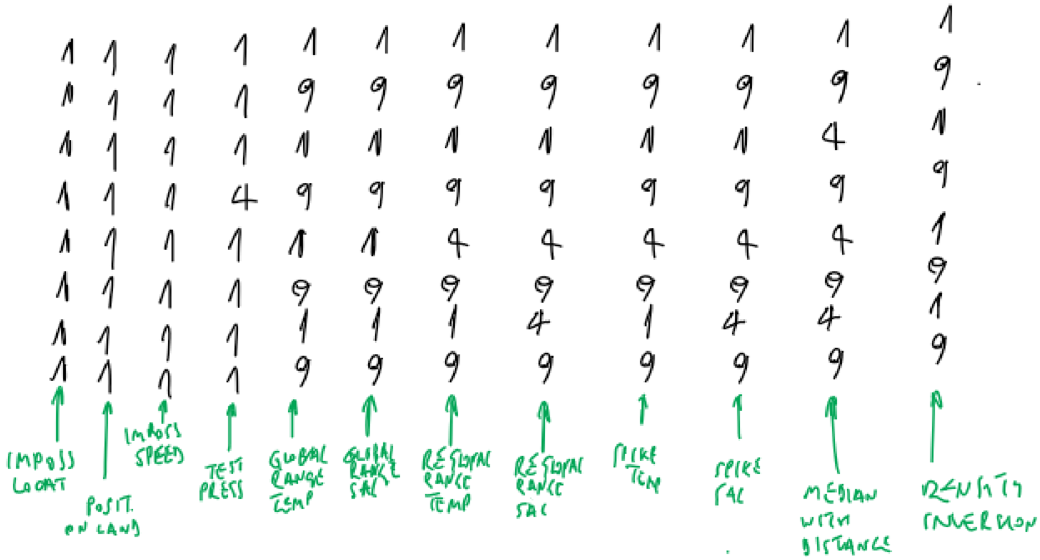
National Institute of Oceanography and Applied Geophysics



OGS GLIDER DATA QUALITY CONTROL

Gerin R.

DOI: 10.6092/ea0d979a-4043-4cf1-9397-6e097c315591

INDEX

1. Introduction

SeaGliders (SG554 and SG661) and Slocum gliders (unit 402 and unit 403) send different kinds of raw data to the dedicated Dockserver at each surfacings. There are two different Dockservers, one for the SeaGliders and one for the Slocum gliders. The Dockservers automatically convert the received raw data in scientific/engineering values and save them in *.nc or *.mat file in the case of the SeaGlider and Slocum gliders, respectively (one file per dive). In the case of the glider missions performed with the SeaGlider SG554 before 2015, the Dockserver did not supported *.nc files at that time and data were saved in *.eng files. Their elaboration will be discussed in a dedicated chapter (chapter 2.9).

The converted files contain the data collected during a single dive[1]. The variables inside the files are different and depend on the type of glider, on its firmware and on the mounted sensors.

The first step in the processing and quality control (QC) of the data is to create a common file to start with. This file should be a single file (one per mission) and should include the same variables regardless of the type of glider. This is feasible except when different sensors are used. In general, the file should contain only the variables measured during the mission.

2. Data preparation

To easily manage the data, the variables are built as vectors having the same length. Since the sensors have different sampling frequency and since the acquisition timing and the reference time of the technical and navigation parameters (and of the oxygen and optical sensors for the SeaGlider) differ from that of the scientific parameters (see Table 1), the union of the the time values is considered and then data are sorted.

If no time correspondence is found for a specific parameter, its value is considered as a missing value and reported as NaN.

[1] a dive is considered as the operations performed underwater by the glider between two consecutive surfacings. It may include one or more yos.

A single *.mat file including all the variables reported in Table 2 is generated for each mission. Table 3 reports the correspondence between the depth and the other variables listed in Table 2.

Glider	time parameter	depth parameter	other parameters
unit 402 unit 403	sci_m_present_time	sci_water_pressure	sci_water_temp, sci_water_cond
	m_present_time	m_depth	m_lon, m_lat, m_pitch, m_de_oil_vol
	sci_m_present_time	sci_water_pressure	sci_oxy4_calphase, sci_oxy4_temp
			sci_flbbcd_chlor_units, sci_flbbcd_cdom_units, sci_flbbcd_bb_units
SG554 between 2015 and 2018	ctd_time	ctd_depth	temperature_raw, conductivity_raw, longitude, latitude
	time	depth	eng_pitchAng, eng_vbdCC
			eng_aa4330_CalPhase, eng_aa4330_Temp
			eng_wlbb2fl_FL1sig, eng_wlbb2fl_BB1sig, eng_wlbb2fl_BB2sig
SG554 before 2018	ctd_time	ctd_depth	temperature_raw, conductivity_raw, longitude, latitude
	time	depth	eng_pitchAng, eng_vbdCC
			eng_aa4330_CalPhase, eng_aa4330_Temp
			wlbbfl2_FL1sig, wlbbfl2_FL2sig, wlbbfl2_BB1sig
SG661	ctd_time	ctd_depth	temperature_raw, conductivity_raw, longitude, latitude
	time	depth	eng_pitchAng, eng_vbdCC
	aa4831_time	interpolated from ctd_depth	aa4831_CalPhase, aa4831_Temp
	wlbbfl2_time		wlbbfl2_FL1sig, wlbbfl2_FL2sig, wlbbfl2_BB1sig

Table 2. Glider parameters as in the *.nc or *.mat converted files.

Parameter	Variable name (all gliders)	Variable name (unit 402, unit 403, SG554 after 2018, SG661)	Variable name (SG554 before 2018)
time (Matlab date; dd.ddd)	Time		
reference depth for the CTD (m)	DepthCTD		
reference depth for the Navigation parameters (LON, LAT, PITCH, OIL) (m)	DepthNav		
reference depth for the Aanderaa (oxygen) sensor (m)	DepthOXY		
reference depth for the WetLabs sensor (m)	DepthWET		
number of Dive	DIVE		
longitude (DDD.DDDDD°)	LON		
latitude (DDD.DDDDD°)	LAT		
pitch (positive = pitch up; Rad)	PITCH		
moved oil volume (positive = positive buoyancy; CC)	OIL		
temperature (°C)	TEMP		
salinity (eventually thermal lag corrected)	SAL		
potential density	PDENS		
oxygen compensated for pressure and salinity (ml/l)	OXY		
oxygen saturation (%)	OXY_SAT		
chlorophyll (μg/l)	CHL		
CDOM (ppb)		CDOM	
backscattering at 470 nm (m^{-1} sr^{-1})			BB470
backscattering at 700 nm (m^{-1} sr^{-1})	BB700		

Table 2. Glider variable names in the final *.mat file.

time variable	depth variable	other parameters
Time	DepthCTD	TEMP, SAL, PDENS
	DepthNav	PITCH, OIL, LON, LAT
	DepthOXY	OXY, OXY_SAT
	DepthWET	CHL, CDOM, BB470, BB700

Table 3. Glider variable correspondence in the final *.mat file.

2.1. Longitude and Latitude

Slocum gliders relate the position variables to the time and depth recorded by the navigation system, while the SeaGlider relates them to the scientific system. Being the GPS fix acquired at surface only and then interpolated by the glider at specific time stamp, and being the longitude and latitude two parameters related to the navigation, it results much more logical to relate them to the time and depth of the navigation system (Table 3). Therefore, the longitude and the latitude as collected by the SeaGlider are interpolated at the navigation time and depth without introducing any kind of error.

2.2. Depth and pressure

Generally, we selected the depth as the vertical coordinate (see Table 1). The Slocum glider reports the pressure instead of the depth for the scientific parameters (the sci_water_depth does not exist). The comparison of the m_depth and m_pressure navigation parameters unexpectedly reveals that the two variables (depth and pressure) display the same values and hence at the factory they (wrongly) have been considered interchangeable. We can speculate that the same line of reasoning has been adopted at the factory while saving the scientific pressure parameter (sci_water_pressure) and that the pressure to depth conversion is not convenient. Indeed, the conversion of the scientific pressure to scientific depth considering the Stevino law ($p=\rho gh$) generates higher differences between the scientific and the navigation parameters and with different sign with respect to the

SeaGlider-case differences (considering similar accuracy for the Slocum and SeaGlider navigation and scientific pressure sensors; compare Figs. 1 and 2).

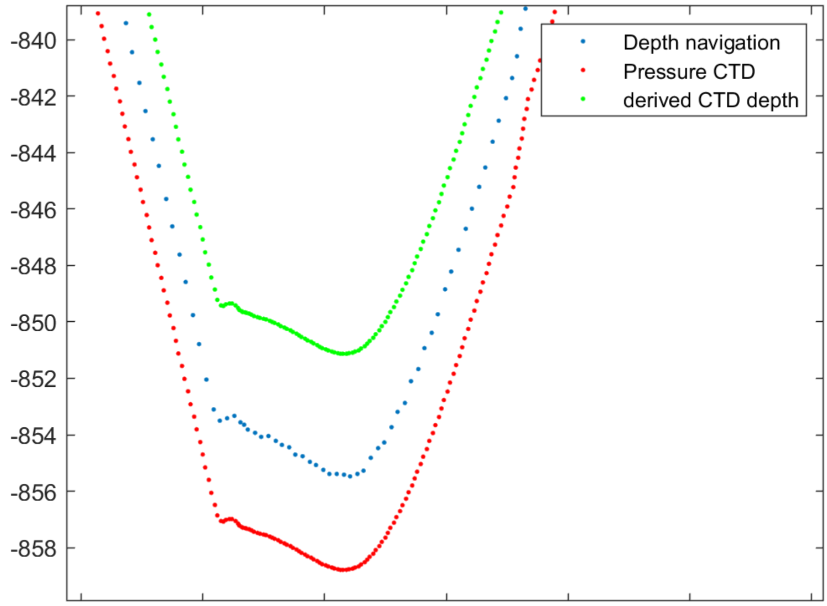


Fig. 1. Depths and pressure of the Slocum glider.

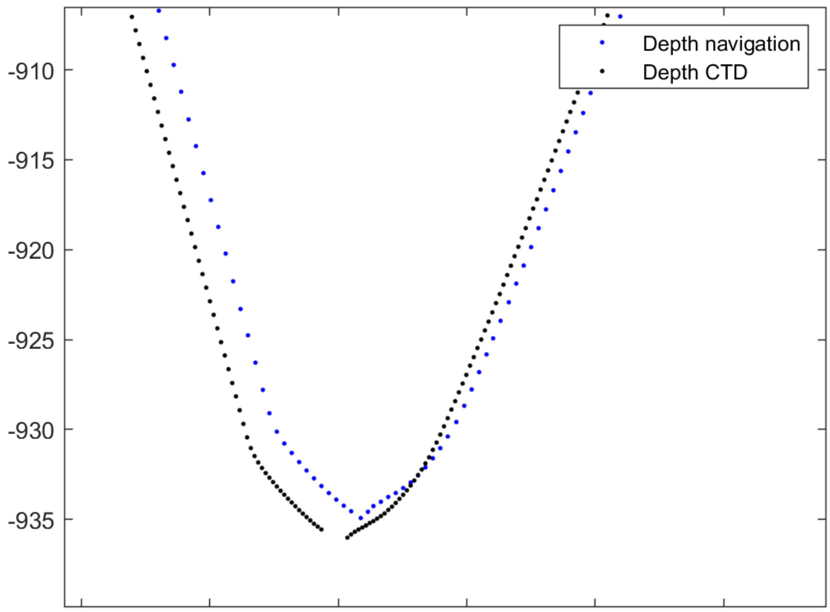


Fig. 2. Depths of the SeaGlider.

2.3. Distance between pressurometers

The Slocum and SeaGlider gliders are equipped with two pressure sensors used for the navigation and for the CTD (see also Table 1). All the scientific data of the Slocum glider are referred to the scientific pressure sensor, while the oxygen and the optical parameters of the SeaGliders are referred to the navigation pressure sensor or have no associated pressure measurement as is the case of the SG661.

All the scientific data are referred to their time of acquisition. If one sensor is mounted at a certain distance from the pressure sensor, its depth of acquisition differs from the one measured by the pressure sensor at the time of the acquisition. This depth can be computed considering the distance between the pressure sensors and the different sensors (CTD, oxygen and optical parameters; Fig. 3 and Fig. 4), the pitch and the mean speed of the glider.

For the sake of clarity, if one CTD sample is taken at t_n and z_n (with z corresponding to the depth) and one oxygen measurement is collected at t_{n+1}, the depth of the oxygen measurement z_{n+1} can be obtained considering the vertical speed of the glider

$$w=dz/dt$$

together with the vertical projection of the distance D between the pressure sensor and the oxygen sensor

$$dz_{oxy}= D \sin(pitch)$$

from the previous, it is possible to derive the delay of the acquisition

$$delay= dz_{oxy}/w.$$

This delay in time can be easily converted into a gap in depth. For example by interpolating the CTD depth at the $(t_{n+1} + delay_{n+1})$ time

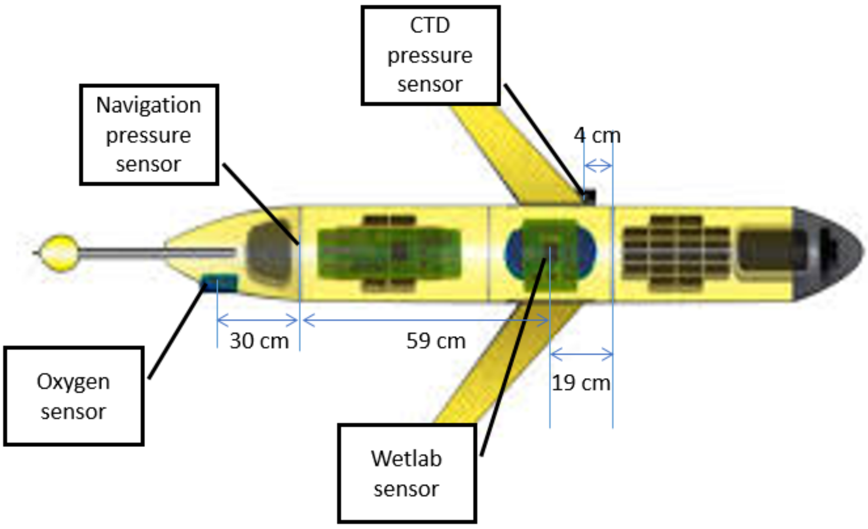


Fig. 3. Distances between the pressure sensors and the scientific sensors for the Slocum glider.

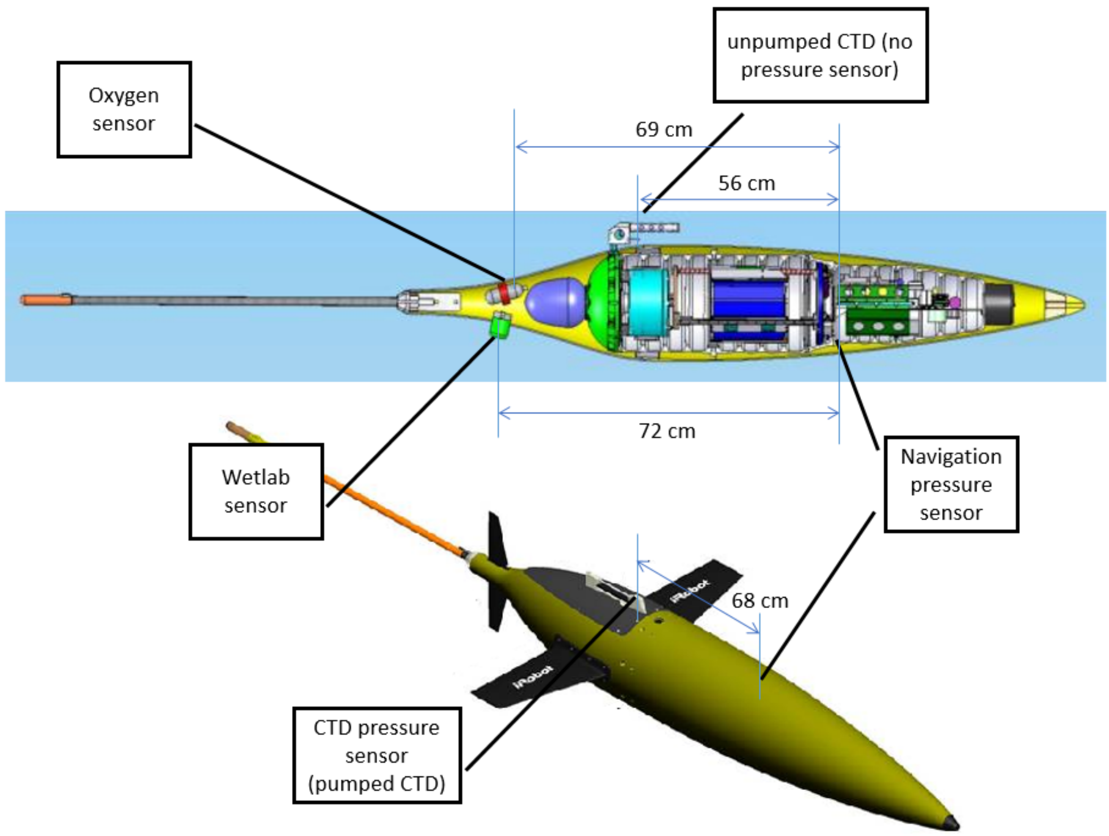


Fig. 4. Same as Fig. 1 but for the Seaglider; unpumped (upper panel) and pumped CTD (lower panel).

2.4. Thermal lag correction

The Slocum gliders and the SeaGlider SG554 are equipped with pumped CTD, while the Seaglider SG661 is equipped with the unpumped CT. Because temperature sensors are located outside the conductivity cell, the temperature reported by the CTD will be slightly different from the actual temperature inside the conductivity cell. Therefore, when those measurements of temperature and conductivity are used in the salinity equation, the computed salinity will be erroneous, especially when crossing strong temperature gradients (thermocline). This issue is known as the thermal lag effect and affects especially the unpumped CTD systems (Garau et al., 2011 and Gerin, 2013).

The difference between the corrected and the original pumped CTD data is about one order less with respect to the unpumped CT data even in case of strong thermocline (see Figs. 5 and 6). The Garau et al. (2011) method was applied to the unpumped CT data only.

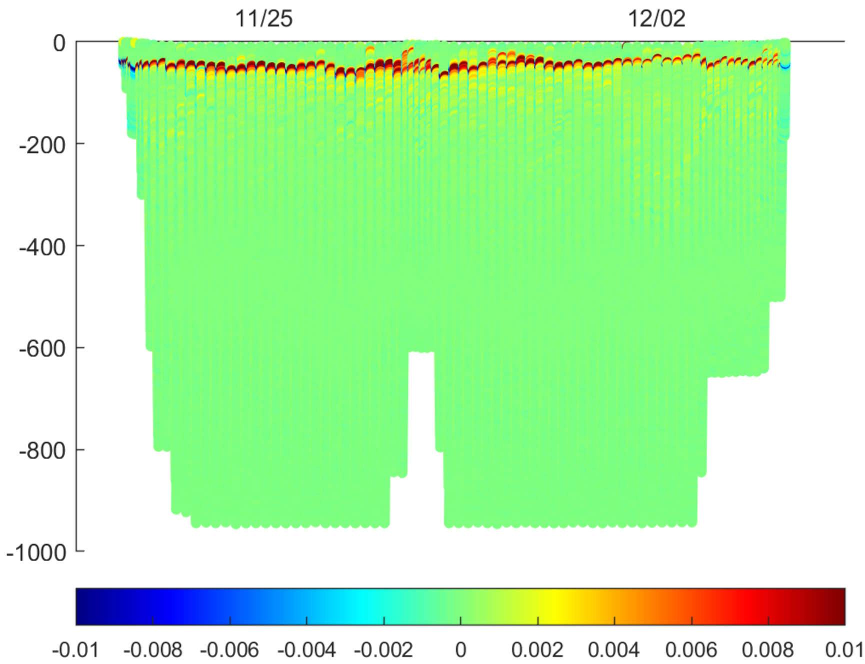
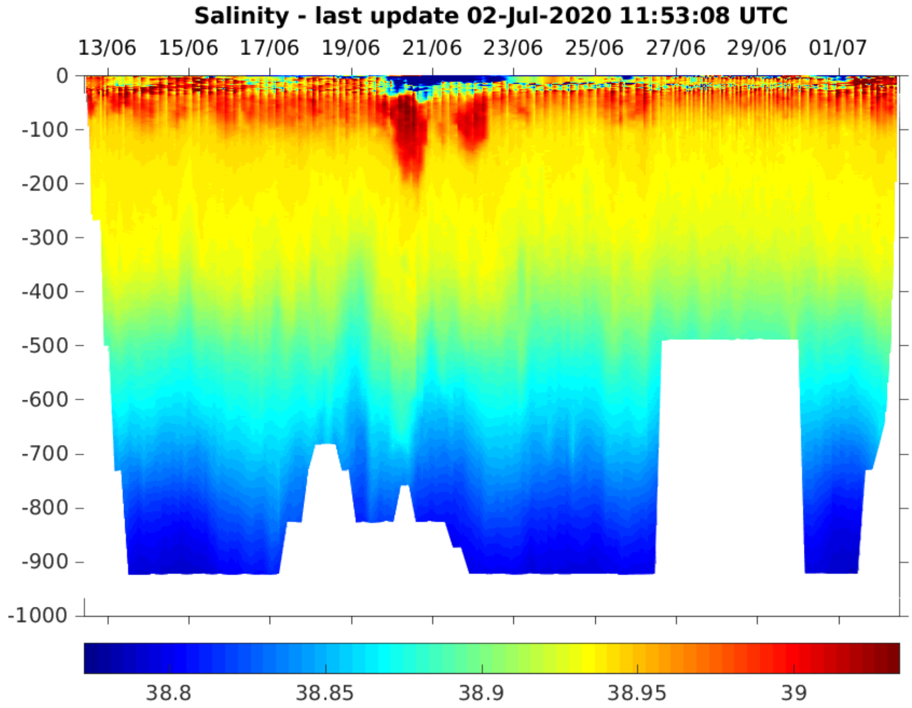


Fig. 5. Salinity difference before and after the Thermal Lag Correction during the PreConvex19 mission performed with the SeaGlider SG661 equipped with an unpumped CT.



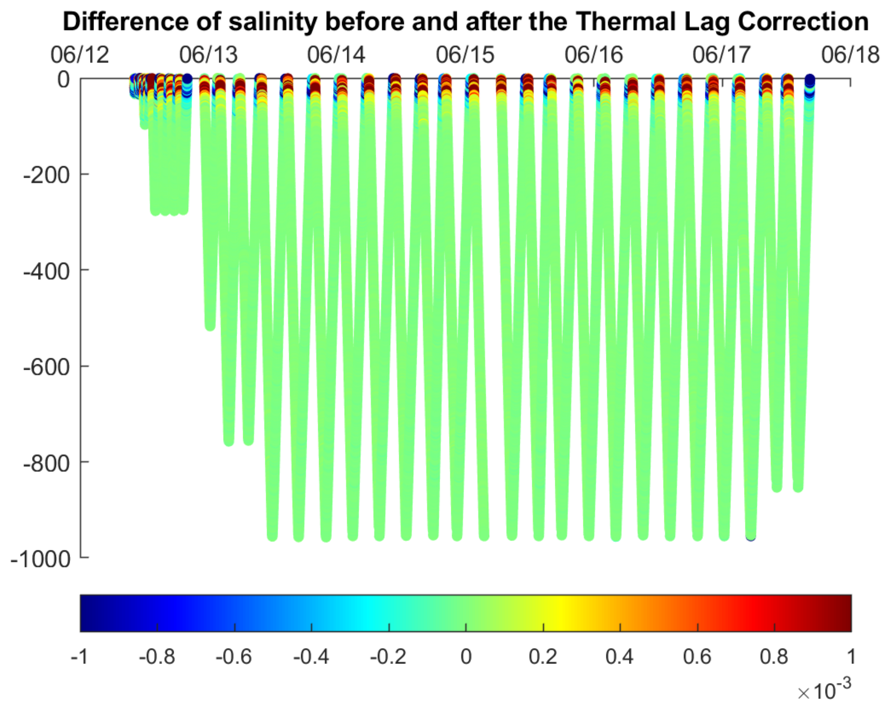


Fig. 6. Salinity during the Convex20 glider mission (top) performed with the SeaGlider SG554 equipped with a pumped CTD and difference of salinity before and after the Thermal Lag Correction during the first 5 days of the mission (bottom).

2.5. Oxygen compensation

The oxygen glider data recorded by the Slocum and SeaGlider gliders have been deeply investigated by Gerin et al. (2020), the oxygen values measured by the different gliders differs noticeably due to the fact that the salinity reference used in the oxygen computation and set in the sensor/glider by the glider factory is different (35 and 0 for the Slocum and SeaGlider gliders, respectively). Additionally, the Aanderaa Optode oxygen sensor is influenced by temperature, salinity and pressure that modify the hardware response of the sensor and compensation is needed.

The computation of the glider oxygen data from the raw Callphase parameter and the compensation procedure are detailed in Gerin et al. (2020). The thermal lag corrected salinity is considered for the compensation of the unpumped glider.

At this stage, the glider oxygen data are not “calibrated” with Winkler samples or Winkler calibrated oxygen data recorded by floats. This “calibration” could be performed later following the method described in Gerin et al. (2020).

2.6. WetLabs dark count correction

The WetLabs optical data are dark count corrected. In particular, data are converted in counts reversing the formula:

$$parameter = count * SF - DC$$

(with *SF* scale factor and *DC* dark count coefficients indicated in the calibration sheet) and the minimum of the counts recorded below the depth *R* is considered as the new *DC* value and the parameter values are computed again. The negative values are discarded.

$$R = 0.8 * max\ (\ Depth\ )$$

2.7. The matlab script

The data preparation is assured by a matlab script named *RG_generate_variables* (stored on Cayman at /storage/sire/work/glider/RG_generate_unique_file)

Example: *RG_generate_variables('preconvex19_full','SG661',1, 15, 15, NaN);*

It requires as input:

- the mission name (all the mission files must be included in a folder with this name at the same level of the script);
- the name of the glider (SG554, SG554old (glider configuration used before 2018), SG661, unit402 or unit 403);
- a flag for the thermal lag correction (0 or 1 to avoid/perform the correction);
- a time limit (used in the thermal lag correction) to exclude profiles with a high time gap (meaning that they do not profile the same water mass; suggested value 15);
- the first profile to consider (used in the thermal lag correction; useful to exclude some shallow profiles at the beginning which were performed to trim the glider);
- the last profile to consider (used in the thermal lag correction; useful to exclude some profiles at the end of the mission which were performed to recover the instrument; NaN to consider the last profile).

The script saves the variables listed in Table 2 in a file named *mission_name.mat* and requires the following additional functions/libraries:

- *RG_compute_TLC_parameters*;
- *RG_thermal_lag*;
- GliderToolbox (https://github.com/socib/glider_toolbox);
- seawater_ver3_3 (http://www.cmar.csiro.au/datacentre/ext_docs/seawater.htm);
- *ZKRG_oxygen_calculation_4330_4831_DAC_oxy_sat*.

Additionally, the script requires a file containing the distinctive coefficients of the Aanderaa oxygen foil (Fig. 7). These coefficients can be retrieved as explained in Gerin et al. (2020).

```
ConcCoefa = 6.200611e-1;
ConcCoefb = 1.018849;
% FoilCoefA
FCa=[-2.988314E-06    -6.137785E-06    1.684659E-03    -1.857173E-01    6.784399E-04    -5.597908E-07    1.040158E+01    -
5.986907E-02    1.360425E-04    -4.776977E-07    -3.032937E+02    2.530496E+00    -1.267045E-02    1.040454E-04];
% FoilCoefB
FCb=[-3.560390E-07    3.816713E+03    -4.475507E+01    4.386163E-01    -7.146342E-03    8.906235E-05    -6.343012E-07
0.000000E+00    0.000000E+00    0.000000E+00    0.000000E+00    0.000000E+00    0.000000E+00    0.000000E+00];
% FoilPolyDegT
FPDT=[1    0    0    0    1    2    0    1    2    3    0    1    2    3    4
0    1    2    3    4    5    0    0    0    0    0    0    0];
% FoilPolyDegO
FPDO=[4    5    4    3    3    3    2    2    2    2    1    1    1    1    1
0    0    0    0    0    0    0    0    0    0    0    0    0];
```

Fig. 7. Aanderaa foil coefficients for the Slocum gliders (unit 402 and unit 403 have the same foil).

2.8. Anomalous mission data

The Cinel17 and Melmas18 missions are irregular missions. Some malfunctions occurred before/during the mission affecting the collected data. The standard procedure described in this report is not sufficient to create the mission file with the variables of Table 2 from which the QC procedure starts.

In the Cinel17 mission, the glider WetLabs sensor was not functioning properly due to some electrical issues (Gerin et al., 2017; Zuppelli et al., 2017) and was substituted with a different WetLabs sensor of the University of Cyprus (UCY, provided by Dan Hayes) having a different wiring and different coefficients. The Cinel17 WetLabs data are adjusted after running the *RG_generate_variables* script by using the *RG_replace_wetlab_Cinel17* script which recall the *all_wetlab_Dan_dark_count.mat* file containing the manually generated and dark count corrected WetLabs data.

During the Melmas18 glider mission, the GPCTD (pumped CTD) did not work properly . The collected down- and up-cast temperature and salinity profiles display high differences especially in the first 500 m (Gerin and Pacciaroni, 2019). After the glider recovery, the CTD pump was inspected and the magnet of the impeller shaft was found broken (Gerin and Pacciaroni, 2019). The malfunction does not allow for any data correction, therefore we choose to perform a mean of the data inside two times the standard deviation. After the *RG_generate_variables* script the *RG_average_Melmas18_data* script is used to average the data. This last script requires the *RG_mean_overlap50_xSTD* function.

Melmas19 mission also presents some anomalies. Dive 237 was not recorded by the glider and during dives 187, 260 and 332, the instrument saved the DepthCTD only during the ascent phase. Additionally, the DepthCTD parameter in these three dives seems to be incremented by 48.1, 64.8 and 78.9 meters (Fig. 7). The DepthCTD was adjusted accordingly.

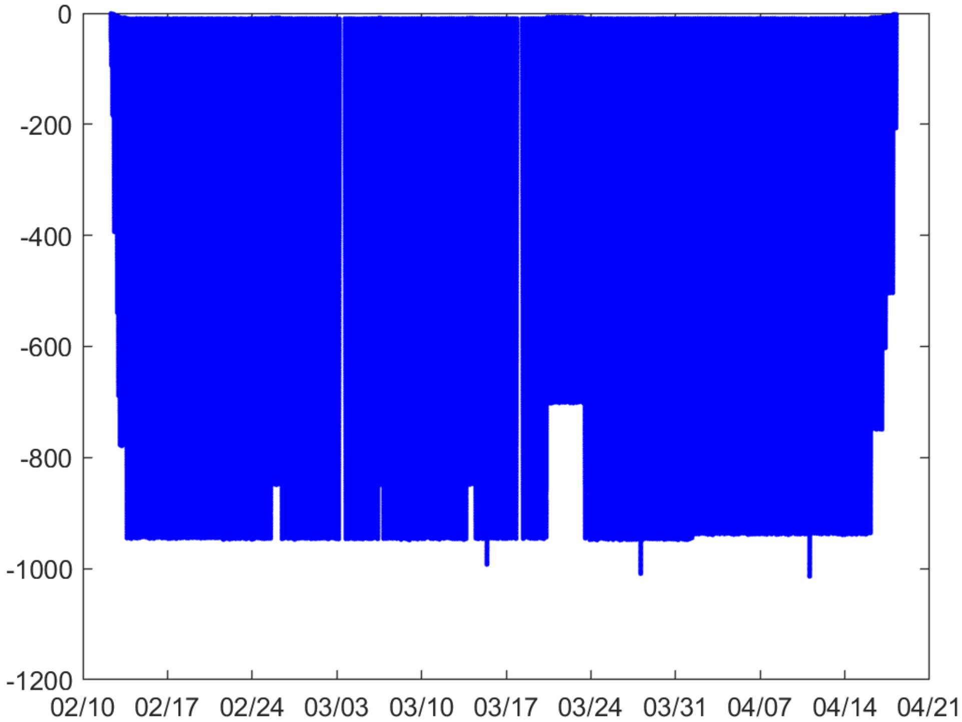


Fig. 7. DepthCTD measured during the Melmas19 mission.

Cinel17 mission displays a few wrong DepthCTD data (Fig. 8). Indeed, the glider seems to report the maximum depth reached during a dive also at the surface. These data were manually removed from the dataset.

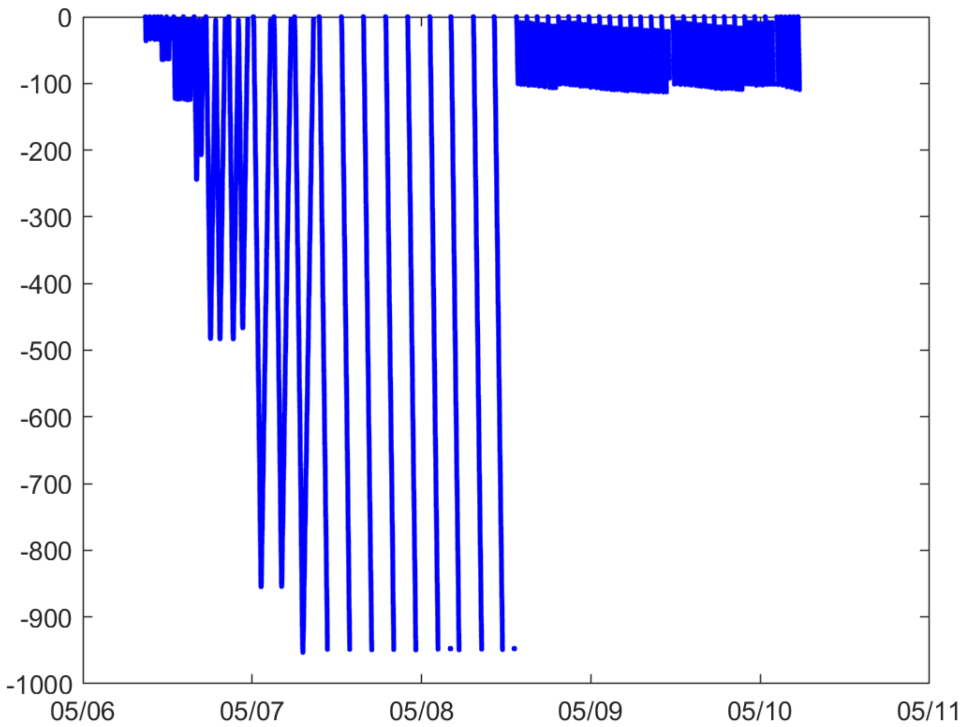


Fig. 8. DepthCTD measured during the Convex17 mission.

2.9. SG554 data before 2015

The glider missions performed with the SeaGlider SG554 before 2015 (COCONET and CONVEX14) are processed using different script (*RG_generate_variables_from_raw*) because the *.nc files were not supported by the Dockserver at that time. The script load the raw data from the *.eng files and get the variable needed for the computation from the header and the columns of the files (see the correspondence in Table 5). The longitude and the latitude are not included in the *.eng file and are retrieved from the dati_log.txt file which resumes the information of the p*.log files. The position measured at surface is then interpolated at the Navigation time.

SG554 before 2015	Time CTD	from the header of the ppc*.eng
	Depth CTD	first column of ppc*.eng file
	temperature	second column of the ppc*.eng file
	conductivity	third column of the ppc*.eng file
	TimeNAV	from the header + second column of the p*.eng file
	DepthNAV	third column of the p*.eng file
	Longitude & Latitude	from the dati_log.txt file
	pitch	column 5 of the p*.eng file
	roll	column 6 of the p*.eng file
	oil	column 9 of the p*.eng file
	bb470	column 13 of the p*.eng file
	bb700	column 15 of the p*.eng file
	chl	column 17 of the p*.eng file
	temperature of the oxygen sensor	column 21 of the p*.eng file
	calphase	column 22 of the p*.eng file

Table 4. SG554 glider variable correspondence before 2015.

2.10. Glider mission data

All the glider mission data, elaborated as described in the previous sections, were saved on Cayman at /storage/sire/work/glider/RG_QC/ since they are required for the Quality Control procedure. It is suggested to reorganize such data in appropriate folders on Cayman at /storage/sire/data/glider/.

3. Data quality control

The ad-hoc script *RG_QC* for the quality control of the OGS glider data is run on the mission files generated as described in section 2. The script considers all the scientific parameters and was realized based on Wong et al. (2021), Troupier et al. (2015), Willis (2015 and 2016) and EuroGOOS DATA-MEQ working group (2010). The script and all the needed functions/libraries are stored on Cayman at /storage/sire/work/glider/RG_QC.

Example: *[Flags]=RG_QC('convex21',0,0);*

The script It requires as input:

- the name of the glider mission file (all the mission files must be included at the same level of the script);
- a force parameter (0 or 1; 1 to force the script over all the tests);
- the number of the test to run (to run a particular test only; 0 for all the tests).

The output of the script is a new glider mission file (*mission_name_QC.mat*) that includes a new parameter named Flags and the variables already stored in the input file. The new parameter is a matrix whose rows correspond to the results of specific tests following the convention reported in Table 5.

The performed tests are described in the following sections and resumed in Table 6.

FLAG	MEANING
0	No QC was performed
1	Good
2	Probably good
3	Probably bad
4	Bad
5	Value changed
6	Not used
7	Not used
8	Estimated value (interpolated, extrapolated, or other estimation)
9	Missing value

Table 5. Quality flag scale.

All the QC glider mission data were saved on Cayman at /storage/sire/work/glider/RG_QC/. Also in this case, it is suggested to reorganize such data in appropriate folders on Cayman at /storage/sire/data/glider/.

TEST NUMBER	MEANING
1	Impossible location test
2	Position on land test
3	Impossible speed test
4	CTD depth test
5	Nav depth test
6	OXY depth test
7	WET depth test
8	Global range test temp
9	Global range test sal
10	Global range test oxy
11	Global range test chl
12	Global range test cdom or bb470
13	Global range test bb700
14	Regional range test temp (Med & Antarctic)
15	Regional range test sal (Med & Antarctic)
16	Regional range test oxy (Med & Antarctic)
17	Regional range test chl (Med & Antarctic)
18	Regional range test cdom or bb470 (Med & Antarctic)
19	Regional range test bb700 (Med & Antarctic)
20	Spike test temp
21	Spike test sal
22, 23	MEDian with a Distance (MEDD) test
24	Digit rollover test TEMP
25	Digit rollover test SAL
26	Stuck value test TEMP
27	Stuck value test SAL
28	Stuck value test PDENS
29	Stuck value test OXY
30	Stuck value test CHL
31	Stuck value test cdom or bb470 (Med & Antarctic)
32	Stuck value test bb700
33	Density inversion test

Table 6. Performed tests. The results (flags) of the N test are reported in the N column of the *Flags* variable (matrix), with N the number indicated in the first column.

3.1. Impossible location test

This test checks that the observed longitude and latitude respect the following convention:

- Latitude in range −90 to 90;
- Longitude in range −180 to 180.

If longitude or latitude fails this test, the corresponding line of the first column of the *Flags* variable is flagged as bad (4); lines of missing values are flagged with 9.

3.2. Position on land test

This test requires that the positions are collected in the ocean. We used the 5-minute bathymetry available at http://www.ngdc.noaa.gov/mgg/global/etopo5.html.

If position fails the test, the corresponding line of the second column of the *Flags* variable is flagged as bad (4); lines of missing values are flagged with 9.

3.3. Impossible speed test

This test requires that the horizontal speeds of the glider are within a threshold of 7.2 m/s (2 km/h). This threshold is higher than the one of Wong et al. (2021) because gliders move faster than floats. Lines of missing values are flagged with 9 and not considered further in the test. The test considers good the first position and time (reference triplet) and compute the speed using the second triplet of values (evaluated triplet; longitude, latitude and time), then, if the computed speed is less than the threshold, the corresponding line (evaluated line) of the third column of the *Flags* variable is flagged as good (1), the evaluated triplet substitutes the reference triplet of the test and the next triplet is considered (evaluated). On the contrary, if the computed speed exceeds the threshold, the corresponding line (evaluated line) of the third column of the *Flags* variable is flagged as bad (4), the reference triplet of the test does not change and the next triplet is considered (evaluated).

3.4. Depth test

This test checks for any depth exceeding the maximum depth of 1000 meters. If some depth values fail the test, the corresponding lines of the *Flags* variable are flagged as bad (4); lines of missing values are flagged with 9. The test also recognizes possible suspicious data (depth exceeding the maximum depth - 5 meters) and flag the corresponding lines as probably bad (3). Additionally, the test verifies the depth data at surface. Lines corresponding to data higher than 0 are flagged as probably bad (3) and lines corresponding to data higher than 5 are flagged as bad (4).

The test is applied on the CTD and Navigation depth data (DepthCTD and DepthNav parameter, respectively); the results of the test are recorded on column 4 and 5 of the *Flags* variable, respectively.

The depth of the oxygen and WetLabs sensor may differ from the one of the CTD or Navigation sensors as described in 2.3. The test is therefore applied to the DepthOXY and DepthWET parameters too and the results are recorded on column 6 and 7 of the *Flags* variable, respectively.

3.5. Global range test

This test applies a gross filter on the observed values of temperature, salinity, oxygen, chlorophyll, CDOM (or backscattering measured at 470 nm) and backscattering measured at 470 nm. The ranges need to accommodate the following extremes:

- Temperature must be in the range –2.5 to 40.0°C;
- Salinity must be in the range 30 to 41.0;
- Oxygen must be in the range 0 to 11.2 ml/l (instrumental limits);
- Chlorophyll must be in the range 0 to 50 μg/l (instrumental limits);
- CDOM must be in the range 0 to 375 ppb (instrumental limits);
- Backscattering at 470 nm must be in the range 0 to 5 $m^{-1}sr^{-1}$ (instrumental limits);
- Backscattering at 700 nm must be in the range 0 to 5 $m^{-1}sr^{-1}$ (instrumental limits);

The lines corresponding to values outside the range are flagged as bad (4); lines of missing values are flagged with 9. The results of the test run on the temperature, salinity, oxygen, chlorophyll, CDOM (or backscattering measured at 470 nm) and backscattering measured at 470 nm are recorded on column 8, 9, 10, 11, 12 and 13 of the *Flags* variable, respectively.

3.6. Regional range test

This test applies specific ranges for observations from the Mediterranean Sea and Antarctica area to further restrict what are considered sensible values.

The Mediterranean Sea is defined by the region:
30N, 6W; 30N, 40E; 40N, 35E; 42N, 20E; 50N, 15E; 40N, 5W; 30N, 6W.

The Antarctica area by the region below 70S.

The applied ranges are indicated in Table 7.

PARAMETER	MED RANGE	ANTARCTICA RANGE
Temperature (°C)	4 to 40	-2.5 to 5
Salinity	34 to 41	32 to 36
Oxygen (ml/l)	0 to 10	NA
Chlorophyll (µg/l)	0 to 5	NA
CDOM (ppb)	NA	NA
Backscattering at 470 nm ($m^{-1}sr^{-1}$)	NA	NA
Backscattering at 700 nm ($m^{-1}sr^{-1}$)	NA	NA

Table 7. Parameter ranges for the Mediterranean Sea and Antarctica area. NA indicates not available.

The lines corresponding to values outside the range are flagged as bad (4); lines of missing values are flagged with 9. The range for some parameters is set NA. In this latter case, the test is not run and the lines are flagged as 0. Nevertheless, the script is ready to run the regional test for these parameters too, as soon as their regional ranges are defined. The results of the test run on the temperature, salinity, oxygen, chlorophyll, CDOM (or backscattering measured at 470 nm) and backscattering measured at 470 nm are recorded on column 14, 15, 16, 17, 18 and 19 of the *Flags* variable, respectively.

3.7. Spike test

This test is the same run by Wong et al. (2021). The temperature thresholds are modified with respect to Wong et al. (2021) due to the higher sampling frequency of the glider with respect to the float. The difference between sequential measurements, where one measurement is significantly different from adjacent ones, is a spike. The algorithm is used on vertical profiles of temperature and salinity.

$$\text{Test value} = | V2 - (V3 + V1)/2 | - | (V3 - V1) / 2 |$$

where V2 is the measurement being tested, and V1 and V3 are the values above and below.

Temperature: The line corresponding to V2 is flagged as bad (4) when:

- the test value exceeds 1.0°C for pressures less than 500 dbar, or
- the test value exceeds 0.33°C for pressures greater than or equal to 500 dbar.

Salinity: The line corresponding to V2 is flagged as bad (4) when:

- the test value exceeds 0.9 PSU for pressures less than 500 dbar, or
- the test value exceeds 0.3 PSU for pressures greater than or equal to 500 dbar.

Lines of missing values are flagged with 9. The results of the test run on the temperature and salinity are recorded on column 20 and 21 of the *Flags* variable, respectively.

3.8. MEDian with a Distance (MEDD) test

This test is the same that is run for the Argo float data (Wong et al., 2021) and is a set of algorithms based on three main steps:

- First, the computation of a sliding median with some customizations.
- Then, limits are computed that are at relative 2-dimensional distance d from the median.

- Finally, these limits are also computed for the density profile. There is a spike if both the density profile and the (temperature or salinity) profile are out of limits. If there is no conductivity sensor, then the spikes in temperature are evaluated using a bigger d value.

Detailed specifications and Matlab codes for this test can be found on:

https://github.com/ArgoRTQC/matlab_MEDD

Lines of missing values are flagged with 9. The lines of temperature and salinity values that fail this test are flagged as bad data (4). The results of the test are recorded on column 22 and 23 of the *Flags* variable for the temperature and salinity, respectively.

3.9. Digit rollover test

The digit rollover occurs when the allocated bits range is exceeded (see for example Wong et al., 2021). This test is used to make sure the rollover is properly detected. In particular, if the parameter difference of between adjacent pressures overpasses a threshold, the corresponding line of data is flagged as bad (4). The threshold is 10°C and 5 for the temperature and salinity, respectively.

Lines of missing values are flagged with 9. The results of the test run on the temperature and salinity are recorded on column 24 and 25 of the *Flags* variable, respectively.

3.10. Stuck value test

This test looks for measurements in a profile being identical. The test is applied to all the scientific parameters (TEMP, SAL, PDENS, OXY, CHL, CDOM (or BB470) and BB700). Please note that the stack values in the case of the optic parameter may be inaccurate due to the low resolution of the WetLabs sensor.

Lines of missing values are flagged with 9. The results of the test run on the temperature, salinity, potential density, oxygen, chlorophyll concentration, CDOM and Backscattering at 470 and 700 nm are recorded on columns from 26 to 32 of the *Flags* variable, respectively.

3.10. Density inversion test

This test compares potential density between measurements in a profile, in both directions, i.e. from top to bottom, and from bottom to top. Values of temperature and salinity at the same pressure level P_i are used to compute potential density ρ_i. A threshold of 0.03 kg m^{-3}/m (see Wong et al., 2021 and Klein et al., 2020) is allowed for small density inversions.

Lines of missing values are flagged with 9. The results of the test run on the potential density data are recorded on columns 33 of the *Flags* variable, respectively.

It may be useful to implement in the next version of the script an inversion test that also checks for the oxygen and optic data. Indeed, in the case of incomplete Seaglider transmissions, we noticed that the data recorded by the Aanderaa and WetLabs sensors are saved "upside down" (Fig. 9). The issue is not present with the full dataset.

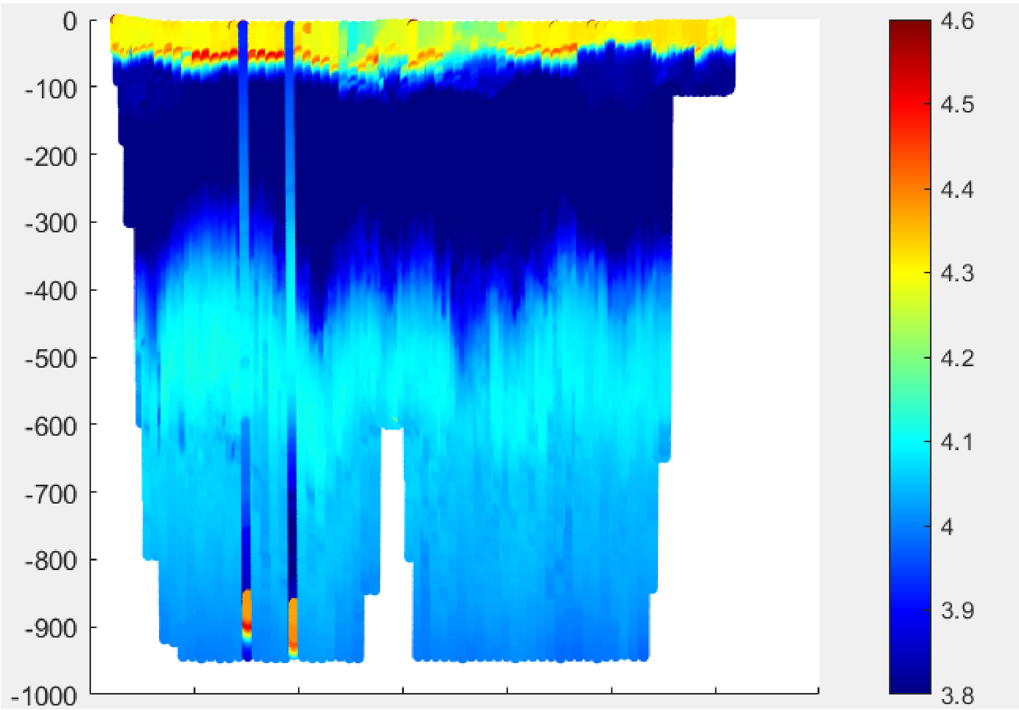


Fig. 9. Oxygen data plotted from the RT data transmitted during the PreConvex19 mission.

3.11. Computational time

All the QC tests are quite fast even for a long glider mission, except the MEDD test that needs a large amount of time to be concluded. The computational time for each glider mission QC elaboration (all the QC tests) was recorded (Table 8 and Figs. 10 and 11) and a time estimation formula is given for the elaboration on Cayman and on a Notebook i7-10510U CPU @ 1.80GHz 2.30 GHz - 32 GB RAM (Table 9).

Mission	number of data (x1000) NOT NaN	time on the Notebook (min)	time on Cayman (min)
estro20	27	5	
melmas18	47	14	
convex17	60	20	12
COCONET	77	25	
preconvex19	107	37	
Quarnero	107	103	
cinel17	133	58	
melmas19	253	192	146
preconvex17	280	195	
CONVEX14	283	228	
convex16	313		230
cinel16	333	284	
preconvex16	353		284
convex19	377		303
preconvex18	423		368
preconvex20	693		958
convex20	1553		4824
convex18	683		924
convex21	2637		13680

Table 8. Computational time for the QC elaboration of each glider mission dataset.

Cayman	$y = 0{,}005x^{1{,}8667}$
Notebook	$y = 0{,}0325x^{1{,}5654}$

Table 9. Time estimation formula for the QC elaboration on Cayman and on the Notebook.

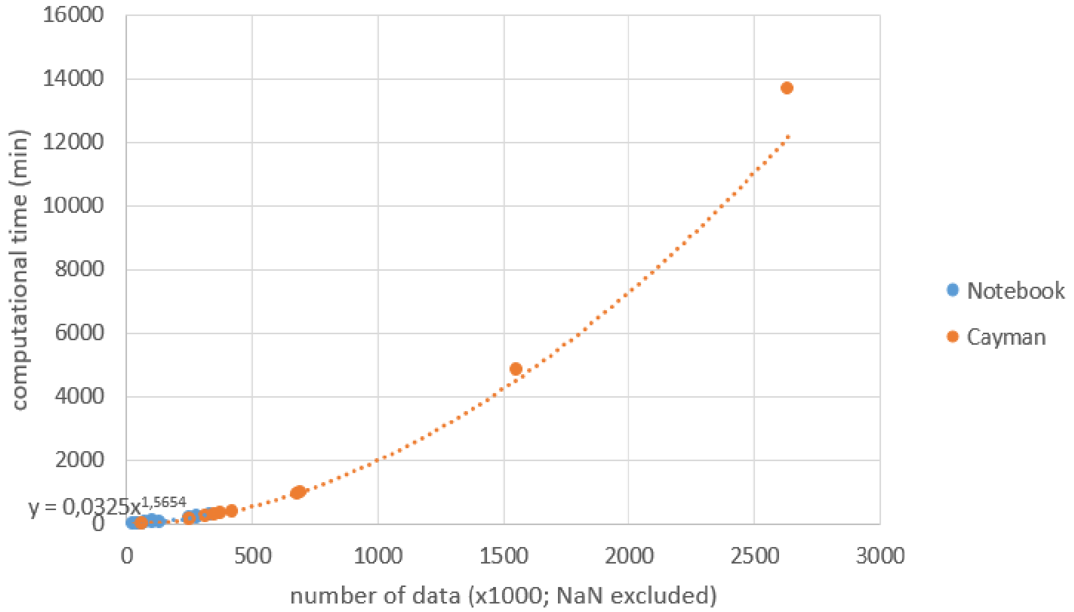


Fig. 10. Computational time for the QC elaboration on Cayman (orange dots) and on the Notebook (blue dots).

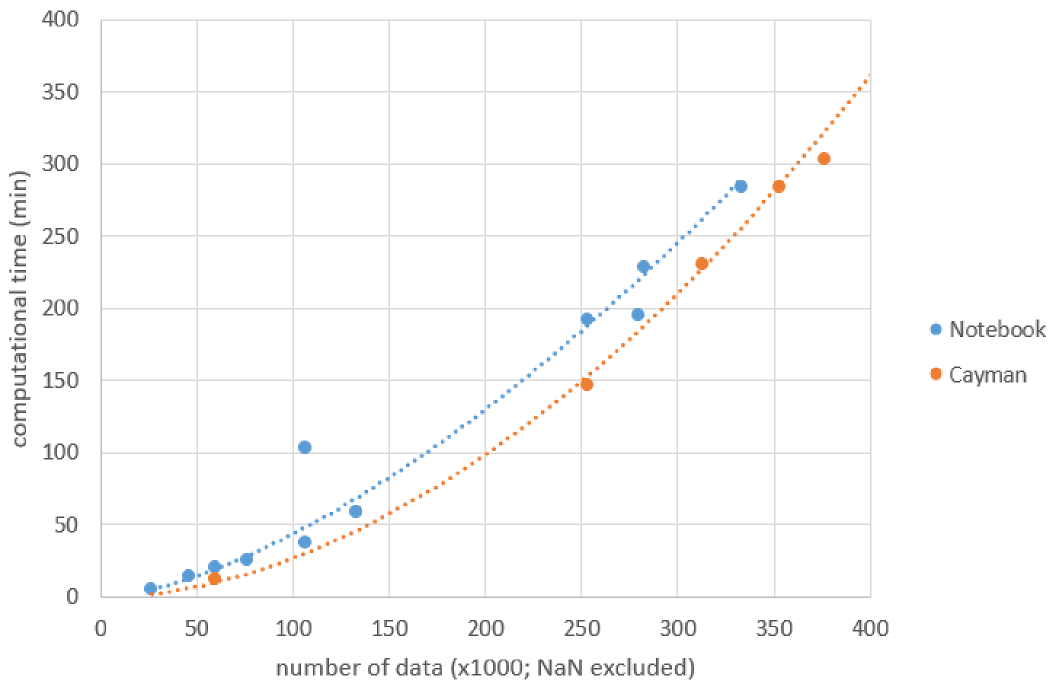


Fig. 11. Zoom of Fig. 10.

4. QC data interpolation and glider mission DOI

The glider mission QC data were averaged every 20 meters (no overlap; 5 meters in the case of the Quarnero mission) after removing the data flagged as bad (4) and the resulting files were saved at the OGS-NODC and identified with the following DOIs:

- COCONET 10.13120/486d2b87-ce42-4632-8e8c-8a1a11272585
- CONVEX14 10.13120/9f33003d-00d2-4e82-88b4-b1bb8ad4fc5b
- Quarnero 10.13120/e20f1fe1-73f5-442a-9baa-212682995a51
- preconvex16 10.13120/354c6c0b-85b8-452b-bba1-209e92256027
- convex16 10.13120/1cd43eef-be72-4477-9213-3aaa9e2289ab
- Cinel16 10.13120/5df8c74c-d95b-4c31-b317-8c18154c1106
- preconvex17 10.13120/85b38632-3903-42c0-b8d0-3783ac6b4204
- convex17 10.13120/ae3bbc1b-acbf-44c4-8e07-a9fd6302e8b9
- Cinel17 10.13120/f754f33a-7809-44cf-a9a4-1927e6db435b
- preconvex18 10.13120/4f6c6503-e0ed-4145-acef-a79e8f0187bb
- convex18 10.13120/e3324610-f72b-4d53-b183-4310fb13f9cb
- Melmas18 10.13120/7f40aa29-31b4-4858-8286-d40c7485a501
- preconvex19 10.13120/35923710-bb27-46ea-9fb0-522776bbbab6
- convex19 10.13120/fd1e091c-0113-44cc-a5d6-f608375a48ee
- Melmas19 10.13120/b8eb991c-f7eb-4f1a-be4a-20cc39aba173
- preconvex20 10.13120/1ccb753a-78d9-4026-80c9-ee883db8921e
- convex20 10.13120/e7277c6b-444a-4d61-8288-596af1bac3ff
- Estro20 10.13120/e4c497a1-c139-4285-911b-6c55bcc942b2
- convex21 10.13120/161e025d-7ddc-4231-ac98-9a772f5ebd3f

References

EuroGOOS DATA-MEQ working group (2010). Recommendations for in-situ data Near Real Time Quality Control. https://doi.org/10.13155/36230, 23pp.

Garau B., Ruiz S., Zhang W.G., Pascual A., Heslop E., Kerfoot J., Tintoré J., 2011. Thermal Lag Correction on Slocum CTD Glider Data. J. Atmos. Oceanic Technol., 28, 1065–1071.

Gerin R. (2013). STSM Scientific Report - COST Action ES0904: Glider CTD problems and glider simulation code. REL. OGS 2013/113 OCE 42 MAOS, Trieste, Italy, 16 pp.

Gerin R., Mauri E., Bussani A., Zuppelli P., Kokkini Z., Pacciaroni M., Kuchler S. and Poulain P.-M. (2017). The CINEL16 and CINEL17 OGS glider missions. Rel. OGS 2017/38 OCE 11 MAOS, Trieste, Italy, 28 pp.

Gerin R. and Pacciaroni M. (2019). MELMAS18 glider CTD issue and data correction. Rel. OGS 2019/33 OCE 12 MAOS, Trieste, Italy, 13 pp.

Gerin R., Martellucci R., Mauri E., Kokkini Z., Medeot N., Nair R., Zuppelli P., Comici C. and Pachou A. (2020). Oxygen concentration in the South Adriatic Sea: the gliders measurements. Rel. 2020/36 Sez. OCE 11 MAOS, Trieste, Italy, 31 pp.

Klein B., Coatanaon C., Purokoski T, Siiriä S.-M., Tuomi L., Walcowski W., 2020. D4.4.7_Data_management_for_floats_in_the_Baltic_v1.0. Mocca, Eoro-Argo ERIC, 23 pp.

Troupin C., Beltran J.P., Heslop E., Torner M., Garau B., Allen J., Ruiz S., Tintoré J. (2015). A toolbox for glider data processing and management. Methods in Oceanography, Volumes 13–14, pp 13-23, https://doi.org/10.1016/j.mio.2016.01.001.

Willis Z.S. (2016). Manual for Quality Control of Temperature and Salinity Data Observations from Gliders. IOOS Program, Version 1.0, May 2016, 95 pp.

Willis Z.S. (2016). Manual for Real-Time Quality Control of Dissolved Oxygen Observations. IOOS Program, Version 2.0, April 2015, 48 pp.

Wong A., Keeley R., Carval T., Argo Data Management Team (2021). Argo Quality Control Manual for CTD and Trajectory Data. https://doi.org/10.13155/33951, 67pp.

Zuppelli P., Kuchler S. and Gerin R. (2017). Procedure per verificare le funzionalità e test del sensore Wetlabs BB2FL-VMT Eco Puck. REL. OGS 2017/29 OCE 9 MAOS, Trieste, Italy, 6 pp.