Zonally averaged Ocean Model (ZOM)

Carsten Eden, KlimaCampus, Hamburg, Germany

1 Introduction

This documentation describes the numerical implementation of a zonally averaged ocean model (ZOM). It exists in two version, a single layer model (make 1D) which represents a zonally averaged Stommel Arons model and a primitive equation model version (make 2D).

2 A simple closure for the St. Arons model

A representation of the Stommel Arons model including the western boundary layer, where Rayleigh friction is introduced, is given by

$$\frac{\partial}{\partial t}u - fv = -g'\frac{\partial}{\partial x}h - ru \ , \ \frac{\partial}{\partial t}v + fu = -g'\frac{\partial}{\partial y}h - rv \ , \ \frac{\partial}{\partial t}h + H_0\left(\frac{\partial}{\partial y}v + \frac{\partial}{\partial x}u\right) = Q - \lambda h$$

where h denotes the thickness of the lower layer of a two-layer ocean with velocities u and v and mean thickness H_0, Q the prescribed deepwater source and $-\lambda h$ the interior upwelling. Averaging over the western boundary layer yields

$$\frac{\partial}{\partial t}\bar{u}_b - f\bar{v}_b = -g'\frac{\Delta h_b}{\delta} - r\bar{u}_b \ , \ \frac{\partial}{\partial t}\bar{v}_b + f\bar{u}_b = -g'\frac{\partial}{\partial y}\bar{h}_b - r\bar{v}_b \ , \ \frac{\partial}{\partial t}\bar{h}_b + H_0\frac{\partial}{\partial y}\bar{v}_b + \frac{H_0}{\delta}u_\delta = \frac{\bar{Q}}{\epsilon} - \lambda\bar{h}_b$$

where $\bar{u}_b$, $\bar{v}_b$ and $\bar{h}_b$ denote the velocity and thickness averaged over the western boundary layer with width $\delta = r/\beta$ (note that r represents an e-folding scale; the boundary layer could also be defined as $r = 2r/\beta$ or $r = 3r/\beta$). The pressure difference over the western boundary layer $\Delta h_b = h(x = \delta) - h(x = 0)$ and the zonal velocity at the offshore edge of the western boundary layer $u_\delta = u(x = \delta)$ have to be parameterised. We simply assume that

$$\Delta h_b = \gamma_1(\bar{h}_i - \bar{h}_b) \quad , \quad u_\delta = \gamma_2\bar{u}_b \tag{1}$$

with two tuning parameters γ_1 and γ_2 of order one. For the closure for Δh_b we also need the budget for the interior mean thickness $\bar{h}_i$

$$\frac{\partial}{\partial t}\bar{h}_i + H_0\frac{\partial}{\partial y}\bar{v}_i - \frac{H_0}{\Delta x}u_\delta = -\lambda\bar{h}_i \tag{2}$$

where Δx represents the width of the interior basin. For the budget of $\bar{h}_i$ we need to know the interior transport $\bar{v}_i$. The zonally averaged equations for the interior are given by

$$\frac{\partial}{\partial t}\bar{u}_i - f\bar{v}_i = -g'\frac{\Delta h_i}{\Delta} - r\bar{u}_i \ , \ \frac{\partial}{\partial t}\bar{v}_b + f\bar{u}_i = -g'\frac{\partial}{\partial y}\bar{h}_i - r\bar{v}_i \ , \ \frac{\partial}{\partial t}\bar{h}_i + H_0\frac{\partial}{\partial y}\bar{v}_b - \frac{H_0}{\Delta}u_\delta = -\lambda\bar{h}_i$$

with the pressure difference $\Delta h_i = h(x = x\delta) - h(x = x_E)$ over the interior, for which we need a closure. For the interior flow, we anticipate a frictionless Sverdrup regime. Using the meridional derivative of the steady zonal momentum budget

$$f\frac{\partial}{\partial y}\bar{v}_i = (g'/\Delta)\frac{\partial}{\partial y}\Delta h_i - \beta\bar{v}_i + r\frac{\partial}{\partial y}\bar{u}_i \tag{3}$$

in the steady thickness budget only yields the anticipated form

$$f\frac{\partial}{\partial y}\bar{v}_b - (f/\Delta)u_\delta = -(f/H_0)\lambda\bar{h}_i \overset{!}{=} -\beta\bar{v}_i \tag{4}$$

i.e. a frictionless Sverdrup balance in the interior if we use

$$g'\frac{\partial}{\partial y}\Delta h_i = fu_\delta - r\Delta\frac{\partial}{\partial y}\bar{u}_i \tag{5}$$

or by meridionally integrating

$$g'\Delta h_i = g'\Delta h_i(y = 0) + \int_0^y fu_\delta dy - r\Delta(\bar{u}_i - \bar{u}_i(y = 0)) \tag{6}$$

as closure for Δh_i. Note that from the steady zonal momentum balance at the equation we find $g'\Delta h_i(y = 0) = -r\Delta\bar{u}_i(y = 0)$. In the numerical code, the grid and all parameters have to be set. Further, the location and strength of the deepwater formation Q has to be supplied.

3 Application to primitive equations

3.1 Zonally averaged primitive equations

The primitive equations averaged over the western boundary layer and also over the interior are given by

$$\begin{aligned}
\frac{\partial}{\partial t}\bar{u}_{b,i} &= f\bar{v}_{b,i} - \Delta p_{b,i}/(\delta, \Delta x) + \bar{F}^u_{b,i} &(7)\\
\frac{\partial}{\partial t}\bar{v}_{b,i} &= -f\bar{u}_{b,i} - \frac{\partial}{\partial y}\bar{p}_{b,i} + \bar{F}^u_{b,i} &(8)\\
\frac{\partial}{\partial t}\bar{b}_{b,i} + \frac{\partial}{\partial y}\bar{b}_{b,i}\bar{v}_{b,i} + \frac{\partial}{\partial z}\bar{w}_{b,i}\bar{b}_{b,i} &= \frac{\partial}{\partial z}K\frac{\partial}{\partial z}\bar{b}_{b,i} - u_\delta\bar{b}_{b,i}/(\delta, -\Delta x) - \frac{\partial}{\partial y}\overline{b'_{b,i}v'_{b,i}} &(9)\\
\frac{\partial}{\partial y}\bar{v}_{b,i} + \frac{\partial}{\partial z}\bar{w}_{b,i} &= -u_\delta/(\delta, -\Delta x) &(10)
\end{aligned}$$

Friction is contained in $F^u_{b,i}$ and $F^v_{b,i}$ respectively and is not further specified here. In the numerical code, Rayleigh friction, bottom friction and harmonic lateral and vertical friction are implemented. The parameterisations

$$\Delta p_b = \gamma_1(\bar{p}_i - \bar{p}_b) \quad , \quad u_\delta = \gamma_2\bar{u}_b \quad , \quad \Delta p_i = \Delta p_i(y = 0) + \int_0^y fu_\delta dy' \tag{11}$$

will be used, which are analogous to the layered model. $\Delta p_i(y=0)$ is found using the steady interior zonal momentum balance at the equator. The eddy fluxes $\frac{\partial}{\partial y}\overline{b'_{b,i}v'_{b,i}}$ will be neglected. Convection is parameterised by locally increasing the vertical diffusivity K to $1000\,m^2/s$ in case of unstable stratification.

3.2 Barotropic mode and surface pressure

To find the barotropic solution, we split the pressure $\bar p_{b,i}$ into a hydrostatic and a surface part, i.e.

$$\bar p_{b,i} = \bar p^s(x,y)_{b,i} + \bar p^h_{b,i} \quad \text{with} \quad \bar p^h_{b,i} = -\int_z^0 \bar b_{b,i}\,dz \tag{12}$$

and similar for $\Delta p_{b,i}$. Note that for both boundary layer and interior pressure and buoyancy, the hydrostatic relation applies. We now vertically average the momentum equations

$$\begin{aligned} \frac{\partial}{\partial t}\int_{-h}^0 \bar u_b dz &= \int_{-h}^0 U_b dz - h\Delta p^s_b/\delta \qquad (13) \\ \frac{\partial}{\partial t}\int_{-h}^0 \bar v_{b,i} dz &= \int_{-h}^0 V_{b,i} dz - h\frac{\partial}{\partial y}\bar p^s_{b,i} \qquad (14) \end{aligned}$$

with the momentum tendencies

$$U_b = f\bar v_b - \Delta p^h_b/\delta + \bar F^u_b \text{ and } V_{b,i} = -f\bar u_{b,i} - \frac{\partial}{\partial y}\bar p^h_{b,i} + \bar F^v_{b,i} \tag{15}$$

excluding the surface pressure components. Now take the appropriate divergences implied by the continuity equations for boundary and interior.

$$\begin{aligned} \delta\frac{\partial}{\partial y}(\int V_b dz - h\frac{\partial}{\partial y}\bar p^s_b) + \gamma_2(\int U_b dz - h\Delta p^s_b/\delta) &= 0 \qquad (16) \\ \Delta x\frac{\partial}{\partial y}(\int V_i dz - h\frac{\partial}{\partial y}\bar p^s_i) - \gamma_2(\int U_b dz - h\Delta p^s_b/\delta) &= 0 \qquad (17) \end{aligned}$$

Adding yields a diagnostic relation for the zonally integrated surface pressure $\bar P^s = \delta\bar p^s_b + \Delta x\bar p^s_i$

$$\frac{\partial}{\partial y}h\frac{\partial}{\partial y}\bar P^s = \frac{\partial}{\partial y}\int_0^h dz(\delta V_b + \Delta x V_i) \tag{18}$$

Substracting yields a relation for the difference between surface boundary and interior pressure $dp^s = \bar p^s_i - \bar p^s_b = \Delta p^s_b/\gamma_1$

$$\frac{\partial}{\partial y}h\frac{\partial}{\partial y}dp^s - \gamma_2 h\gamma_1 dp^s/\delta(1/\Delta x + 1/\delta) = \frac{\partial}{\partial y}\int_h^0 dz(V_i - V_b) - \gamma_2(1/\Delta x + 1/\delta)\int_h^0 dz U_b$$

From $\bar P^s$ and dp^s one can obtain interior and boundary surface pressures. Both equations are solved using a conjugate gradient algorithm, which is also used in the fully resolved primitive equation model version CPFLAME.

3.3 Wind stress forcing

Wind stress forcing can be incorporated in the usual way, i.e. as upper boundary flux condition for the vertical friction terms contained in $F^u_{b,i}$ and $F^v_{b,i}$. Further, the surface pressure can be found as before.

3.4 Southern Ocean and eddy parameterisation

In the Southern Ocean we can zonally average the equations and no zonal pressure difference shows up. On the other hand, meso-scale eddies drive an important part of the overturning circulation. This process can be accounted for by the Gent and McWilliams parameterisation introducing the isopycnal thickness diffusivity K. The zonally averaged primitive equations become

$$\frac{\partial}{\partial t}\bar{u} = f\bar{v} + \frac{\partial}{\partial z}\left(\frac{K_{gm}f^2}{N^2}\frac{\partial}{\partial z}\bar{u}\right) + \bar{F}^u_{b,i} \tag{19}$$

$$\frac{\partial}{\partial t}\bar{v} = -f\bar{u} + \frac{\partial}{\partial z}\left(\frac{K_{gm}f^2}{N^2}\frac{\partial}{\partial z}\bar{v}\right) - \frac{\partial}{\partial y}\bar{p} + \bar{F}^u_{b,i} \tag{20}$$

$$\frac{\partial}{\partial t}\bar{b} + \frac{\partial}{\partial y}\bar{b}\bar{v} + \frac{\partial}{\partial z}\bar{w}\bar{b} = \frac{\partial}{\partial z}K\frac{\partial}{\partial z}\bar{b} \tag{21}$$

This is the residual mean formulation where the velocities are given by the sum of the Eulerian mean and the (parameterised) eddy-driven velocities. The residual mean St. Arons model for $\bar{v}_d$ becomes

$$\frac{\partial}{\partial t}\bar{u} - f\bar{v} = -(r + \frac{K_{gm}f^2}{N^2H_0^2})\bar{u}\ ,\ \frac{\partial}{\partial t}\bar{v} + f\bar{u} = -g'\frac{\partial}{\partial y}\bar{h}_b - (r + \frac{K_{gm}f^2}{N^2H_0^2})\bar{v}\ ,\ \frac{\partial}{\partial t}\bar{h} + H_0\frac{\partial}{\partial y}\bar{v} = \bar{Q} - \lambda\bar{h}$$

Note that the large thickness diffusivity K_{gm} puts a constrain on the stability of the time stepping in the St. Arons model and the primitive equations. The large vertical viscosity in the primitive equation need implicit treatment of this term.

In the numerical code, we simply set the zonal pressure gradients in the western boundary layer and the interior of the Southern Ocean to zero, and we also set $u_\delta = 0$. The Southern ocean works only for the prognostic interior at the moment both for the St. Arons and the primitive equation model.

3.5 Lateral friction

The zonal mean of lateral friction in the western boundary layer and the interior is given by

$$\bar{F}^u_{b,i} = A_{h,x}\overline{\frac{\partial^2}{\partial x^2}u_{b,i}} + A_h\overline{\frac{\partial^2}{\partial y^2}u_{b,i}} = A_{h,x}(\frac{\partial}{\partial x}u_{\delta,E} - \frac{\partial}{\partial x}u_{W,\delta})/(\delta,\Delta x) + A_h\frac{\partial}{\partial y}\bar{u}_{b,i} \tag{22}$$

Approximating $\frac{\partial}{\partial x}u_\delta = (\bar{u}_i - \bar{u}_b)/\delta$ and $\frac{\partial}{\partial x}u_W = \bar{u}_b/\delta$ and $\frac{\partial}{\partial x}u_E = 0$ yields

$$\bar{F}^u_{b,i} \;=\; A_{h,x}(\bar{u}_i - 2\bar{u}_b, \bar{u}_i - \bar{u}_b)/(\delta^2, -\delta\Delta x) + A_h \frac{\partial}{\partial y}\bar{u}_{b,i} \tag{23}$$

Assuming free-slip for v yields accordingly

$$\bar{F}^v_{b,i} \;=\; A_{h,x}(\bar{v}_i - \bar{v}_b)/(\delta^2, -\delta\Delta x) + A_h \frac{\partial}{\partial y}\bar{v}_{b,i} \tag{24}$$

Note that lateral friction works for the prognostic interior version only. Note also that the parameter A_h is set independently from $A_{h,x}$ in the code.

3.6 Salinity

Salinity S can be enabled by a preprocessor switch. It is then calculated as an prognostic variable in the western boundary layer and the interior. The buoyancy variables take then the physical meaning of temperature T and buoyancy b is calculated by

$$b = \alpha T + \beta S \tag{25}$$

where the constant α and β are set in the numerical code. Further, isopycnal diffusion is enabled in that case with isoypcnal diffusivity K_{iso}. While temperature (and buoyancy) is forced by a relaxation condition at the surface, salinity can be forced by a fixed flux or a relaxation condition.

4 Directory structure and model configuration

4.1 Directory structure

The numerical code is distributed over several directories. Here is a list of these directories and a short characterisation of its contents.

- `./shallow`: the code for the zonally averaged St. Arons model
- `./deep`: the code for zonally averaged primitive equation model
- `./include`: contains a single top-level include file with CPP directives
- `./mpp`: (most of the) code specific to parallelisation
- `./misc_modules`: more general and multi purpose code
- `./configure`: contains alternatives for compiler specifications
- `./doc`: contains documentation
- `./bin`: contains executable after successful compilation

The top-level program of both models is `model.F`. Here, the work flow of the main routines can be seen.

4.2 Model configuration

Model configuration is done by choosing model parameters and surface forcing. All parameters and forcing functions are set in template files which are included at the top in the files `./shallow/model.F` and `./deep/model.F` respectively. These template files are named `EXP_setup.F`, where "EXP" is a name for the respective experiment, and are located in the directory `./shallow` and `./deep`, respectively, and contain the subroutines `set_parameter`, `setup_forcing` and `setup_topography`. These subroutines specify the model parameters, the surface forcing and the topography for the experiment. Examples for these templates files are given.

Choose the option of prognostic or diagnostic model for the interior by the preprocessor switch in `./include/options.inc`. It is also possible to choose between different advection schemes used in the primitive equation model. Further, the vertical viscosity according to the residual mean formalism can be chosen.

4.3 Installation

4.3.1 Quick start

This short installation procedure assumes a Unix system and that all libraries and compilers are available. See below for necessary prerequisites in case of problems to compile.

- Make a new directory where the model code shall be placed. That directory is called `dir` hereafter.
- Extract the tar-archive in `dir`.
- Edit the file `Makefile`. The variable `MODEL` at the head of `Makefile` has to be set to `dir`.
- Copy one of the files in `dir/configure` to a (new) file `dir/Makefile_host` depending on the specific system and compiler.
- Type `make 1D` or `make 2D`. If something went wrong it is necessary to modify `dir/Makefile_host` as described below.
- Find the executable `model.x` in `dir/bin`.
- Run the executable, investigate the output and change model setup.

4.3.2 Detailed instructions

If the above procedure did not work in order to compile the model code, it is most likely that the file `dir/Makefile_host` has to be modified. This file is automatically included in the main Makefile `dir/Makefile` and in each of the Makefiles in the subdirectories. The file `Makefile` is read by the command `make` to obtain informations how and in which order to compile the code. The file `dir/Makefile_host` contains system and compiler specific details, which are supplied to `Makefile` (and `make`) and are explained in this section.

In order to compile the code a working Fortran90 compiler is needed. Examples are the freely available GNU `gfortran` or Intel's `ifort`. The compiler name (i.e. the way to invoke it) is specified in `dir/Makefile_host` by the variable `F90`. In addition to the Fortran90 compiler, a C compiler and a CPP compiler are also needed. Their names have to be specified by the variables `CC` and `CPP` in `dir/Makefile_host` respectively. Usually `CC = cc` and `CPP = ${F90} -E` should work. For linking of all compiled files a linker has to be specified by the variable `LINKER` which is usually identical to the Fortran90 compiler. Specific options can be passed to the Fortran90 and C compiler by the variables `F90_OPTS` and `CC_OPTS`. Note that it is assumed that the model in run in double precision, i.e. that real variables in Fortran90 code are represented by 8 bytes. Usually, this has to be specified to the Fortran90 compiler by a option in `F90_OPTS`.

The other important prerequisite for the model is the NetCDF library, which is freely available at `http://www.unidata.ucar.edu/software/netcdf`. However, on most system it should be already installed. The location of the library has to be specified in `dir/Makefile_host` by the variable `NETCDF_LIB_DIR` (location of the file `libnetcdf.a`) and `NETCDF_INCL_DIR` (location of the file `netcdf.inc`) which can be usually set to `/usr/lib` and `/usr/include` respectively.