analyze_surface_v1_0 - a MATLAB toolbox for water-mass analysis on density surfaces

A. Klocker

December 6, 2009

1 Introduction

This document describes a Matlab toolbox including a Graphical User Interface (GUI) which can be used to label a data set with a density variable (e.g. potential density or neutral density), form surfaces of constant density and calculate several properties on these surfaces (e.g. neutral helicity, fictitious diapycnal diffusivity and diapycnal advection due to nonlinearities in the equation of state). This toolbox also includes an algorithm which optimizes density surfaces to ensure they are as close to neutral tangent planes as possible - thus minimizing the fictitious diapycnal diffusivity. This algorithm therefore produces surfaces which are ideal to be used as water-mass boundaries in inverse models or for water-mass analysis in ocean models or observational data.

2 Installation

The Matlab toolbox can be found on the CD included in this thesis or can be downloaded from

`http://www.TEOS-10.org`

as a tar.gz-file for Linux users or as a zip-file for Windows users. Under Linux run

tar -xvf analyze_surface.tar.gz

to unpack the code. This creates a directory called *analyze_surface_ver1_0*. Set the path in Matlab to the directory in which you have installed the Matlab code.

Additional freely available libraries have to be installed for full functionality. For the calculation of density and related quantities the library *eoslib05* is needed. This library contains routines involving the equation of state as described in Jackett et al. (2006) and can be found at

`http://www.marine.csiro.au/~jackett/eos`.

To use the function calculating neutral density as described in Jackett and McDougall (1997) additional Matlab/Fortran code has to be downloaded from

`http://www.marine.csiro.au/~jackett/NeutralDensity`.

Installation information for this software can be found in a README file within the code. Additionally the seawater library has to be downloaded from

`http://www.cmar.csiro.au/datacentre/ext_docs/seawater.htm`.

The Matlab path then has to be set to point to these libraries as well.

To start the GUI, type *analyze_surface.m*. A GUI as seen in Fig.1 should open.

3 Input

The input can be any gridded data (model output or observational data) and must include salinity, temperature, pressure, latitude and longitude. Temperature can be either potential temperature or conservative temperature (McDougall, 2003). If the input is potential temperature it will be converted to conservative temperature for all further calculations as it is the most conserved temperature variable available (McDougall, 2003). Optional input variables are buoyancy frequency, N^2, gravitational acceleration, g, latitudinal and meridional velocities, u and v, and a prelabelled density field, *pre*. N^2 and g will be calculated automatically after opening a file if they do not exist. Velocity data are only necessary if calculating the diapycnal velocity

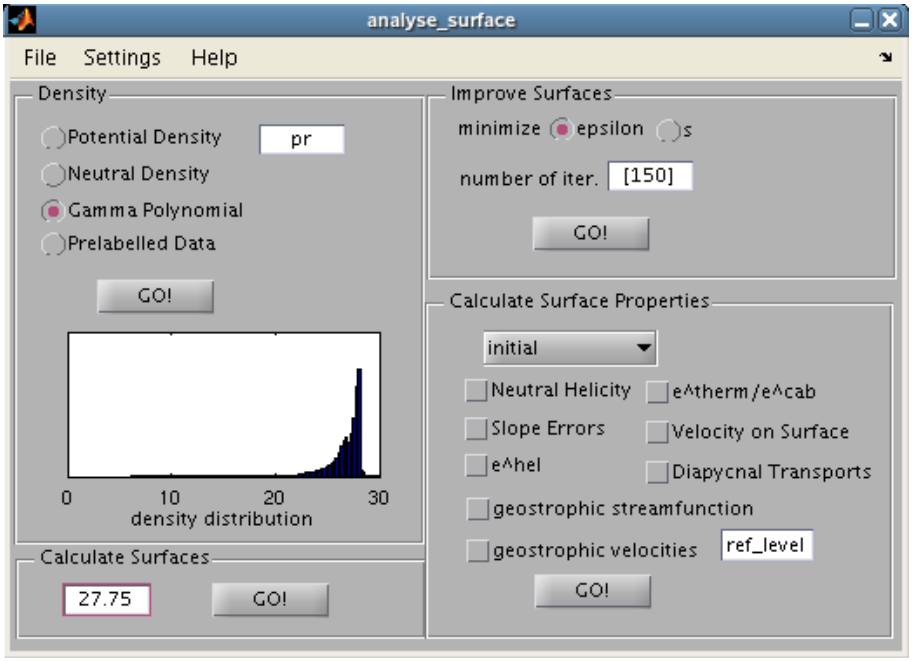


Figure 1: Shown is the main GUI of the analyze_surface toolbox. The figure in this GUI shows the density distribution in the data set.

due to neutral helicity and its associated transports (see below). Prelabelled density is only necessary if one wants to use a density variable which is not included in this GUI.

The input variables have to be labelled and in a format as shown in Table 1 (bold entries are necessary inputs, all other entries are optional).

After opening the GUI by typing *analyze_surface.m* one can open a file by going to *File* → *Open*. The file opened has to be a *.mat* file. After choosing a file and pressing *ok* a GUI in which one can change input settings as seen in Fig.2 opens. In this GUI one can choose the Arakawa grid used (only necessary if the file includes velocity data; otherwise this option does not make any difference) and a wrap option. The wrap option treats the most eastern points as being adjacent to the most western points of the gridded data (when using a global data set) or treats the most eastern and western points as independent (when using a regional data set).

If opening a file which has been saved with this GUI before, all data will already be in a Matlab-structure called *ocean* (e.g. conservative temperature is found in *ocean.ct*, etc.). This structure can be opened by the GUI without input variables as described above. A more detailed description of this structure can be found in section 7.

Table 1: This table shows necessary and optional input for the analyze_surface GUI. Necessary input variables are bold, all other input variables are optional. The right column shows the necessary format of the input variables.

variable	variable name	format
s	salinity	[depth lats longs]
t or ct	potential or conservative temp.	[depth lats longs]
p	pressure	[depth lats longs] or [depth]
lats	latitude	[lats longs] or [lats]
longs	longitude	[lats longs] or [longs]
n2	buoyancy frequency	[depth lats longs]
g	gravitational acceleration	[lats longs]
u	latitudinal velocity	[depth lats longs]
v	meridional velocity	[depth lats longs]
pre	prelabelled density	[depth lats longs]

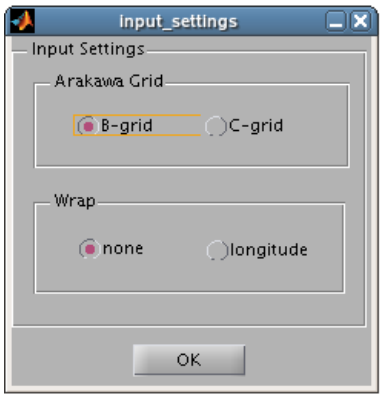


Figure 2: In this GUI the Arakawa grid and the wrap option are set. It opens automatically if a new mat-file is opened or it can be opened by going to *Settings* → *Input Settings*.

4 Labelling a data set with density and finding surfaces of constant density

The hydrographic data can be labelled with the following density variables:

- potential density, σ_{p_r} (where p_r is reference pressure);
- neutral density, γ^n (Jackett and McDougall, 1997);
- a rational function approximating neutral density, γ^{rf} (McDougall and Jackett, 2005);
- any prelabelled density variable

The choice of density variable depends on the application it is used for and the data (i.e. model output or observational data), with the advantages and disadvantages of these density variables explained in Klocker et al. (2009a,b) and Klocker and McDougall (2010). If choosing potential density a reference pressure has to be set in the field p_r (use 2000 for 2000 dbar,etc.). Once a density variable is chosen and the *GO!* button is pressed a density distribution should appear (see Fig.1). It is then possible to calculate a surface of constant density by typing a density into the field *levels* (the field with the input *27.75* in Fig.1) and pressing the *GO!* button. At the moment it is only possible to calculate one density surface at a time - changing this to be able to calculate more density surfaces at a time will be one of the next improvements of this code.

This Matlab toolbox also contains a m-file to compute neutral density according to Eden and Willebrand (1999), γ^{EW}, which can be used to label a data set and be used as prelabelled density in this GUI. Note that this density variable was constructed for the North Atlantic only.

Also note that in all calculations data above the mixed-layer depth are excluded. The mixed layer is calculated as in de Boyer Montégut et al. (2004) using a threshold value of density from the surface value ($\Delta\rho = 0.3\ kg/m^3$).

5 Optimizing density surfaces

This part of the software optimizes density surfaces as described in Klocker et al. (2009a,b), giving a surface which is much closer to being neutral than any other density surfaces previously used. The algorithm used to optimize density surfaces is computationally more expensive than labelling data sets with a density variable as described above. In contrast to the labelling of

a three-dimensional data set with a density at every grid point and then finding a surface of constant density, this algorithm optimizes existing density surfaces. At this point one has to decide if this computationally intensive optimization of density surfaces is necessary or not. The improved neutrality of these surfaces might for example not be significant if used for water-mass boundaries in large box inverse models of non-synoptic hydrographic sections but will definitely improve inverse models using synoptic sections for process studies that particularly target the determination of mixing.

The algorithm used to optimize density surfaces needs an initial surface as input. This surface can be produced by labelling a data set with a density variable and finding a surface of constant density as described above. The closer this initial density surface is to being 'neutral' the faster the residual slope error between the surface and the neutral tangent plane will be minimized.

In more detail, the algorithm first calculates the initial density gradient error on the initial surface,

$$\boldsymbol{\epsilon}^{init} = \beta^{\Theta} \nabla_{init} S - \alpha^{\Theta} \nabla_{init} \Theta, \tag{1}$$

where the gradients ∇_{init} are calculated on the initial surface. Then the algorithm solves a least-squares problem in which it calculates a field of pertubation densities, $\phi'(x, y)$ (where ϕ is the locally referenced potential density), which minimizes the density gradient error on the new surface. This field of pertubation densities is then used to improve the initial denity surface as can be seen in

$$\boldsymbol{\epsilon}^{new} = \boldsymbol{\epsilon}^{init} + \nabla_a \phi'. \tag{2}$$

Due to the algorithm not knowing about vertical gradients of density its first guess will not lead to an ideal solution and several iterations are needed until $\boldsymbol{\epsilon}$ converges. Further detail about the algorithm can be found in Klocker et al. (2009a,b).

Before optimizing a density surface one can decide on the following options:

- decide between minimizing the slope error, $\boldsymbol{s}$, or the density gradient error, $\boldsymbol{\epsilon}$. Minimizing $\boldsymbol{\epsilon}$ is better when trying to understand the theory behind these optimized surfaces and $\boldsymbol{s}$ is better for minimizing the fictitious diapycnal diffusivity. Both give very similar results (Klocker et al., 2009a,b).

- choose the number of iterations; this number highly depends on the data set used and on the distance of the initial condition from the optimized surface (usually 50-200 iterations are adequate; the default is 150). The number of iterations is also highly dependent on the damping factor λ (see below).

- the damping factor λ; λ is used to damp the pressure change added to an initial density surface at every iteration to optimize this surface. In some cases (especially in shallower regions) if λ is set too high the optimization code will become unstable, therefore a smaller λ has to be chosen increasing the necessary iterations. The stability of the algorithm (and therefore the value for λ which can be used) is dependent on the distance of the initial surface to being 'neutral'. In some cases λ has to be set as low as $\lambda = 0.1$ (the default values is $\lambda = 0.3$ and has to be changed in the code if necessary).

  Examples of the depth change used by the algorithm to minimize the density gradient errors and the convergence of $\boldsymbol{\epsilon}$ to a minimum value can bee seen in Fig.3. An example for a case in which the damping λ is not well chosen can be seen in Fig.4.

Note that when producing an optimized surface all data above the mixed-layer depth will be excluded. The mixed layer is a place where other physics than neutral physics are relevant and therefore no density surfaces are produced in this region.

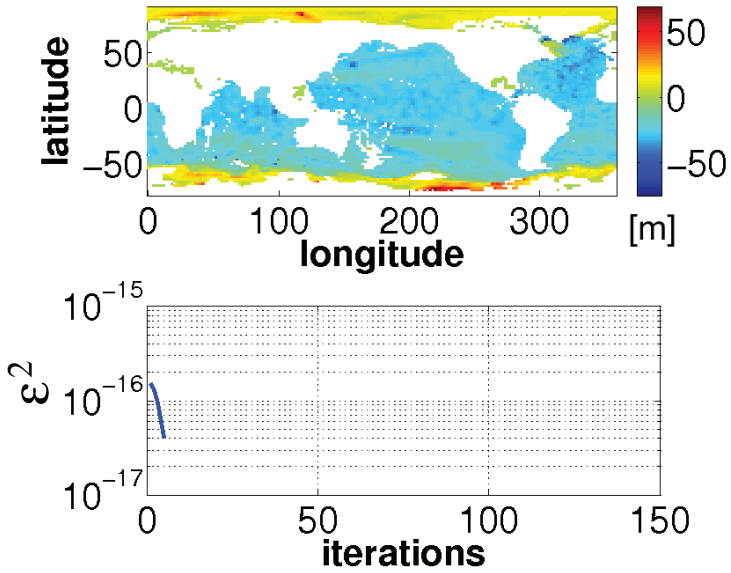
6 Calculating properties on density surfaces

The following properties can be calculated on the initial and optimized density surfaces (initial, improved or both surfaces can be chosen in the pull-down menu, all properties are in SI-units):

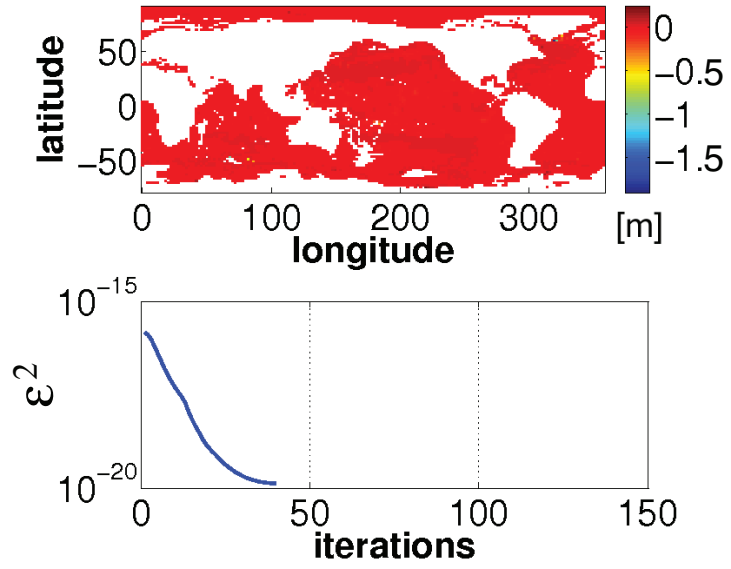
- Neutral Helicity - Neutral helicity, H^n, is calculated on the density surface according to McDougall and Jackett (2007),

$$H^n \approx g^{-1} N^2 T_b^\Theta \nabla_a p \times \nabla_a \Theta \cdot \boldsymbol{k}, \tag{3}$$

  where T_b^Θ is the thermobaric parameter, $T_b^\Theta = \beta^\Theta (\alpha^\Theta / \beta^\Theta)_p$, and ∇_a is the gradient along a continous 'density' surface. This equation is approximate due to the gradients of p and Θ being taken on a continous 'density' surface instead of being taken along a neutral tangent



(a)



(b)

Figure 3: Depth changes (per iteration) calculated from the field of pertubation densities, $\phi'(x, y)$ (top panels) and the decrease in the square of density gradient errors, $\boldsymbol{\epsilon}$ (shown on a log_{10}-scale, lower panels) for (a) the surface after 5 iterations and (b) for the surface after 40 iterations. In this figure a good value for λ has been chosen and $\boldsymbol{\epsilon}$ converges smoothly.

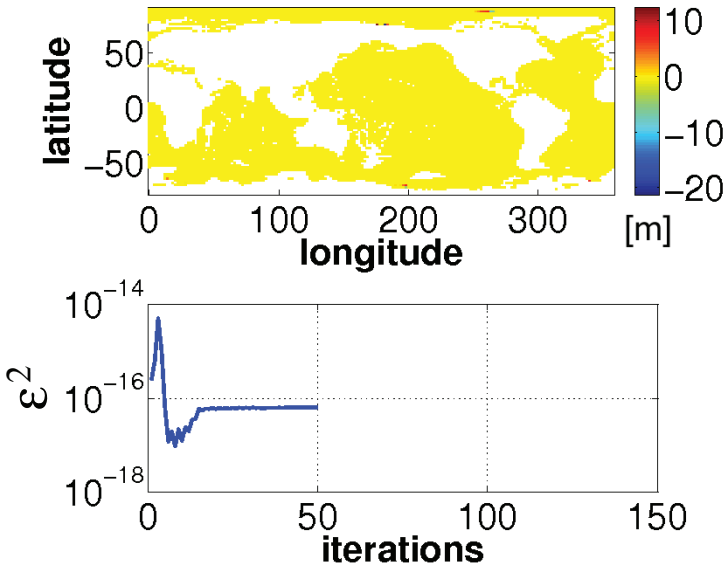


Figure 4: Depth changes (per iteration) calculated from the field of pertubation densities, $\phi'(x, y)$ (top panel) and the decrease in the square of density gradient errors, $\boldsymbol{\epsilon}$ (shown on a log_{10}-scale, lower panel) for the surface after 50 iterations. In this figure a bad (i.e. too large) value for λ has been chosen and $\boldsymbol{\epsilon}$ does not converge at a minimum value.

plane. On optimized approximately neutral surfaces these gradients are very similar to those along neutral tangent planes. Neutral helicity is calculated on tracer grid points (the red points in Fig.5).

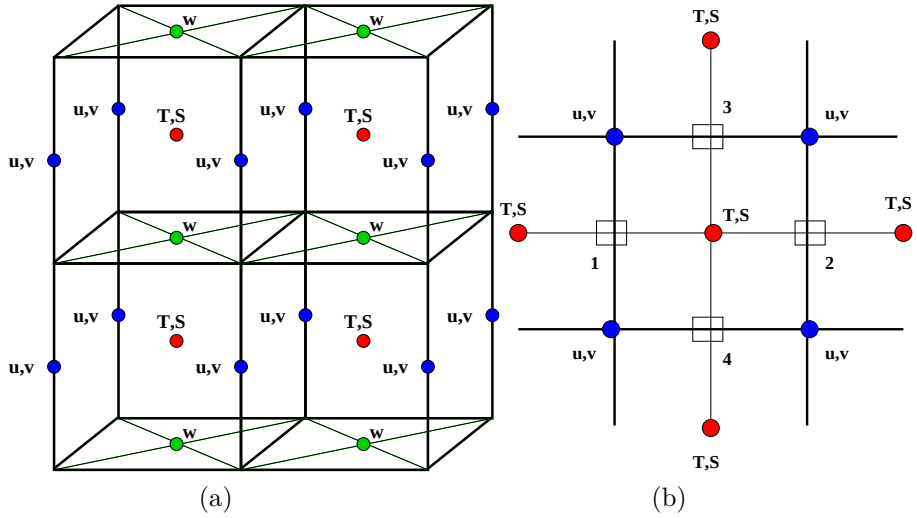


Figure 5: (a) A side-view and (b) a top-view of an Arakawa B-grid. Red dots are the tracer points, blue points the lateral velocity points and green points the vertical velocity points. Diapycnal velocities calculated with this toolbox are calculated at the tracer points. The boxes labelled 1 and 2 (3 and 4) are the locations where zonal (meridional) gradients are calculated.

- Slope Errors - This includes the calculation of the slope error, $\boldsymbol{s}$ (with $|\boldsymbol{s}|$, its x- and y-components and its curl, written as $ss, ss\_x, ss\_y, curl\_s$ in the code) and the density gradient error, $\boldsymbol{\epsilon}$ (with $|\boldsymbol{\epsilon}|$, its x- and y-components and its curl, written as $ee, ee\_x, ee\_y, curl\_e$ in the code). The slope error, $\boldsymbol{s}$, and the density gradient error, $\boldsymbol{\epsilon}$, can be written as (Klocker et al., 2009a,b)

$$\boldsymbol{\epsilon} = \beta^{\Theta}\nabla_a S - \alpha^{\Theta}\nabla_a \Theta = \frac{N^2}{g}(\nabla_n z - \nabla_a z) = \frac{N^2}{g}\boldsymbol{s}, \tag{4}$$

where ∇_n is the gradient along a neutral tangent plane.

The x-component of the gradient of S in Eqn.(4) is calculated as (giving the slopes at points 1 and 2 in Fig.5(b))

$$\nabla_x S = \frac{S^{east} - S^{west}}{e1t}, \tag{5}$$

where $e1t$ is the zonal distance between grid points. These slopes are then interpolated back onto the tracer grid points. The same is done for the other components. α^Θ, β^Θ, N^2 and g are calculated on the tracer grid points. The slope errors are not computed using the second part of Eqn.(4) due to the computational cost of the neutral tangent plane calculations.

- e^{hel} - The diapycnal velocity caused by neutral helicity. Depending on the surface on which e^{hel} is calculated, this diapycnal velocity can be due to errors in the way density surfaces are constructed and therefore is not solely caused by neutral helicity (this problem is minimized when using ω-surfaces, see Klocker and McDougall (2010) for a more detailed explanation). e^{hel} is defined as (Klocker et al., 2009a,b; Klocker and McDougall, 2010)

$$e^{hel} = \boldsymbol{V} \cdot \boldsymbol{s}, \tag{6}$$

where $\boldsymbol{V}$ is the lateral velocity and $\boldsymbol{s}$ is the slope error (Eqn.(4)). $\boldsymbol{s}$ is interpolated onto the tracer grid points and $\boldsymbol{V}$ is interpolated horizontally onto the tracer grid points as well and then vertically interpolated onto the density surface on which e^{hel} is calculated. For this interpolation the right choice of Arakawa B-grid or C-grid is important.

- e^{therm}/e^{cab} - The diapycnal velocities caused by thermobaricity (e^{therm}) and cabbeling (e^{cab}). These diapycnal velocities can be written as (McDougall, 1984, 1987)

$$e^{therm} = -\frac{g}{N^2} K T_b^\Theta (\nabla_a \Theta \cdot \nabla_a p), \tag{7}$$

$$e^{cab} = -\frac{g}{N^2} K C_b^\Theta (\nabla_a \Theta \cdot \nabla_a \Theta), \tag{8}$$

where the thermobaric coefficient and the cabbeling coefficient (McDougall, 1984, 1987) are

$$T_b^\Theta = \frac{\partial \alpha^\Theta}{\partial p} - \frac{\alpha^\Theta}{\beta^\Theta} \frac{\partial \beta^\Theta}{\partial p} \tag{9}$$

and

$$C_b^\Theta = \frac{\partial \alpha^\Theta}{\partial \Theta} + 2\frac{\alpha^\Theta}{\beta^\Theta}\frac{\partial \alpha^\Theta}{\partial S} - \left(\frac{\alpha^\Theta}{\beta^\Theta}\right)^2 \frac{\partial \beta^\Theta}{\partial S}, \tag{10}$$

and K is the isopycnal diffusivity. Due to our lack of knowledge about accurate values to use for the isopycnal diffusivity, a value of $K = 1000\ m^2 s^{-1}$ is used. These diapycnal velocities are calculated on the tracer grid points with the calculation of the gradients of p and Θ being done the same way as in the calculation of the slope errors.

- Velocity on density surface - The three-dimensional velocity field is vertically interpolated onto the continous 'density' surface.

- Diapycnal transports - The diapycnal velocities due to neutral helicity, thermobaricity and cabbeling are used to calculate diapycnal transports.

- Geostrophic streamfunction - The geostrophic streamfunction is calculated as (McDougall and Klocker, 2009)

$$\begin{aligned} \varphi^n(S_A, \Theta, p) = & \frac{1}{2}(p - \tilde{p})\tilde{\delta}(S_A, \Theta, p) \\ & - \frac{1}{12}\frac{T_b^\Theta}{\rho}(\Theta - \tilde{\Theta})(p - \tilde{p})^2 \\ & - \int_0^p \tilde{\delta} dp' \end{aligned} \tag{11}$$

for approximately neutral surfaces (i.e. γ^n- and ω-surfaces). $\tilde{\delta}(S_A, \Theta, p)$ in the first term of Eqn.(11) can be written as

$$\tilde{\delta}(S_A, \Theta, p) = v(S_A, \Theta, p) - v(\tilde{S_A}, \tilde{\Theta}, p), \tag{12}$$

where v is the specific volume anomaly. At every grid point (apart from those next to continents) we write differences of Eqn.(11) in the zonal and meridional direction,

$$\varphi^n = \varphi^n_{east} - \varphi^n_{west}$$
$$\varphi^n = \varphi^n_{north} - \varphi^n_{south}. \tag{13}$$

The tilde-values in the above equations are the reference values at the western/southern grid points. These differences in geostrophic streamfunctions are then used to construct a geostrophic streamfunction by using a least-squares approach. Further details about this method can be found in McDougall and Klocker (2009).

- Geostrophic velocities - From the geostrophic streamfunction geostrophic velocities can be calculated. These are referenced to the dynamic height at a specified pressure level (set in ref_level in Fig.1). Geostrophic velocities ± 2 degrees of the equator are excluded.

7 Saving files

To save a file go to *File* → *Save*. After choosing a file name all the variables are saved in a mat-file with the structure *ocean*. All the variables saved in the structure *ocean* can be seen in Table 2 (continued in Table 3). This structure can be opened directly by the analyze_surface GUI without the variables mentioned in section 3.

8 Contents of the toolbox

The following m-files are part of the analyze_surface toolbox. Files with an (*) are part of the toolbox but not the GUI.

- **ab_from_ct.m** - Calculate the thermal expansion coefficient α^Θ, the saline contraction coefficient β^Θ, $\frac{\alpha^\Theta}{\beta^\Theta}$, and the thermobaric coefficient, $T_b^\Theta = \beta^\Theta * (\frac{\alpha^\Theta}{\beta^\Theta})_p$.

- **analyze_surface.m** - analyze_surface.m – The main GUI for the analyze_surface toolbox.

- **analyze_surface_license.m** - License statement and permissions for the analyze_surface package.

- **bfrq_ct.m** - Calculate Brunt-Vaisala frequency squared (N^2) at the mid depths, p_mid, from the equation $N^2 = -\frac{g}{\rho} * \frac{\delta\rho}{\delta z}$ using the equation

Table 2: This table shows the output variables saved in the structure *ocean* This table is continued in Table 3.

variable	**variable name**
ocean.s	salinity
ocean.t	potential temperature
ocean.ct	conservative temperature
ocean.lats	latitude
ocean.longs	longitude
ocean.u	lateral velocity in x-direction
ocean.v	lateral velocity in y-direction
ocean.e1t	scale factor in x-direction
ocean.e2t	scale factor in y-direction
ocean.n2	buoyancy frequency
ocean.g	gravitational acceleration
ocean.p_mid	pressure at mid-points
ocean.pre	pre-labelled density
ocean.sigma	potential density
ocean.pr	reference pressure
ocean.gamma	neutral density
ocean.glevels	levels of density surfaces
ocean.mld	mixed-layer depth
ocean.sns	salinity on density surface
ocean.ctns	conservative temperature on density surface
ocean.pns	pressure on density surface
ocean.n2ns	buoyancy frequency on density surface
ocean.hel	neutral helicity on density surface
ocean.ss	slope error on density surface
ocean.sx	x-component of slope error on density surface
ocean.sy	y-component of slope error on density surface
ocean.curl_s	curl of slope error on density surface
ocean.ee	density gradient error on density surface
ocean.ex	x-component of density gradient error on density surface
ocean.ey	y-component of density gradient error on density surface
ocean.curl_e	curl of density gradient error on density surface

Table 3: Continued from Table 2.

variable	**variable name**
ocean.e_hel	diapycnal velocity due to neutral helicity
ocean.e_hel_x	x-component of diapycnal velocity due to neutral helicity
ocean.e_hel_y	y-component of diapycnal velocity due to neutral helicity
ocean.e_therm	diapycnal velocity due to thermobaricity
ocean.e_cab	diapycnal velocity due to cabbeling
ocean.*_trans	transport caused by the respective diapycnal velocity
ocean.*_trans_sum	integral of the transport caused by the resp. dia. velocity
ocean.streamfunc	geostrophic streamfunction
ocean.geo_vel_x	zonal component of geostrophic velocity
ocean.geo_vel_y	meridional component of geostrophic velocity
ocean.*_i	same variables as above but for the optimized surface
ocean.settings	input settings

of state, $\rho = \rho(s, \Theta, p)$ in terms of conservative temperature. This calculation involves moving the parcels to mid pressure, as in Jackett et al. (2006).

- **change.m** - Changes certain values in a matrix to different values according to some relation.
- **cut_off.m** - Cut off data on a surface above a certain depth (e.g. exclude data above the mixed layer).
- **delta_streamfunc.m** - Calculate differences in streamfunctions for optimization of streamfunctions according to McDougall and Klocker (2009).
- **dia_trans.m** - Calculate dia-surface transport and integrate over the area.
- **dyn_height.m** - Calculate dynamic height.
- **e_hel.m** - Calculate the diapycnal velocity ($e^{hel} = \boldsymbol{V} * \boldsymbol{s}$) due to neutral helicity ($\boldsymbol{V}$ is lateral velocity, $\boldsymbol{s}$ is the slope difference between a density surface and the neutral tangent plane).
- **e_therm_cab.m** - Calculate the diapycnal velocities caused by cabbeling and thermobaricity.

- **gamma_ew.m** (*) - Calculate γ^{EW} through an approximate function in terms of salinity and potential temperature according to Eden and Willebrand (1999); This function is constructed for the North Atlantic only.

- **gamma_n_3d.m** - Label three-dimensional s,ct,p field with neutral density (γ^n) according to Jackett and McDougall (1997).

- **gpoly16ct.m** - Calculate neutral density through a rational function (γ^{rf}) in terms of salinity and conservative temperature according to McDougall and Jackett (2005)

- **grad_surf.m** - Find the surface gradient of a variable in x and y-direction.

- **grav.m** - Calculate gravitational acceleration according to Moritz (2000).

- **helicity_pressure.m** (*) - Calculate neutral helicity on a pressure level according to McDougall and Jackett (2007).

- **helicity_surface.m** - Calculate neutral helicity on a density surface according to McDougall and Jackett (2007).

- **input_settings.m** - input_settings.m – GUI for setting input settings.

- **mld.m** - Calculate the mixed-layer depth as in de Boyer Montégut et al. (2004). The criterion selected is a threshold value of density from a surface value ($\Delta\rho = 0.03\ kgm^3$).

- **montgomery_streamfunc.m** (*) - Calculate the Montgomery streamfunction (the acceleration potential) in a steric anomaly surface according to Montgomery (1937) or according to Zhang and Hogg (1992).

- **ns_3d.m** - For a three-dimensional field of hydrographic data (s,ct,p) that has been labelled with a density variable, find the salinities, conservative temperatures and pressures of specified density surfaces.

- **optimize_stream_func.m** - Optimize streamfunction on density surfaces according to McDougall and Klocker (2009).

- **optimize_surface.m** - Optimize density surface using an iterative method to minimize fictitious diapycnal diffusivity according to Klocker et al. (2009a,b).

- **scale_fac.m** - Find distances between gridpoints of given latitude/longitude.

- **slope_error.m** - Calculate slope errors, $\boldsymbol{s}$, density gradient errors, $\boldsymbol{\epsilon}$, their curl and the fictitious diapycnal diffusivity, D^f.

- **var_on_surf.m** - Vertically interpolate one or two variable(s) onto a density surface.

- **z_from_p.m** - Calculate depth from pressure according to Leroy and Parthiot (1998).

9 Future changes to the code

The next planned improvements to this Matlab toolbox are listed below:

- The damping factor λ will be adjusted automatically to maximize the speed of convergence of $|\boldsymbol{\epsilon}|^2$.

- The calculation of ω-surfaces will stop when the decrease in $|\boldsymbol{\epsilon}|^2$ from one step to the next is less than a set value.

- Two-dimensional fields for the isopycnal diffusivities can be used as input instead of the constant value $K = 1000\ m^2 s^{-1}$.

- The calculation of surfaces of constant density will be possible at more than one level at once.

10 License

4. This notice must be included with any full or partial distribution.

[The above is a modified version of the zlib/libpng license, available at www.opensource.org.]

References

de Boyer Montégut, C., G. Madec, A. S. Fischer, A. Lazar, and D. Iudicone, 2004: Mixed layer depth over the global ocean: An examination of profile data and a profile-based climatology. *Journal of Geophysical Research*, **109**, C12 003.

Eden, C. and J. Willebrand, 1999: Neutral density revisited. *Deep Sea Research II*, **46**, 33–54.

Jackett, D. R. and T. J. McDougall, 1997: A neutral density variable for the world's oceans. *Journal of Physical Oceanography*, **27**, 237–263.

Jackett, D. R., T. J. McDougall, R. Feistel, D. G. Wright, and S. M. Griffies, 2006: Algorithms for density, potential temperature, conservative temperature and freezing temperature of seawater. *Journal of Atmospheric and Oceanonic Technology*, **23**, 1709–1728.

Klocker, A. and T. J. McDougall, 2010: Quantifying the ill-defined nature of neutral surfaces, Journal of Physical Oceanography, in press.

Klocker, A., T. J. McDougall, and D. R. Jackett, 2009a: A new method for forming approximately neutral surfaces. *Ocean Science*, **5**, 155–172.

Klocker, A., T. J. McDougall, and D. R. Jackett, 2009b: Corrigendum to 'A new method for forming approximately neutral surfaces' published in Ocean Science, 5, 155-172,2009. *Ocean Science*, **5**, 191–191.

Leroy, C., Claude and F. Parthiot, 1998: Depth-pressure relationships in the oceans and seas. *J. Acoust. Soc. Am.*, **103**, 1346–1352.

McDougall, T. J., 1984: The relative roles of diapycnal and isopycnal mixing on subsurface water mass conversion. *Journal of Physical Oceanography*, **14**, 1577–1589.

McDougall, T. J., 1987: Thermobaricity, cabbeling, and water-mass conversion. *Journal of Geophysical Research*, **92**, 5448–5464.

McDougall, T. J., 2003: Potential enthalpy: A conservative oceanic variable for evaluating heat content and heat fluxes. *Journal of Physical Oceanography*, **33**, 945–963.

McDougall, T. J. and D. R. Jackett, 2005: The material derivative of neutral density. *Journal of Marine Research*, **63**, 159–185.

McDougall, T. J. and D. R. Jackett, 2007: The thinness of the ocean in $S-\Theta-p$ space and the implications for mean diapycnal advection. *Journal of Physical Oceanography*, **37**, 1714–1732.

McDougall, T. J. and A. Klocker, 2009: An approximate geostrophic streamfunction for use in density surfaces, Ocean Modelling, in press.

Montgomery, R. B., 1937: A suggested method for representing gradient flow in isentropic surfaces. *Bull. Amer. Meteor. Soc.*, **18**, 210–212.

Moritz, H., 2000: Geodetic reference system 1980. *Journal of Geodesy*, **74**, 128–133.

Zhang, H.-M. and N. G. Hogg, 1992: Circulation and water mass balance in the Brazil Basin. *Journal of Marine Research*, **50**, 385–420.