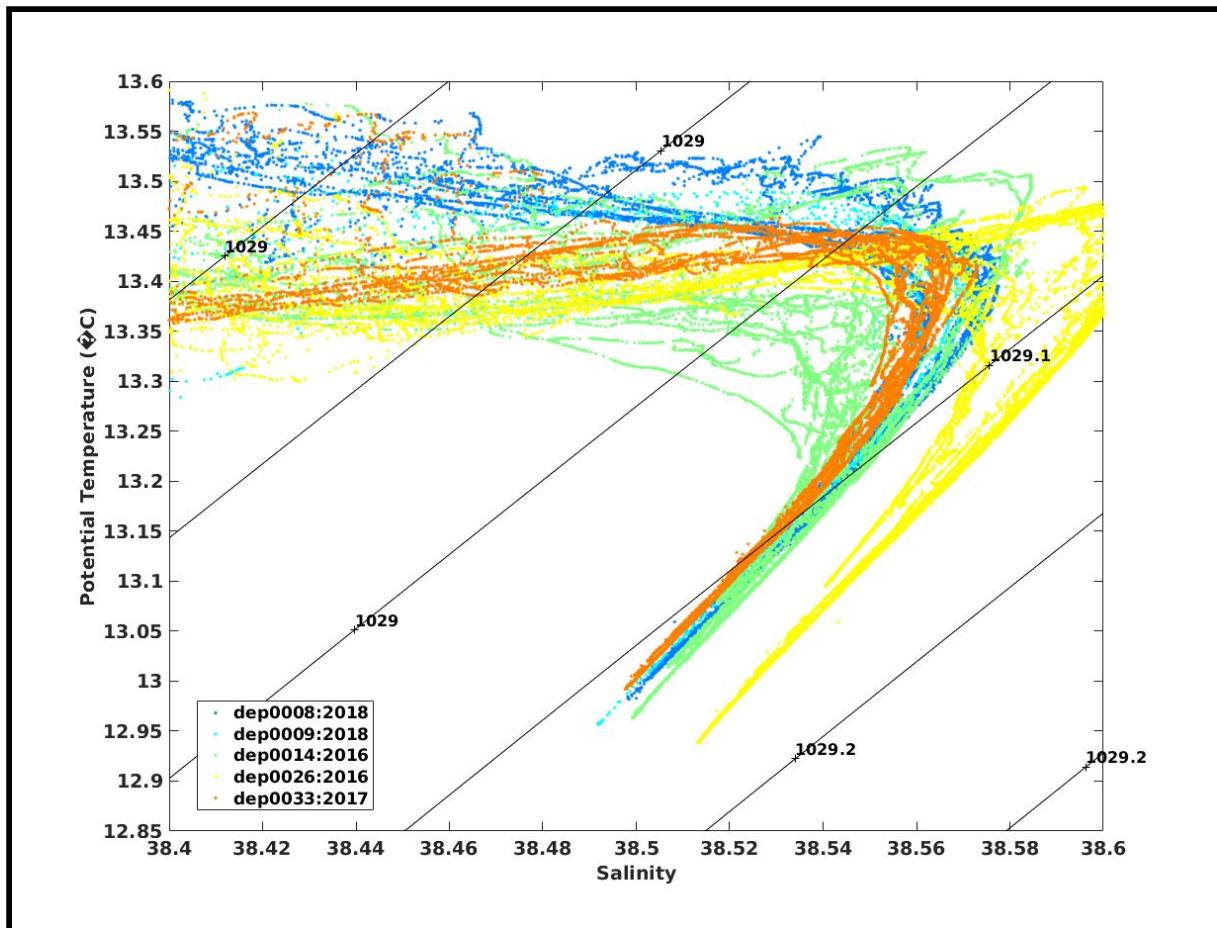


7.3. Theta/S diagrams with and without corrections



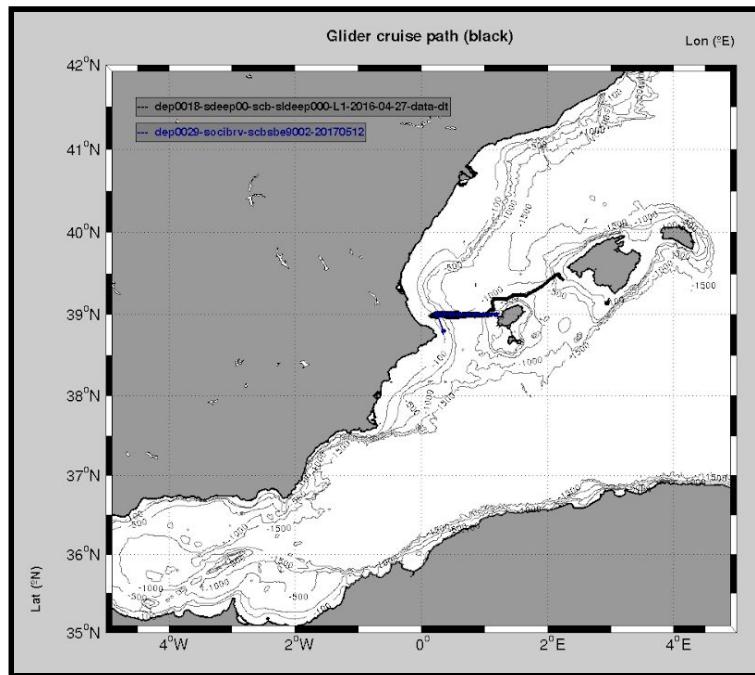
7.4. Theta/S diagrams for all halfm corrected data available



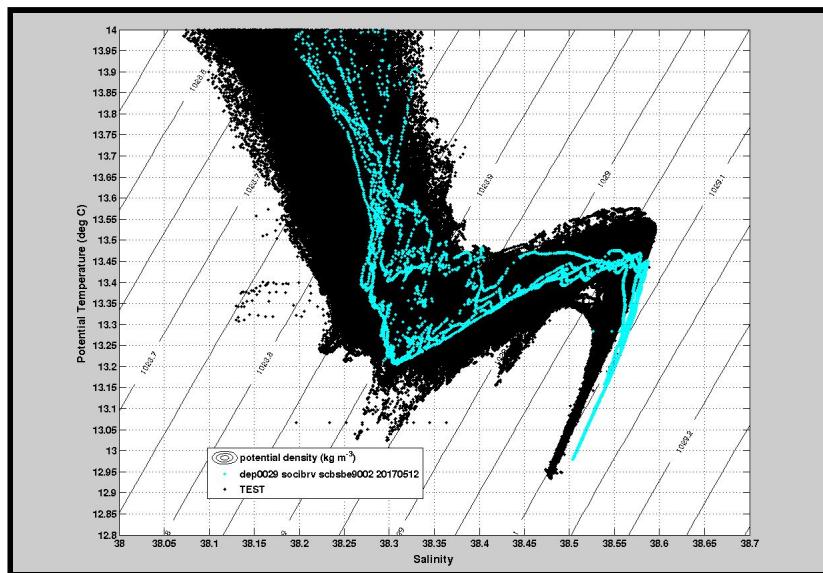


8. APPENDIX B: Glider Correction Pack Figures

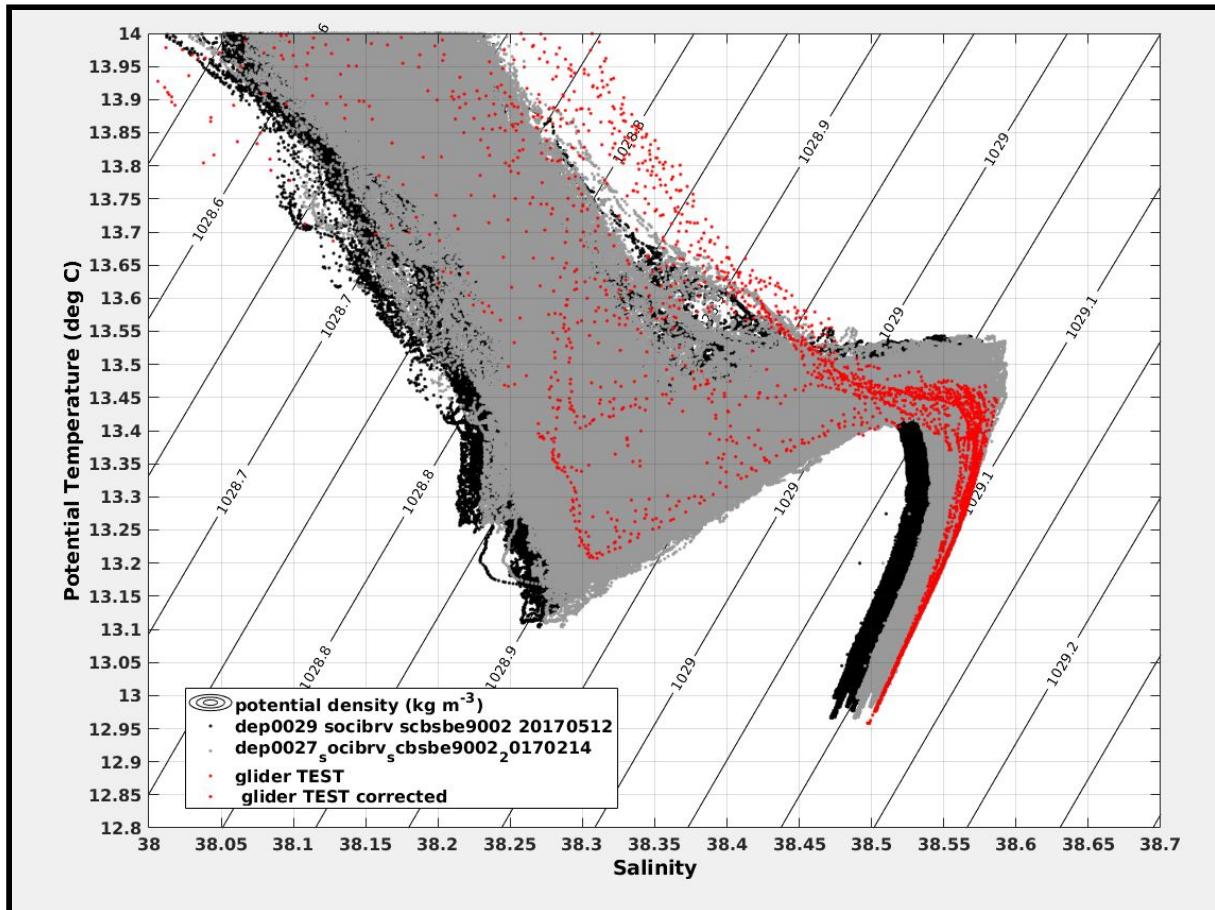
8.1. Cruise location map



8.2. Theta/S diagram halfm corrected vs glider test data



8.3. Theta/S diagram halfm corrected vs glider test data and glider test data corrected



9. APPENDIX C: Argo profiler Correction Pack Figures

9.1. Argo profiler Correction Pack