Documentation for the Slocum glider toolbox for MATLAB.

Mathieu Dever

March 2016

Contents

1 Motivation

This MATLAB toolbox as been developed to provide a starting point when wanting to process and visualize Slocum glider data. It is in no way exhaustive and can certainly be improved in many ways. Some of these routines depends on other MATLAB functions that are not built-in the software. Some of them I authored, which are all included in the toolbox, while others are from MATLAB packages that are commonly used in oceanography (e.g., The *TEOS-10 Gibbs seawater oceanography toolbox*; [1])

2 Toolbox content

2.1 GLIDER_DATA_CSV2MAT.M

2.1.a Objectives

The function `glider_data_csv2mat.m` is probably the least universal of the package. It has been written to extract the data from the csv files provided by the *glider group at Dalhousie*. It depends on some subroutines included in the package (1 subroutine for each instrument), as well as on the TEOS-10 oceanographic toolbox [1]

2.1.b Inputs

The user is only required to provide the prefix used for the instrument files. The function assumes that the filename format follows [prefix]_[instrument suffix]. As an example, for the file `2012-05-24_view_sci_water.csv` the prefix would be `2012-05-24_view_sci`, and the instrument suffix would be `water`, corresponding to the CTD data. A common mission would usually include 5 files, with identical prefixes and the following suffixes:

- `water` for the CTD data,
- `bb2fls` for the backscattering data used to derive CDOM and backscattering at wavelengths 660 μ m and 880 μ m,

- `bb2flsv2` for the backscattering data used to derive Chlorophyll concentration (from fluorescence) and backscattering at wavelengths 470 μ m and 532 μ m,
- `ocr504i` for the irradiance data,
- `oxy3835` for the optode (i.e., oxygen) data.

2.1.c Outputs

The function extracts the data and saves a mat-file that is composed of a MATLAB structure for each extracted instrument (see **??**)

- `ctd` for the CTD data,
- `fls` for the backscattering data used to derive CDOM and backscattering at wavelengths 660 μ m and 880 μ m,
- `flsv2` for the backscattering data used to derive Chlorophyll concentration (from fluorescence) and backscattering at wavelengths 470 μ m and 532 μ m,
- `irrad` for the irradiance data,
- `oxy` for the optode (i.e., oxygen) data.

2.2 GLIDER_SECTIONNER.M

2.2.a Objectives

The function `glider_sectionner.m` extracts different sections within a glider mission, which are defined by the user. It depends on the `GPS2dist.m` function.

2.2.b Inputs

The only required the coordinates (longitude, latitude) of each datapoint. The user is first prompted to provide the number of sections that will be extracted. The user is then prompted to click on the displayed figure to select the starting and ending points of each section the user wishes to extract. If the user clicks before(after) the first(last) datapoint, the first(last) datapoint is automatically selected.

2.2.c Outputs

The function creates a matrix of dimensions [N × 2], where N is the number of sections defined by the user. The first column includes the starting indices of each section, while the second column include the indie of the last datapoint of the section.

2.3 `GLIDER_PROFILER.M`

2.3.a Objectives

The function `glider_profiler.m` isolate each upcast and downcast throughout a glider mission, or section, specified by the user. Its major weakness is that it relies on an arbitrary threshold value defined by the author. The value of 10 m is used to defined what constitute a cast; it goes around the issue that the glider sometimes briefly moves in the opposite direction of the cast direction due to vertical currents.

2.3.b Inputs

The function requires a input variable (as a column vector), or variables concatenated together into a matrix [M × N], where M is the number of observations and N the number of variable to process (e.g., temperature, salinity, backscatter, ...). It also requires the depth of each observation, used to separate upcasts from downcasts. An additional input can be specified to produce a plot showing the starting point and finishing point used to define each profile.

2.3.c Outputs

The function returns a structure that contains a cell variable for both upcasts and downcasts that includes the a matrix of dimension [R × S], where R is the number of depth levels, and S is the number of profiles detected, for each variable specified as an input. It also provides a matrix for both upcasts and downcasts of the same dimensions that includes the depth for each observation.

2.4 GLIDER_GRIDDER.M

2.4.a Objectives

The function `glider_sectionner.m` aims at converting the the glider data from a vector to a 2-dimensional matrix. It relies on the fact that gliders samples at very high resolution; no interpolation is therefore necessary in-between datapoint. Instead, a 2-dimensional moving block averages all the datapoint included within the block to produce an averaged value at each grid point.

The user must be aware that such an approach has evidently some limitations, especially when dealing with very high spatial resolution. A `counter` input is therefore available to provide the number of datapoint averaged within each block. It is to the responsibility of the user to ensure that the number of datapoint included in each averaging block is satisfactory.

A future update will include the option of using interpolation techniques instead of the basic 2-D averaging block.

2.4.b Inputs

The user is required to provide the vector of the data to be gridded (e.g., temperature, salinity, fluorescence, ...), as well as the 3-D coordinates (longitude, latitude, depth) of each data point. Execute `help glider_gridder` for more information on the optional arguments.

2.4.c Outputs

The function provides 3 variables: the gridded variable (2-D matrix), as well as the grid coordinates (2 row vectors). An additional output including the number of datapoint averaged to calculate each grid point is also provided if the user specified `'counter','yes'` when calling the function. The x-axis is defined as the distance from the first datapoint by default. The user has the option of specifying the reference point for the distance (e.g., distance from shore, distance from a buoy, ...). Execute `help glider_gridder` for more information.

3 Velocities from gliders

3.1 `glider_gridder.m`

3.2 `glider_data_csv2mat.m`

Bibliography

[1] Trevor J McDougall and Paul M Barker. Getting started with teos-10 and the gibbs seawater (gsw) oceanographic toolbox. *SCOR/IAPSO WG*, 127:1–28, 2011.