Procedure for correcting CTD salinity

Draft version 1: 27/10/2015

DATE UPDATE 1: 19/02/2016
DATE UPDATE 2: 17/01/2017

Krissy Reeve

This document quickly outlines the procedure for investigating the quality of the CTD salinity data for all SOCIB ship campaigns.

Stage 1: Stage_1_intercruise_ThetaSdiagram_comparison

PATHNAME for CODES for stage one:
\SHIP\CODES\CTD\CTD_cond_correctionPack\Stage_1_intercruise_ThetaSdiagram_comparison

Download all ship CTD data from thredds (*\MATLAB\SHIP\DATA\CTD\CTD_L1_Thredds*) and import into MATLAB; create T/S diagrams of all stations for each cruise, ensuring the scales and size of the plots match exactly, scaled so that 1 cm on the salinity axis corresponds to 0.01 psu. Print all T/S diagrams. Code for this: (***Stage_1_readData_from_ThreddsCTD_L1.m***)

Commence an inter-comparison between all cruises in order to determine if there may be any offsets from cruise to cruise, by comparing all T/S diagrams and measuring by hand any possible offsets. Create a table listing the offsets between all cruises and the corresponding salinity offset: Cross_cruise_comparison_salinity.xcl). Keep this updated as cruise data becomes available. Keep a copy of this table on google drive.

From this table, we should have an idea of which cruises may require a correction, and which cruises not so much. We can also check the "***CTD_insitu_calibration_history***" spreadsheet to see when the Seabird instrument was last serviced (on Google drive: in \OCEANOGRAPHY\SHIP\PHYSICAL\CTD\Calibration history). Keep this spreadsheet updated after each cruise campaign.

Stages 2 to 6: importing in situ bottle data and Seabird Processed .btl files into Matlab

PATHNAME for CODES for stages 2 to 6:
\SHIP\CODES\CTD\CTD_cond_correctionPack
Stage_2_to_6_import_format_btl_and_sample_data

Stage 2: Copy and paste the sea-bird processed .btl files into the following directory: "*SHIP\DATA\CTD\CTD_btlFILES\btlFILES\ MMMdd_yyyy_CRUISENAME*".

Stage 3: Once the lab work for in situ bottle salinity calculations are available, set up a standardised excel spreadsheet and copy the data over from the portisal spreadsheet, following the correct standardised format. To do this, download the excel template:

"*SHIP\DATA\CTD\CTD_btlFILES\inSitu_btl_sal**template_inSitu_sal.xcl***"

Make sure the station names match the station names listed under the sea-bird processed .btl files from stage 2. Save the file with the correct cruise name, following a standardised format (MMMdd_yyyy_CRUISENAME_inSitu_Sal) under *"\SHIP\DATA\CTD\CTD_btlFILES\inSitu_btl_sal\Standardized*"

Stage 4: Import the above standardized in situ salinity spreadsheet into MATLAB using the following code: "***Stage_4_read_insituSalBTL.m***" located within MATLAB in the directory: "*SHIP\CODES\CTD\ Stage_2_to_6_import_format_btl_and_sample_data*". This creates a structured array of the data which is then saved under the following directory with the same standardised file name as in stage 2: *"\SHIP\DATA\CTD\CTD_btlFILES\inSitu_btl_sal\MATfiles\".*

Stage 5: Next is to find the ".btl" files of the corresponding cruise. Each station of the cruise will have its own ".btl" file. Save these files to a folder using the SAME STANDARDISED CRUISENAME AS ABOVE: "MMMdd_yyyy_CRUISENAME" under the directory within MATLAB: "*SHIP\DATA\CTD\CTD_btlFILES\btlFILES\ MMMdd_yyyy_CRUISENAME*"

Stage 6: We now need to import the relevant data from the .btl file into MATLAB, and into a user-friendly structured array. For this, use the function "***Stage_6a_read_dotbtl_files.m***". If there are multiple cruises that one wants to process, the following code runs the above function for multiple cruises: "***Stage_6b_runbtlFileread_allCruises.m***" in the same directory. These codes create structured arrays of the relevant data, for each cruise, where the first-order fields are the names of each station, and the second-order fields consist of all the relevant data needed for the rest of the procedure. The structured arrays are saved as "***btlDATstruct.mat***" within the corresponding cruise name directory under *"\SHIP\DATA\CTD\CTD_btlFILES\btlFILES*", i.e. the same directory as the original .btl files.

Stages 7 to 13: calculating conductivity correction coefficient and applying correction to derive a corrected salinity

PATHNAME for CODES for stages 7 to 13:
\SHIP\CODES\CTD\CTD_cond_correctionPack
Stage_7_to_13_condCorrection_and_application

Stage 7: We now have two corresponding structured array files for any one cruise: "***MMMdd_yyyy_CRUISENAME_insitu_Sal.mat***" of the lab bottle sample salinity data, and *"...\ MMMdd_yyyy_CRUISENAME**btlDATstruct.mat***" of the .btl CTD files. The next step is to, for each in situ bottle salinity value, extract the corresponding CTD .btl data. This is carried out in *"**Stage_7a_mbtl_insitu_SAL_matchup.m**"*. A new structured array is created ("SALdata") as a result. NOTE: at this stage there may be an error because of "Reference to non-existent field". If this is the case, review which variables are missing. It may be the case that the sea-bird processing code for creating .btl files needs to be re-run. If only one T/S sensor is/are available, then the following function should be activated: "***Stage_7b_mbtl_insitu_SAL_matchup_unique_Sensor.m***". These codes need to be run manually as they need to be altered throughout based on outlier removal; stage 8 below.

Stage 8: Within the same MATLAB code as above, we now investigate how well the CTD conductivity sensors agree with the lab determined salinity of the bottle samples. Firstly, the residuals between the in situ lab salinity and the CTD .btl salinity are plotted for both sensors on a single graph. This is the stage where one should decide and carefully check which outliers are removed. The conditions provided are as follows:

1. Make a decision whether or not to manually remove outliers. The vector "rmX" should be filled with the index of the data points to be removed; this is done by assessing the first figure that is produced, which will be saved as "***SalResid_beforeCorrection…CRUISENAME.png***".
2. Make a decision whether to remove data points based on the comments in the excel spreadsheet of the in situ salinometer analyses, run the corresponding section of code (within "1) MANUAL OUTLIER REMOVAL".
3. Identify as outliers any data points where the difference between the two conductivity sensors (i.e. the difference between sal00 and sal11) is larger than 0.01 (an arbitrarily selected number).
4. Excluding the above outliers, calculate the mean and standard deviation for both sensors. Identify as outliers all residuals that are deviating from the mean by more than one or two standard deviations (start with 2, but if this results in a large standard deviation of corrected salinity residuals, or if the residuals have a large range, it may be necessary to take 1 standard deviation as the condition for outlier removal).
5. Yet again excluding the above outliers, calculate a new mean and standard deviation. Identify as outliers all residuals that are deviating from the mean by more than one or two standard deviations (see above).

Stage 9: After removal of the above outliers, calculate the conductivity calibration equations as follows:

$$conductivity = A * (primary\ conductivity) \quad [1]$$

$$conductivity = B * (secondary\ conductivity) \quad [2]$$

where

$$A = \frac{\sum Cond_{bot} Cond_{ctd}}{\sum (Cond_{ctd})^2} = \frac{\overline{Cond_{bot} Cond_{ctd}}}{\overline{(Cond_{ctd})^2}} \quad [3]$$

and

$$B = \frac{\sum Cond2_{bot}Cond2_{ctd}}{\sum (Cond2_{ctd})^2} = \frac{\overline{Cond2_{bot}Cond2_{ctd}}}{\overline{(Cond2_{ctd})^2}} \quad [4]$$

Note, we have to calculate conductivity of the in situ bottle salinity using the in situ temperature sensors (both of them, hence why there is also $cond2_{bot}$).

The new "corrected conductivities" are then used to calculate the "corrected salinities" of both sensors. The calibration correction coefficients A and B are stored within the structured array "SALdata" along with corrected conductivities and salinities and saved under the directory: '*SHIP\DATA\CTD\CTD_btlFILES\MASHUP*'. The mean and standard deviation of the residuals between in situ salinity and corrected btl salinity are also provided; ideally the correction coefficients should lead to a reduction of the mean to about 0.0000x ± 0.003. This will only be possible with sufficient, good quality data.

Stage 10: Apply corrections to CTD salinity in "***Stage_10a_Ship_CTD_salinity_corrections.m***". This function selects the cruiseName and loads the relevant structured array of correction coefficients previously calculated, and imports the CTD data from the corresponding netcdf file as downloaded from Thredds (and stored under the directory: '*SHIP\DATA\CTD\CTD_L1_Thredds*'). The structured array of the CTD data is created, and a new field "Corrected" is added, which lists the correction coefficients of both sensors, and applies the calibration equations (eq. 1 and 2) to the conductivity data, and also provides the corrected salinity data. This structured array is saved under *"\SHIP\DATA\CTD\CTD_correction_files*\Thredds\" with the same filename as downloaded from thredds with the addition: "with_corrections".

Stage 10-2: the final line of the above code calls a function "***Stage_10a1_matching_TS_diags_with_without_corrections.m***" which takes the above structured array and plots TS diagrams with uncorrected salinity in red and corrected salinity in blue. A second zoomed in T/S diagram is also plotted of the deeper, more stable part of the water column, which can be printed out for cross-cruise comparisons. The .fig and .pdf files are saved under the following directory: "\SHIP\FIGS\TSdiags_with_without_corrections\cruiseName\".

Stage 11: The final stage is to print out the T/S diagrams of all cruises and compare them. This is in the same way as stage 1. Do they all match? Are there any clear outliers? Are there any questionable corrections? This should then be written up in the form of a post-cruise report (This is something on the to-do list!). At this stage, it is necessary to update the excel spreadsheet "***CTD_insitu_calibration_history***" on google drive with the correction coefficients of both sensors.

Stage 12: Now the correction should be applied to the ship CTD 5 m bin averaged data as downloaded by Thredds in NetCDF format. This is carried out in the matlab code: ***Stage_10c_Thredds_Ship_CTD_salinity_corrections.m***, which downloads the corresponding netcdf file of the CTD ship campaign stored under the pathway: `'SHIP\DATA\CTD\CTD_L1_Thredds\'`. It reads in the data, meta data and global meta data of the netcdf file and applies the correction. It creates meta data for the new corrected conductivity and salinity variables and saves as a new NetCDF file under the same filename with the addition of "_Corrected", under the path "*SHIP\DATA\CTD\CTD_correction_files\ncfiles*".

Stage 13: Lastly, a netcdf file of just the meta data for the corrected conductivity and corrected rederived salinity variables are created and provided to the data centre. This is for the data centre to incorporate into their data system. This code is called: ***Stage_13_create_nc_correction_file_for_dataCentre.m*** and saves the netcdf file to the path: '*SHIP\DATA\CTD\CTD_correction_files\Correction_meta_details*', and then copies this file onto the network: \\POSEIDON\users\vessel\SHIP_DATA\SBE911\Corrected__netcdfFiles_CTD\meta_data_correction_coefficients\
And the data centre should be alerted when it is available on the network. The code calls a nested function: '*\SHIP\CODES****depNum_to_cruiseName.m***' which takes the year, month and cruise name from the mat file of the correction data and matches it to a deployment number assigned by the data centre. This became necessary because the correction procedure works from the .btl files directly output from the Seabird Processing software before the data centre have assigned a deployment number for the corresponding campaign. The .btl files are saved as whatever cruise name the EDT use on-board the RV Socib. A suggestion for improving the connection between the Seabird processing data and the data centre could be for the data to assign a deployment number ***before*** each campaign, which then becomes the first part of the file-name in SeaBird. Codes could then be edited to become simpler, and there becomes less opportunity for errors.

Stage 14: applying correction to half metre bin CTD data

PATHNAME for CODES for stage 14:
\SHIP\CODES\CTD\CTD_cond_correctionPack
Stage_14_APPLY_to_halfMetreBIN_data

In this section, we apply the conductivity and salinity corrections to the half metre bin averaged ship CTD data. This is required for scientific analysis of the data but also for using the data to correct the glider salinity using the Theta-S whitespace maximisation procedure in the subsequent work package.

Stage 14: ***Stage_14_read_and_correct_halfm_cnv_ctd.m***

This code asks the user to select a cruise campaign, either from the internal network, from the users document drive, or allows the user to manually navigate to where the Seabird half metre binned .cnv data are stored. The default is the internal network, from which users must navigate to the half metre bin folder from here:

'\\poseidon\users\vessel\RTDATA\socib_rv',

then, for example, from here, navigate to:

\SCB-SBE9002\rawArchive\canales_dep0012_socib-rv_scb-sbe9002_L1_2015-07-11\PROCESSED_SOCIB_halfm\'

The code then downloads the meta data for the corrected variables created in stage 13 above, from:

“\\POSEIDON\users\vessel\SHIP_DATA\SBE911\Corrected__netcdfFiles_CTD\meta_data_correction_coefficients\”

It then reads the .cnv files into matlab by calling a nested function: ***Stage_14b_cnv2mat_2013.m***, and applies the correction coefficients to the conductivity variables (both sensors if both are available) and then re-derives a corrected salinity. The corrected half metre binned data are then saved as a .mat file in:

'SHIP\DATA\CTD\CTD_Correction_files\halfmetreBIN\matfiles\'

The next code loads all corrected half metre data of all campaigns into one structured array. It is important to run this code after each correction procedure as the output mat file is used for the background data when correcting glider data. The code, ***Stage_14c_allDATA_halfm_corrected.m***, checks if the mat file is already existing:

*SHIP\DATA\CTD\CTD_correction_files**SHIP_allDATA_halfm_corrected.mat***

If the file exists, it is imported into matlab. The code then checks the data within the structure against the corrected half metre files in:

“SHIP\DATA\CTD\CTD_correction_files\halfmetreBIN\matfiles\”

For any missing data, the code simply imports that data from the above path as another field within the structured array. The output file is saved with the same file name as previously: ***SHIP_allDATA_halfm_corrected.mat.***

Finally, meta data are added to the corrected half metre data and both are saved in netcdf format, using the global_meta and meta data from the Thredds 5m corrected binned data from stage 12. This is carried out in:

Stage_14d_add_meta_create_nc_halfmetre_correction_CTD.m.

The resulting netcdf files are saved here:

SHIP\DATA\CTD\CTD_correction_files\halfmetreBIN\ncfiles

There is now an option to create a reference Theta-S diagram of all half metre binned corrected data, by running the code ***Stage_14e_TSdiag_all_halfMetreBIN_corrected.m***; this creates a single corrected Theta-S diagram of all ship data, for reference purposes (can also see the gradual increase in salinity with time). The second plot is zoomed into the deepest part of the water column to highlight any long-term change in salinity, which appears to be occurring at a significant rate. The third is created without setting the axes limits and helps to highlight when we have an issue with a sensor (i.e. Meddess Station 12, sensor sal00). The whole and zoomed in Theta-S diagrams are saved as png files under the pathname:

SHIP\FIGS\TSdiag_corrected_Reference\halfmetre_all\Theta_S_shipCorrected-DATE_whole.png

And

SHIP\FIGS\TSdiag_corrected_Reference\halfmetre_all\Theta_S_shipCorrected-DATE_ZOOM.png

The correction process for Salinity CTD data are now complete and ready to be implemented in glider salinity correction.

NOW WE’RE READY FOR 1. O2 CTD CORRECTION AND 2. GLIDER SALINITY CORRECTION!!!