ICTP Ocean Modelling Summer School Lab
CVMix Vertical Mixing Library

CVMix Team:
M. Levy[1], G. Danabasoglu[1], S. Griffies[2], T. Ringler[3], A. Adcroft[2], R. Hallberg[2], D. Jacobsen[3], and W. Large[1]

[1]**National Center for Atmospheric Research**

[2]**Geophysical Fluid Dynamics Laboratory**

[3]**Los Alamos National Laboratory**

ICTP Summer School
Ankara, Turkey
September 28 - Oct 1, 2015

Outline

Community Earth System Model (CESM)

Community Earth System Model

About

A flexible global climate model

- Couples atmosphere, ocean, land, and sea ice components.
- CESM 1.2 was released in June 2013 (CESM 1.2.2 in June 2014)
- CESM 2.0 will be available in June 2016 (I think)

NCAR's coupler + community's component models; lots of component development done at NCAR as well

How flexible?

- Active components: CAM [atmosphere], POP [ocean], CLM [land], CICE [sea ice]
- Provides data models (for example - force POP with atmospheric data instead of an active model)
- Also provides "stub" models (for example, running CLM with data atmosphere does not require any ocean or sea ice forcing)

Running CESM

Four commands to run CESM on a supported machine

Run the following from `$CESMROOT/scripts`**:**

(1) `./create_newcase -case $CASEDIR/$CASENAME -compset $COMPSET -res $RESOLUTION -mach $MACHINE`

Run the following from `$CASEDIR`**:**

(2) `./cesm_setup`

(3) `./$CASENAME.build`

(4) `./$CASENAME.submit`

What will this run?

The default setup is a 5-day simulation; output will depend on the components selected.

More Information

Coming up

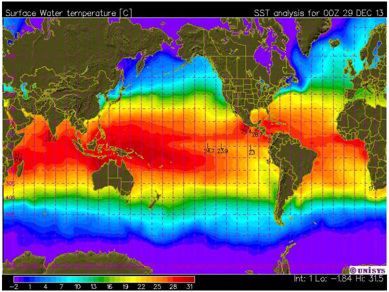
- Later in this talk, I will talk about setting up a couple of ocean-only runs on `argo`
  - I'll point you to `$CESMROOT`
  - I'll tell you that ocean is the only active component in a "C" compset (data atmosphere and sea ice, stub land)
- For general information, see http://www.cesm.ucar.edu/models/cesm1.2/

But first...

... Let's talk about vertical mixing and CVMix

The Community Vertical Mixing Project

Why is Vertical Mixing Important to Ocean Models?



http://weather.unisys.com/archive/sst/sst-131229.gif (Dec 29, 2013)

Basics

- Sea surface temperature (SST) has a major role in atmosphere ↔ ocean energy exchange
- Vertical mixing is one of many processes affecting SST
  - Occurs on scales that are not resolved by current ocean models, need to use parameterization instead
- Other physical quantities (tracers, salinity, etc) also affected by mixing

Mixing in Ocean Models

Current state

- Numerous techniques for parameterizing the mixing process
- Model developers choose their favorite parameterization(s) and code them up as part of the ocean model

CVMix project

- Our goal: produce an easy-to-use library containing a range of parameterizations
- Secondary goal: provide a stand-alone driver to test the library on its own
  - Note: we use the term "stand-alone driver" a bit loosely. CVMix can compute single-column diffusivities given proper input, but lacks the capability to see how diffusivities change over time.

Why CVMix?

Driving force

CESM Workshop (Breckenridge) 2012: MPAS-O (ocean model from LANL) did not have a KPP module yet and MOM5 (from GFDL) was using an outdated implementation – lab wanted to improve on for MOM6.

- CVMix is now used in development of MPAS-O and MOM6, and will [eventually] replace the mixing modules in POP.

Other benefits

1. Reduce duplicate code – for example, static mixing occurs as a step in many parameterizations
2. CSEG is working to include non-POP / non-data ocean models in CESM
    - Vertical mixing library allows [some] physics to stay the same even if dynamics change
    - Allow more detailed model inter-comparisons

The CVMix Mission

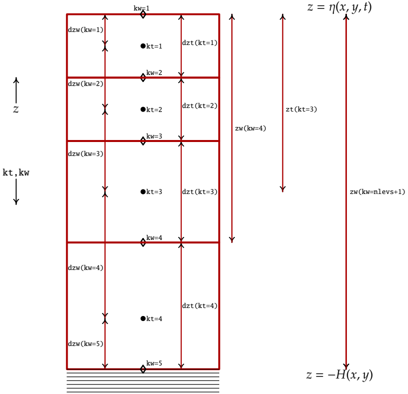
CVMix will...

- Provide a transparent, robust, flexible, well documented, open source library for use in parameterizing ocean vertical mixing processes.
- Contain a consensus of first-order closures that return a vertical diffusivity, viscosity, and possibly a non-local transport.
- Be comprised of Fortran modules that may be used in a stand-alone manner or incorporated into ocean models.
- Be developed within a community of scientists and engineers who make use of CVMix modules for a variety of research needs.

CVMix modules will be freely distributed under GPLv2 using an open source methodology.

Vertical Mixing Overview

- Divide column into `nlevs` levels
- Data on cell centers and interfaces
- Center index $kt = 1 \ldots$ `nlevs`
- Interface index $kw = 1 \ldots$ `nlevs+1`
- Depth z
  - η at surface
  - 0 at average sea level
  - $-H(x,y)$ at bottom (positive up!)



What does a vertical mixing parameterization look like?

- Inputs: combination of parameters and physical values in column
- Outputs: viscosity (ν) and tracer diffusivity (κ) coefficients on cell interfaces

[Some] Mixing Parameterizations

1. **Static background mixing**
   - Constant mixing (Ekman, 1905)
   - Bryan-Lewis (1979)
   - Henyey et al. (1986)
2. **Tidal mixing**
   - Simmons et al. (2004)
   - Polzin (2009) / Melet et al. (2013)
3. **Shear-induced mixing** ("Richardson number mixing")
   - Pacanowski and Philander (1981)
   - Large et al. (1994), henceforth LMD94
   - Jackson et al. (2008)
4. **Double diffusion mixing** (Schmitt, 1994 / LMD94 / Danabasoglu et al., 2006)
5. **K-profile parameterization** ("KPP"; LMD94)
6. **Convective adjustment** (density based as well as Brunt-Väisälä)

Blue indicates method is already in package.

More Parameterizations on the Way

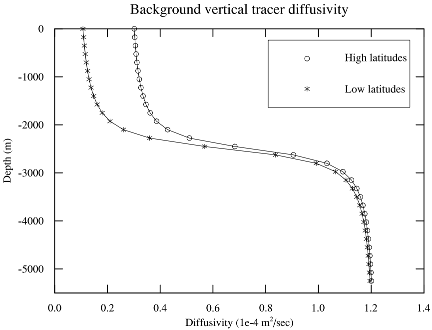
Various stages of progress

- Langmuir (surface wave) mixing – brought in to KPP by group at Brown University, currently in testing
- KPP bottom surface layer – mentioned by Enrique in a couple of conversations, maybe on a to-do list?
- Your favorite scheme here – this is a community model!

Bryan-Lewis Profile

Want diffusivity to increase towards bottom of ocean (mimic tidal mixing).

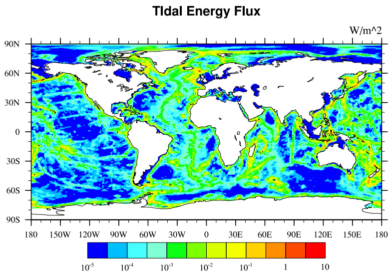
At right: diffusivity profile of two columns representing columns in different latitudes.



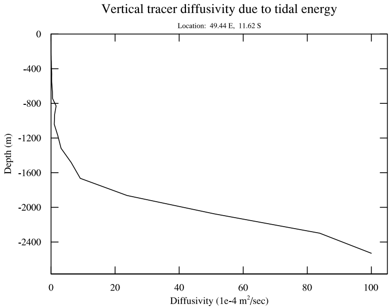
Diffusivity and viscosity depend on depth

$$
\begin{aligned}
\kappa &= c_0 + \frac{c_1}{\pi}\tan^{-1}\left(c_2((-z) - c_3)\right) \\
\nu &= \mathrm{Pr}\kappa
\end{aligned}
$$

Tidal Mixing



Tidal Energy Flux Map



Diffusivity North of Madagascar

Diffusivity depends on tidal energy flux, depth, density, and buoyancy

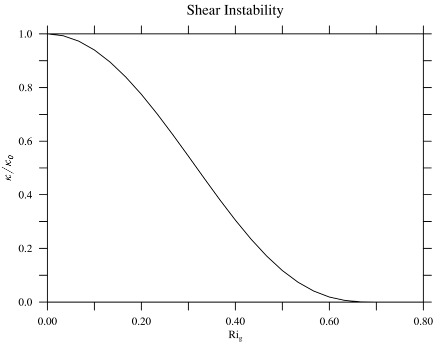
$$\kappa = \frac{q\Gamma E(x,y)F(x,y,z)}{\rho N^2}$$

$$F(x,y,z) = \frac{e^{-z/\zeta}}{\zeta\left(e^{H(x,y)/\zeta} - e^{-\eta(x,y)/\zeta}\right)}$$

Shear Mixing

Want diffusivity to decrease as Richardson number (Ri) increases; $\kappa = 0$ if Ri $\geq$ $\text{Ri}_0 = 0.7$.

At right: The stand-alone driver produces the shear mixing diffusivity profile plot from Fig. 3 of LMD94.



Diffusivity and viscosity depend on Richardson number

$$
\begin{aligned}
\kappa &= \begin{cases} \kappa_0 & \text{Ri} \leq 0 \\ \kappa_0\left[1 - (\text{Ri}/\text{Ri}_0)^{p_1}\right]^{p_2} & 0 < \text{Ri} < \text{Ri}_0 \\ 0 & \text{Ri} \geq \text{Ri}_0 \end{cases} \\
\nu &= \text{Pr}\kappa
\end{aligned}
$$

Double Diffusion Mixing

Two regimes

Determine which regime we are in via stratification parameter

$$\mathrm{R}_\rho = \frac{\alpha}{\beta}\left(\frac{\partial\Theta/\partial z}{\partial S/\partial z}\right),$$

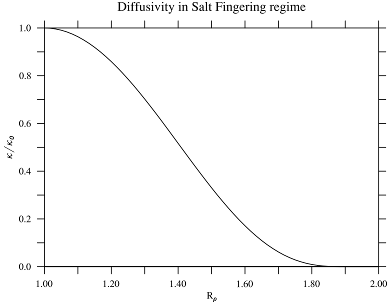
where α is the thermal expansion coefficient and β is the haline contraction coefficient:

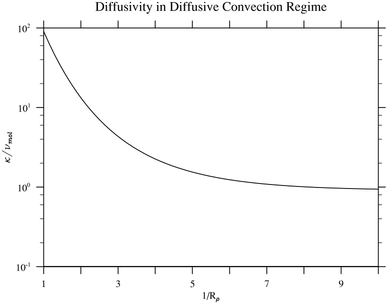
1. Salt Fingering ($\partial S/\partial z > 0$ and $1 < \mathrm{R}_\rho < \mathrm{R}_\rho^0$); salt water above fresher water $\Rightarrow$ salt water will sink
2. Diffusive Convective Instability ($\partial\Theta/\partial z < 0$ and $0 < \mathrm{R}_\rho < 1$); cold water above warm water $\Rightarrow$ cold water will sink

And that's not all...

Double diffusion also introduces idea of different diffusivity for temperature and salinity (κ_Θ and κ_S, respectively).

Double Diffusion Mixing





Diffusivity profiles for the two regimes (Fig. 4 in LMD94).

Salt-fingering regime

$$\kappa_S = \kappa_0 \left[1 - \left(\frac{R_\rho - 1}{R_\rho^0 - 1} \right)^{p_1} \right]^{p_2}$$

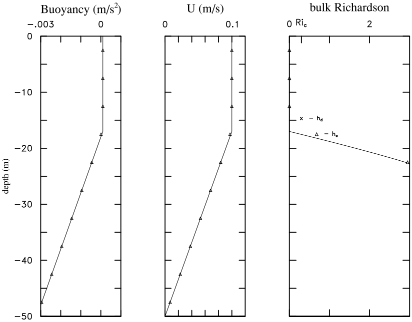
$$\kappa_\Theta = 0.7 \kappa_S$$

Diffusive convective regime

$$\kappa_\Theta = \nu_{\text{mol}} \cdot c_1 \exp\left(c_2 \exp\left[c_3 \frac{1 - R_\rho}{R_\rho} \right] \right)$$

$$\kappa_S = \max(0.15 R_\rho, 1.85 R_\rho - 0.85) \kappa_\Theta$$

KPP Mixing



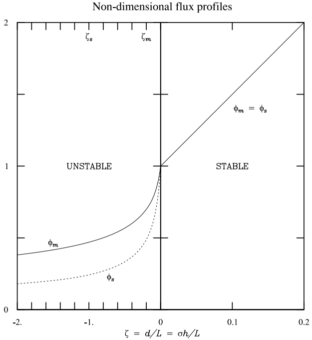
Bouyancy, velocity, and bulk Richardson values (Fig. C1 in LMD94).

The boundary layer depth (h) computed based on bulk Richardson number

$$Ri_b(z) = \frac{-z(B_r - B(z))}{|\mathbf{V}_r - \mathbf{V}(z)|^2 + \mathbf{V}_t^2(z)}$$

$\mathbf{V}_t$ is unresolved velocity shear, see Eq. (23) in LMD94.

KPP Mixing



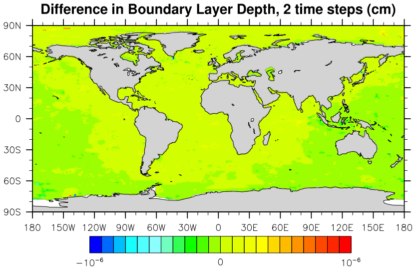
Flux profiles ϕ (Fig. B1 in LMD94).

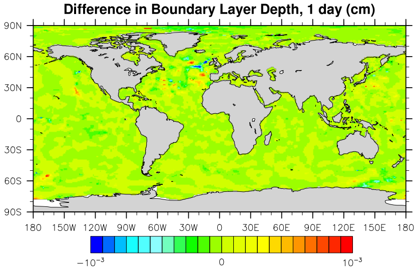
Inside the boundary layer, diffusivity is given by

$$\nu|\kappa = h w_{m|s}(-z/h) G(-z/h)$$

$w_{m|s}$, the turbulent velocity scale for *m*omentum or *s*calar quantities, is inversely proportional to $\phi_{m|s}$; G is a shape function defined to ensure a smooth κ.

CVMix in POP





POP vs POP+CVMix (slide out of date)

What part of CVMix is available in POP?

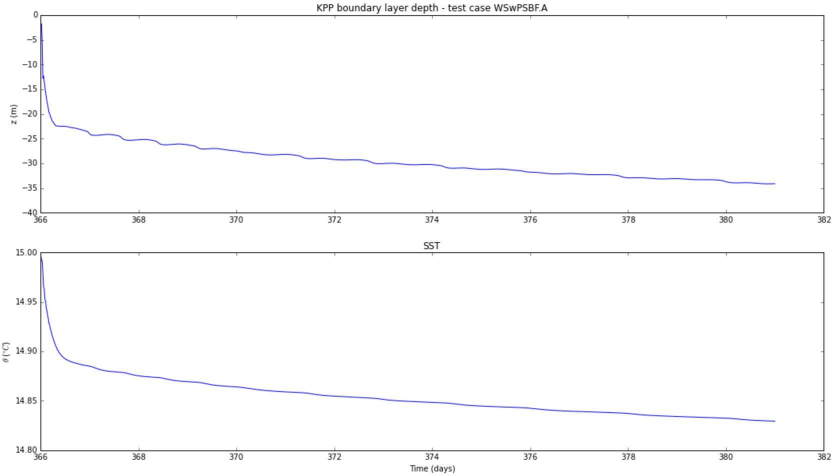
- Just used in KPP boundary layer (no internal mixing yet)
  1. CVMix is used to compute bulk Richardson number and boundary layer depth
  2. POP computes internal mixing coefficients
  3. CVMix updates mixing coefficients inside boundary layer

An Idealized Testcase for 1D Mixing

Testcase setup

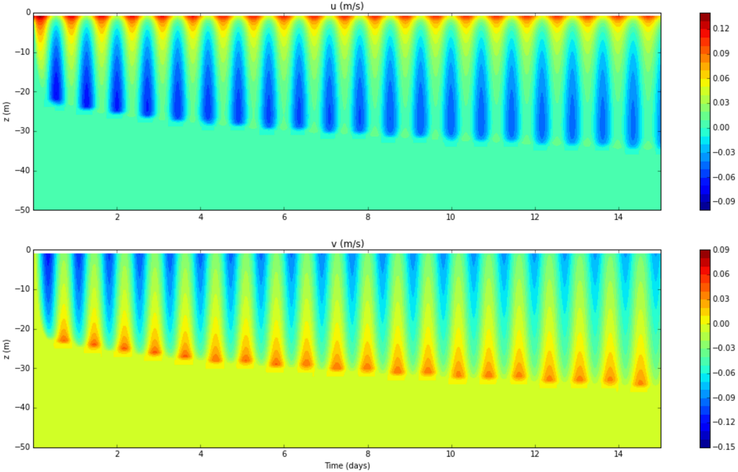
- 400 m column of water, 1m vertical resolution ($\Rightarrow$ 400 levels)
- Initial conditions
  - Temperature is 15° at surface and 11° at bottom (decreasing linearly)
  - Salinity is 35 psu throughout
  - Velocities (u, v) are set to 0
- Surface forcing:
  - Wind stress is 0.1 N/m^2 in zonal direction, 0 meridonal
  - Surface heat flux is 0 W/m^2
- Coriolis: $f = 10^{-4}$ sec^{-1}
- No mixing except for KPP
- Run for 15 days with 20 minute time step

Output - Boundary Layer Depth and SST



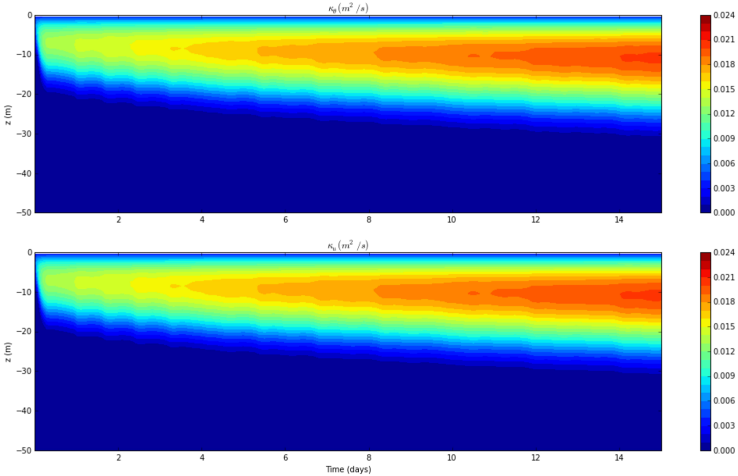
Top - boundary layer depth; Bottom - SST

Output - Velocity



Top - zonal velocity; Bottom - meridonal velocity

Output - Diffusivity



Top - Temperature diffusivity; Bottom - momentum diffusivity

Lab Exercises

Filling in Specifics from an Earlier Slide

First step for running CESM

Run the following from `$CESMROOT/scripts`:

(1) `./create_newcase -case $CASEDIR/$CASENAME -compset $COMPSET -res $RESOLUTION -mach $MACHINE`

Variable definitions

- `$CESMROOT` = `/home/mlevy/codes/cesm1_2_2.cvmix`
- `$CASEDIR` is up to you, but I recommend something like `~/cases/`
- `$CASENAME` is also up to you, but be descriptive
  Pro tip: Useful to include compset and resolution
- `$COMPSET` = *C*
- `$MACHINE` = *argo*
- `$RESOLUTION` varies – global experiments use *T62_g37* (nominal ocean resolution of 3°, displaced pole); 1D experiments use *1D_1D*

A Little Diversion Regarding Resolutions

Understanding the '-res' option

- CESM assumes you are running components on two grid
  1. Atmosphere and land models share a grid
  2. Ocean and sea ice share a grid

  Specify resolution of the form *[atm]_[ocn]*
- You can run the atmosphere and land on different grids by specifying *[atm]_[lnd]_[ocn]* but ocean and sea ice MUST use same grid
- The 'production' ocean grid is *g16*, nominal 1° displaced pole but the 3° grid is much cheaper $\Rightarrow$ better for testing / teaching
- The *1D* ocean grid is not part of the CESM release – Gokhan and I created it to make it easier to test CVMix
  - Turns off horizontal mixing / other horizontal forcings
  - Tracers and velocity are collocated (staggered in global resolution)
  - Uses linear equations of state
- *T62* is the "native" resolution of atmospheric forcing data

Effect of 1D Resolution on Namelist

KPP namelist in global resolution

```
&vmix_kpp_nml
bckgrnd_vdc1 = 0.16
bckgrnd_vdc2 = 0.0
bckgrnd_vdc_ban = 1.0
bckgrnd_vdc_dpth = 1000.0e02
bckgrnd_vdc_eq = 0.01
bckgrnd_vdc_linv = 4.5e-05
bckgrnd_vdc_psim = 0.13
lcheckekmo = .false.
lcvmix = .false.
ldbl_diff = .true.
lhoriz_varying_bckgrnd = .true.
linertial = .false.
llangmuir = .false.
lrich = .true.
lshort_wave = .true.
num_v_smooth_ri = 1
prandtl = 10.0
rich_mix = 50.0
/
```

Effect of 1D Resolution on Namelist

KPP namelist in single column resolution

```
&vmix_kpp_nml
bckgrnd_vdc1 = 0.0
bckgrnd_vdc2 = 0.0
bckgrnd_vdc_ban = 0.0
bckgrnd_vdc_dpth = 0.0
bckgrnd_vdc_eq = 0.0
bckgrnd_vdc_linv = 4.5e-05
bckgrnd_vdc_psim = 0.0
lcheckekmo = .false.
```

lcvmix = .false.

```
ldbl_diff = .false.
lhoriz_varying_bckgrnd = .false.
linertial = .false.
llangmuir = .false.
lrich = .false.
lshort_wave = .true.
num_v_smooth_ri = 1
prandtl = 10.0
rich_mix = 50.0
/
```

Effect of 1D Resolution on Namelist

Also added a new namelist

	Description
`&pop1d_nml`	
`global_shf_coeff = 0.0`	Surface heat flux (W/m^2)
`global_taux = 0.1`	τ_x (N/m^2)
`lcompute_coriolis = .false.`	if true, $f = 2\Omega\sin(\theta)$
`single_col_coriolis = 0.0001`	otherwise, this is f
`single_col_lat = 0.0`	latitude of single column
`single_col_lon = 0.0`	longitude of single column
`/`	(must match domain file!)

Apologies

I should not have used `global_*` for variable names, that's a terrible convention! The first implementation of 1D model was actually run on a 3° grid producing identical columns; I should have changed these to `single_col_*` for the new grid.

Configuring and Building Your Case

Issue commands from $CASEDIR

1. Run `./cesm_setup`
2. Make build-time changes
   - CESM build settings are in `env_build.xml`
   - Can modify CESM component source code by copying module to `SourceMods/src.`*[comp]*`/` and editing in your case directory
3. Build the model by running `./$CASENAME.build`
4. Make run-time changes
   - Component namelists are changed by adding entries to `user_nl_`*[comp]* (after change run `preview_namelists`)
   - Read-only copies of component namelists in `CaseDocs/` (named `*_in`)
   - CESM settings are in `env_run.xml`
     Pro tip: You don't need to edit the text file directly! Instead run `./xmlchange VARIABLE=new_value`
5. Run the model with `./$CASENAME.submit`
   - Output will be in `~/scratch/$CASENAME/run`

Running on argo

Logging in

Two step process:

1. `ssh -XY [username]@ssh.ictp.it`
2. `ssh -XY argo.ictp.it`

Parallel jobs

- We have access to the `esp_guest` queue
  - 4 nodes ⇒ only 4 jobs running simultaneously
  - Run `showfree esp_guest` to see how many nodes are available
  - Run `qstat -u $USERNAME` to see the status of your jobs
    `Q` for "in the queue", `R` for "running"
  - Run `qstat -r esp_guest` to see what is running in the guest queue
- Most exercises should run in 5 or 10 minutes
- Short runtime + single node job means we can also squeeze into `testing` queue (1 more node)

ODTÜ Wi-Fi Password

Connecting to the wireless

Network name: `ng2k`

Username: `ozsoy`

Password: `ozzo123`

Visualization Tips

Loading tools on argo

- `ncview` should be in your path by default
- Run `module load nco` to add the netCDF operators – we specifically want `ncdiff` to look at differences between two cases
- For 1D: there is a very basic NCL script in `/home/mlevy/1D_scripts`
  1. Copy to your home directory and edit case name.
  2. Before you run for the first time, run `module load ncl`
  3. execute with `ncl 1D\ visualization.ncl`

Looking at output locally

- If you have `ipython` on your laptop, there is a notebook designed for looking at 1D output available via git by cloning `git@github.com:mnlevy1981/1D_POP_output_ICTP.git`

Visualization Tips

Variables of interest

- HBLT – Boundary layer depth
- RI_BULK – Bulk Richardson number
- VVC, VDC_T, VDC_S – Diffusivities (momentum, temperature, salinity)
- TEMP, SALT, UVEL, VVEL – Temperature, salinity, velocity components

Default output (resolution-dependent)

- Look in ~/scratch/$CASENAME/run/
- Global exercises
  - Monthly average of 3D fields
  - Output stored in $CASENAME.pop.h.0001-01.nc
- Single column exercise
  - Output every timestep (20 minutes)
  - Output stored in $CASENAME.pop.h.0001-01-02-01200.nc

Exercise #1 – Global Control

About the test

In this test, we run a global ocean model for one month.

1. Create a new case using the following settings
   1. Component Set: *C*
   2. Resolution: *T62_g37*
2. The default simulation is 5 days long, change this to 1 month:

   `./xmlchange STOP_N=1,STOP_OPTION=nmonths`
3. Build and run the model

Exercise #2 – Use CVMix for KPP

About the test

In this test, we use the CVMix library's version of KPP instead of the POP-specific implementation.

1. Create a clone of your case from exercise #1 (See next slide!)
2. Make the necessary change to `user_nl_pop2` to enable CVMix for the boundary layer computation
   Hint: look for `lcvmix` *in* `CaseDocs/pop2_in`
3. Build and run the model for one month. Verify cloning the case brought over the changes from `env_run.xml` by running

   ```
   ./xmlquery STOP_N,STOP_OPTION
   ```

4. How are the results different from those of exercise #1?
   Note: In the POP history files, the variable `HBLT` *is boundary layer depth. What other variables might be interesting to look at?*

How to Use `create_clone`

1. Instead of `./create_newcase`, run `./create_clone -clone $CASEDIR/$OLDCASENAME -case $CASEDIR/$NEWCASENAME`

   Copies `env_*.xml` and user_nl_*
2. In `$CASEDIR/$NEWCASENAME`, run `./cesm_setup`
3. Time-saving hint: if you are not making changes to the source code, you can use the same executable (otherwise you need to run $NEWCASENAME.build). To re-use the executable:
   1. In `$CASEDIR/$OLDCASENAME`, run `./xmlquery EXEROOT` and copy the executable directory to your clipboard
   2. In `$CASEDIR/$NEWCASENAME`, run `./xmlchange EXEROOT=`*`[copied directory]`*
   3. In `$CASEDIR/$NEWCASENAME`, run `./xmlchange BUILD_COMPLETE=TRUE`

Exercise #3 – 1D Control

About the test

In this test, we run a single column ocean test for 15 days using the CVMix KPP routines. There is no surface heat flux and the windstress is specified by $(\tau_x, \tau_y) = (0.1, 0)$ N/m^2

1. Create a new case using the following settings
   1. Component Set: *C*
   2. Resolution: *1D_1D*
2. The default simulation is 5 days long, change this to 16 days:

   `./xmlchange STOP_N=16`

   NOTE: this only runs POP for 15 days
3. The default option uses POP's KPP routines, edit `user_nl_pop2` to use CVMix's routines instead
4. Build and run the model

Exercise #4 – Change τ_x and Surface Heat Flux

About the test

In this test, we remove the wind stress but apply strong cooling at the surface (heat flux $= -100$ W/m^2).

1. Create a clone of your case from exercise #3
2. Make the necessary change to `user_nl_pop2` to set zonal surface wind stress to 0 N/m^2 and surface heat flux to -100 W/m^2
   Hint: look for `global_taux` *and* `global_shf_coeff` *in* `CaseDocs/pop2_in`
3. Build and run the model for 16 days
4. How are the results different from those of exercise #3?

Exercise #5 – Change τ_x and Surface Heat Flux (again)

About the test

In this test, we keep the 0.1 N/m^2 wind stress and also apply a strong surface heat flux (100 W/m^2).

1. Create a clone of your case from exercise #3
2. Make the necessary change to `user_nl_pop2` to set surface heat flux to 100 W/m^2
3. Build and run the model for 16 days
4. How are the results different from those of exercise #3 and #4?

Running CVMix Without an Ocean Model

Setup (argo or your local machine)

- Clone `git` repository at `git@github.com:CVMix/CVMix-src.git`
- Build by running `make` (or `make netcdf`) from `src/`
- First time you build, you will be prompted for compiler name and netcdf location. On `argo`:
  - First run `module load intel netcdf/4.3.1/intel/2013`
  - Compiler is `ifort` and location of `nc-config` is `/opt/netcdf/4.3.1/intel/2013/bin`
- Successful build ⇒ run pre-set tests in `reg_tests`
- No MPI needed means no waiting in the queue

Caveat

CVMix does not solve equations of state or do any time stepping – it just tests that "correct" diffusivities are produced for specific parameter specifications

Homework – Bring CVMix to your Model

If you are interested

- POP source code is in `$CESMROOT/models/ocn/pop2/source/`
- Main mixing driver is `vertical_mix.F90`; KPP is in `vmix_kpp.F90`
- Look for "`call cvmix_`" in the KPP code
  1. `call cvmix_init_kpp` initializes KPP module
  2. `call cvmix_kpp_compute_OBL_depth` computes depth of ocean boundary layer
  3. `call cvmix_coeffs_kpp` computes diffusivity coefficients
- To compute boundary layer depth, need bulk Richardson number (`cvmix_kpp_compute_bulk_Richardson` function)
- Need to massage model data into CVMix data type
  - See `cvmix_data_type` structure, defined in `cvmix/cvmix_kinds_and_types.F90`
  - POP uses `cvmix_put`, but data type can also use pointers instead of memory copy
- Probably easier to start with non-KPP module...

References

References

1. Bryan, K. and L. J. Lewis, 1979: A water mass model of the world ocean. *Journal of Geophysical Research*, **84**, 2503-2517.
2. Danabasoglu, G., W. G. Large, J. J. Tribbia, P. R. Gent, B. P. Briegleb, and J. C. McWilliams, 2006: Diurnal coupling in the tropical oceans of CCSM3. *Journal of Climate*, **19**, 2347-2365.
3. Ekman, V. W., 1905: On the influence of the Earth's rotation on ocean-currents. *Arkiv för Matematik, Astronomi och Fysik*, **2**, 1-53.
4. Henyey, F., J. Wright, and S. M. Flatte, 1986: Energy and action flow through the internal wave field: an eikonal approach. *Journal of Geophysical Research*, **91**, 8487-8496.
5. Jackson, L., R. Hallberg, and S. Legg, 2008: A parameterization of shear-driven turbulence for ocean climate models. *Journal of Physical Oceanography*, **38**, 1033-1053.
6. Large, W., J. McWilliams, and S. Doney, 1994: Oceanic vertical mixing: a review and a model with a nonlocal boundary layer parameterization. *Reviews of Geophysics*, **32**, 363-403.
7. Melet, A., R. Hallberg, S. Legg, and K. Polzin, 2013: Sensitivity of the ocean state to the vertical distribution of internal-tide-driven mixing. *Journal of Physical Oceanography*, **43**, 602-615.
8. Pacanowksi, R. C. and G. Philander, 1981: Parameterization of vertical mixing in munerical models of the tropical ocean. *Journal of Physical Oceanography*, **11**, 1442-1451.
9. Polzin, K. L., 2009: An abyssal recipe. *Ocean Modelling*, **30**, 298-309.
10. Schmitt, R. W., 1994: Double diffusion in oceanography. *Annual Review of Fluid Mechanics*, **26**, 255-285.
11. Simmons, H. L., S. R. Jayne, L. C. St. Laurent, and A. J. Weaver, 2004: Tidally driven mixing in a numerical model of the ocean general circulation. *Ocean Modelling*, **6**, 245-263.