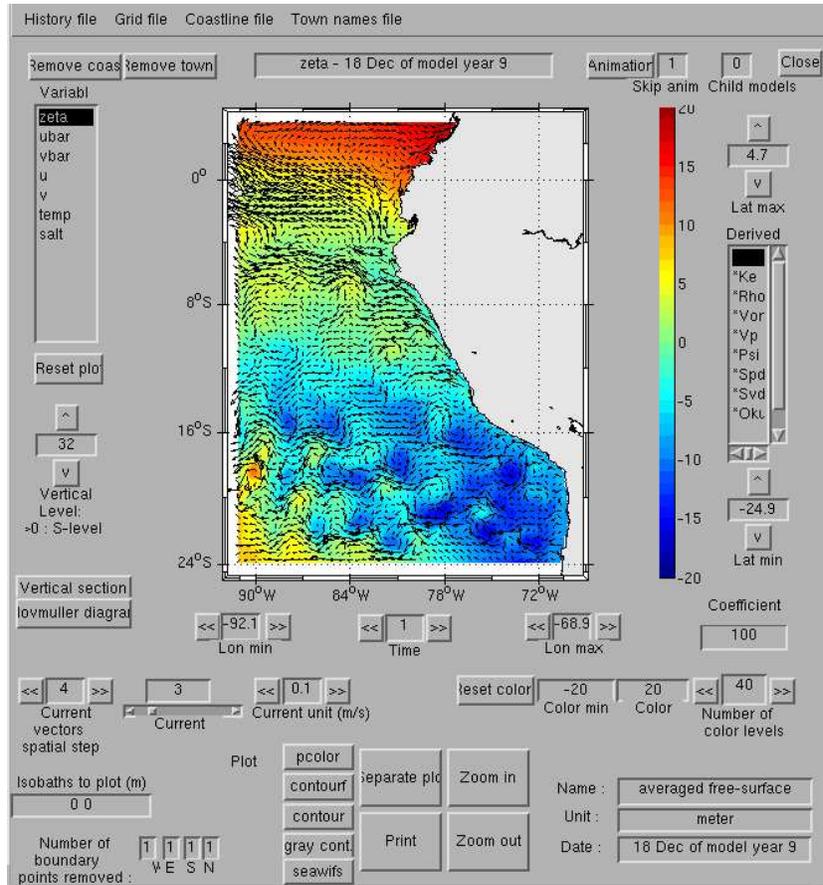


ROMSTOOLS User's Guide

Pierrick Penven

Institut de Recherche pour le Développement, 213 rue Lafayette, Paris, France



April, 2004

Contents

1	Introduction	1
1.1	Installation	1
1.1.1	System requirement	1
1.1.2	Getting the files	1
1.1.3	Extracting the files	2
1.2	Future plans	3
1.3	Warnings and bugs	4
2	The ROMS model	4
3	Tutorial : the Peruvian Upwelling System example	4
3.1	Getting started	4
3.2	Building the grid	5
3.3	Getting the wind and other surface fluxes	6
3.4	Getting the lateral boundary conditions	7
3.5	Compiling and running the model	11
3.6	Getting the results	15
4	Tides	17
4.1	Preparing the tides boundary conditions	17
4.2	Run the model	18
5	Embedding	18
5.1	Introduction	18
5.2	Prepare the child model	19
5.3	Compile and Run	20
6	Diagnostics	21
7	ROMSTOOLS users	21

1 Introduction

This guide presents a series of Matlab routines that could be useful for the pre- and post-processing of oceanic regional ROMS simulations.

The Regional Ocean Modeling System (ROMS) is a new generation ocean circulation model (Shchepetkin and McWilliams, 2004) that has been specially designed for accurate simulations of regional oceanic systems. The reader is referred to Shchepetkin and McWilliams (2003a) and to Shchepetkin and McWilliams (2004) for a complete description of the model. ROMS has been applied for the regional simulation of a variety of different regions of the world oceans (e.g. Blanke et al., 2002; Di Lorenzo et al., 2003; Haidvogel et al., 2000; MacCready et al., 2002; Marchesiello et al., 2003; Penven et al., 2001).

To perform a regional simulation using ROMS, the modeler needs to provide a whole set of data in a specific format: horizontal grid, topography, surface forcing, lateral boundary conditions. He also needs to analyze the model outputs. The tools that are described here have been designed to perform these tasks. The goal is to be able to build a standard regional model configuration in a minimum time.

In the first chapter, the system requirements and the installation process are exposed. A short note on ROMS model is presented in chapter 2. A tutorial on the use of ROMSTOOLS is shown in the third chapter. Tidal and nesting tools are presented in chapter 4 and 5.

1.1 Installation

1.1.1 System requirement

This toolbox has been designed for Matlab. It needs at least 2 Gbytes of disk space. It has been tested on Linux workstations, but it can be used on any platform if a Matlab netcdf mex file is provided. A Matlab netcdf mex file can be found at the location:

<http://woodshole.er.usgs.gov/staffpages/cdenham/public.html/MexCDF/nc4ml5.html#DOWNLOADING>
. All the necessary Matlab toolboxes (netcdf and m_map) are included in the ROMSTOOLS package. The topography (Smith and Sandwell, 1997), hydrography (Conkright et al., 2002), and surface fluxes (Da Silva et al., 1994) global datasets are also included.

1.1.2 Getting the files

A compressed tar file (roms_tools.tar.gz) containing the Matlab programs and all the necessary datasets (and also a version of ROMS) is located at :

http://fraise.univ-brest.fr/~penven/Roms_tools/

Be careful, the file is large (~ 530 Megs). If you are unable to obtain the file in this fashion, feel free to contact me at :

Pierrick Penven
Institut de Recherche pour le Développement
Laboratoire de Physique des Océans
Boite Postale 809
6, Avenue Le Gorgeu
29285 Brest CEDEX
FRANCE

email: Pierrick.Penven@ird.fr

It is also possible to get ROMSTOOLS in three separate tar files:

- data.tar.gz for the datasets
- tools.tar.gz for the Matlab programs

- Roms_agrif.tar.gz for the ROMS source code.

This way, it is not necessary to download again the whole dataset when a new ROMS release is available.

1.1.3 Extracting the files

Uncompress and untar the file (gunzip and tar -xvf). You should get the following directory tree :

```
Roms_tools
|- COADS05
|  |- Matlab
|- Compile
|- Diagnostic_tools
|- Documentation
|  |- Matlab
|- m_map
|  |- private
|- Nesting_tools
|- netcdf_g77
|- netcdf_ifc
|- netcdf_matlab
|  |- listpick
|  |- ncatt
|  |- ncbrowser
|  |- ncdim
|  |- ncfiles
|  |- ncitem
|  |- ncrec
|  |- nctype
|  |- ncutility
|  |- nvar
|  |- netcdf
|- Preprocessing_tools
|- Roms_Agrif
|  |- AGRIFZOOM
|  |  |- AGRIF_FILES
|  |  |- AGRIF_INC
|  |  |- AGRIF_OBJS
|  |  |- AGRIF_YOURFILES
|  |  |- CVS
|  |  |- LIB.clean
|  |- CVS
|- Run
|- Tides
|- Topo
|  |- Matlab
|- Visualization_tools
|- WOA2001
```

Meaning of the different directories :

- COADS05 : Location of the COADS surface fluxes data (monthly climatology at 0.5° resolution).
- Compile : Empty scratch directory for ROMS compilation.

- Diagnostic_tools : A few Matlab scripts for animations and statistical analysis.
- Documentation
- m_map : The Matlab mapping toolbox (<http://www2.ocgy.ubc.ca/~rich/map.html>).
- Nesting_tools : preprocessing tools used to prepare nested grids.
- netcdf_g77 : The netcdf Fortran library for Linux compiled with g77 (<http://www.unidata.ucar.edu/packages/netcdf/index.html>).
- netcdf_ifc : The netcdf Fortran library for Linux compiled with ifc. The Intel Fortran 95 compiler (ifc) is available at <http://www.intel.com/software/products/compilers/flin/noncom.htm>.
- netcdf_matlab : The Matlab netcdf toolbox (<http://woodshole.er.usgs.gov/staffpages/cdenham/public.html/MexCDF/nc4ml5.html>). This is where the mexfile (mexcdf60.mexglx) is located (this needs to be replaced if you are not using a Linux PC).
- Preprocessing_tools : preprocessing Matlab scripts.
- Roms_Agrif : ROMS Fortran sources.
- Run : Working directory. This is where the ROMS input files are generated and where the model is running.
- Tides : Location of the tidal dataset and the Matlab routines to prepare ROMS tidal simulations. The tidal dataset is derived from the Oregon State University global model of ocean tides TPXO.6 (Egbert and Erofeeva, 2002): <http://www.oce.orst.edu/po/research/tide/global.html>
- Topo : Location of the global topography dataset at 2' resolution (Smith and Sandwell, 1997). Original data can be found at: http://topex.ucsd.edu/cgi-bin/get_data.cgi
- Visualization_tools : Matlab scripts for the ROMS visualization graphic user interface.
- WOA2001 : World Ocean Atlas 2001 global dataset (monthly climatology at 1° resolution).

If you need to create and play .fli animations, you should install ppm2fli and xanim on your system. If you have a Linux PC, you can follow these steps:

1. log in as root
2. go to the Roms_tools directory
3. type : rpm -Uvh ppm2fli-2.1-1.i386.rpm
4. type : rpm -Uvh xanim-2.80.1-12.i386.rpm
5. log out

If you are not using a Linux PC, you should ask your system administrator to install these programs.

1.2 Future plans

- A graphic user interface should be useful for the preprocessing tools.

1.3 Warnings and bugs

- Since Geostrophy is used to derived horizontal currents from hydrology data, this method can not be applied close to the equator. To perform equatorial simulations (in a 5°S-5°N band), the user should find another way to provide horizontal currents at model boundaries.
- The interpolation procedure does not handle grids that cross the 180°E longitude meridian. Grids should stay between longitudes -180°E and 180°W, and avoid the middle of the Pacific Ocean.
- On extended grids, the objective analysis used for data extrapolation can be relatively costly in memory and CPU time. The "nearest" Matlab function that is less costly is used instead. If the computer starts to swap, you should think of reducing the dimension of the model's domain.

2 The ROMS model

ROMS solves the primitive equations in an Earth-centered rotating environment, based on the Boussinesq approximation and hydrostatic vertical momentum balance. ROMS is discretized in coastline- and terrain-following curvilinear coordinates. ROMS is a split-explicit, free-surface ocean model, where short time steps are used to advance the surface elevation and barotropic momentum, with a much larger time step used for temperature, salinity, and baroclinic momentum. ROMS employs a special 2-way time-averaging procedure for the barotropic mode, which satisfies the 3D continuity equation (Shchepetkin and McWilliams, 2004). The specially designed predictor-corrector time step algorithm used in ROMS allows a substantial increase in the permissible time-step size.

ROMS has been designed to be optimized on shared memory parallel computer architectures such as the SGI/CRAY Origin 2000. Parallelization is done by two dimensional sub-domains partitioning. Multiple sub-domains can be assigned to each processor in order to optimize the use of processor cache memory. This allow super-linear scaling when performance growth even faster than the number of CPUs.

The third-order, upstream-biased advection scheme implemented in ROMS allows the generation of steep gradients, enhancing the effective resolution of the solution for a given grid size (Shchepetkin and McWilliams, 1998). Explicit lateral viscosity is null everywhere in the model domain except in sponge layers near the open boundaries where it increases smoothly close to the lateral open boundaries.

A non-local, K-profile planetary (KPP) boundary layer scheme (Large, 1994) parameterizes the unresolved physical vertical subgrid-scale processes. If a lateral boundary faces the open ocean, an active, implicit, upstream biased, radiation condition connects the model solution to the surroundings (Marchesiello et al., 2001).

3 Tutorial : the Peruvian Upwelling System example

This section presents the essential steps for preparing and running a regional ROMS simulation. This is done following the example of a model of the Peruvian Upwelling System at low resolution.

3.1 Getting started

Once the installation has been successful, start a Matlab session in the directory: `~/Roms_tools/Run`. Launch the `start.m` script to set the Matlab path for this session:

```
>> start
Add the paths of the different toolboxes...
>>
```

You are now ready to start a new configuration. It is now important to respect the order of the following steps.

3.2 Building the grid

Open the file : `~/Roms_tools/Preprocessing_tools/make_grid.m` in you favorite editor. The part of the file that you should change is :

```
%
% Title
%
title='Peru Test Model';
%
% Grid file name
%
grdname='roms_grd.nc';
%
% Slope parameter (r=grad(h)/h) maximum value for topography smoothing
%
rtarget=0.2;
%
% Grid dimensions:
% lonmin : Minimum longitude [degree east]
% lonmax : Maximum longitude [degree east]
% latmin : Minimum latitude [degree north]
% latmax : Maximum latitude [degree north]
%
lonmin=-85;
lonmax=-75;
latmin=-15;
latmax=-7;
%
% Grid resolution [degree]
%
dl=1/3;
%
% Minimum depth [m]
%
hmin=10;
%
% Topography netcdf file name (ETOPO 2)
%
topofile='../Topo/etopo2.nc';
%
% GSHSS user defined coastline (see m_map)
%
coastfileplot="";
```

The variables that you can change :

- `title='Peru Test Model'` : You can give any name you want for your configuration
- `grdname='roms_grd.nc'` : Name of the ROMS netcdf grid file you are building. In general, we follow the style : `XXX_grd.nc`.

- `rtarget=0.2` : This variable control the maximum value of the r -parameter that measures the slope of the sigma layers (Beckmann and Haidvogel, 1993):

$$r = \frac{\nabla h}{2h} = \frac{h_{+1/2} - h_{-1/2}}{h_{+1/2} + h_{-1/2}}$$

To prevent horizontal pressure gradients errors, well known in terrain-following coordinate models (Haney, 1991), realistic topography requires some smoothing. Empirical results have shown that reliable results are obtained if r does not exceed 0.2.

- `lonmin=-85` : Western limit of the grid in longitude [-180°E, 180°E]. The grid is rectangular in latitude/longitude.
- `lonmax=-75` : Eastern limit [-180°E, 180°E]. Should be superior to `lonmin`.
- `latmin=-15` : Southern limit of the grid in latitude [-90°N, 90°N].
- `latmax=-7` : Northern limit [-90°N, 90°N]. Should be superior to `latmin`.
- `dl=1/3` : Grid longitude spacing in degrees. The latitude spacing is deduced to obtain an isotropic grid ($d\phi = dl \cos(\phi)$).
- `hmin=10` : Minimum depth in meters. The model depth is cut a this level to prevent, for example, the occurrence of model grid cells without water. This does not influence the masking routines.
- `topofile='../Topo/etopo2.nc'` : Default topography file. We are using here `etopo2` (Smith and Sandwell, 1997).
- `coastfileplot=""` : If you have a `m_map` binary coastal file to make nice plots, you can add it here. These files can be generated with `make_coast.m`. If there is no file, the program will use a default global coarse coastline file. This has only an effects on graphics.

Save `make_grid.m` and run it in the Matlab session :

```
>> make_grid
```

Be careful to note the values given for `LLm` and `MMm` during the process, they will be necessary for the ROMS file `param.h` (in this test case, `LLm0 = 29` and `MMm0 = 24`). Figure (1) presents the bottom topography obtained with `make_grid.m` for the Peru Upwelling example. Note that at this low resolution ($1/3^\circ$), the topography has been strongly smoothed. Once you are happy with the horizontal grid, you can add the surface forcing.

3.3 Getting the wind and other surface fluxes

The next step is to create the file containing the different surface fluxes. Edit the file `~/Roms_tools/Preprocessing_tools/make_forcing.m`. Since we are going to use the default climatology surface forcing (Da Silva et al., 1994), the part that you need to change is only:

```
%
% Title - Grid file name - Forcing file name
%
title='Forcing (COADS)';
grdname='roms_grd.nc';
frcname='roms_frc.nc';
%
```

These variables are :

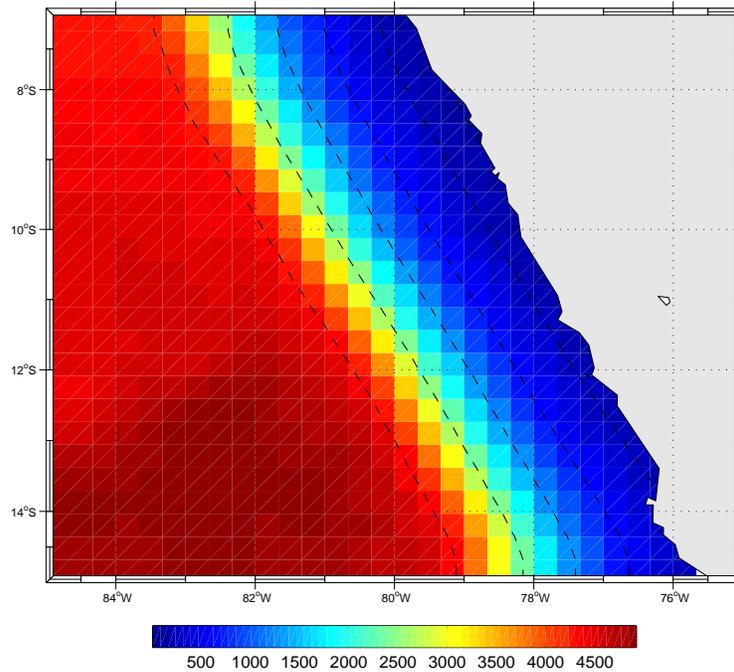


Figure 1: Result of make_grid.m for the Peru example

- title='Forcing (COADS)' : Any title is good.
- grdname='roms_grd.nc' : Name of the ROMS netcdf grid file that has been previously created.
- frcname='roms_frc.nc' : Name of the ROMS netcdf surface forcing file you are building. In most cases, we follow the style : XXX_frc.nc.

Save make_forcing.m and run it in the Matlab session :

```
>> make_forcing
```

This program can take a relatively long time to process all the variables. Figure (2) presents the sea surface temperature and the wind vectors that has been interpolated from the climatology (Da Silva et al., 1994) at 4 different times of the year. The sea surface temperature is used for the restoring term in the heat flux calculation.

3.4 Getting the lateral boundary conditions

The last preprocessing step consist in generating the files containing the necessary informations for the lateral open boundaries and the initial conditions of the model. This is done using the file `~/Roms_tools/Preprocessing_tools/make_clim.m`. This script process two files : the climatology file that gives the lateral boundary conditions and the initial condition file. The part that can be changed by the user is as follow:

```
%
% Title
%
title='Climatology';
%
% Switches for selecting what to process (1=ON)
```

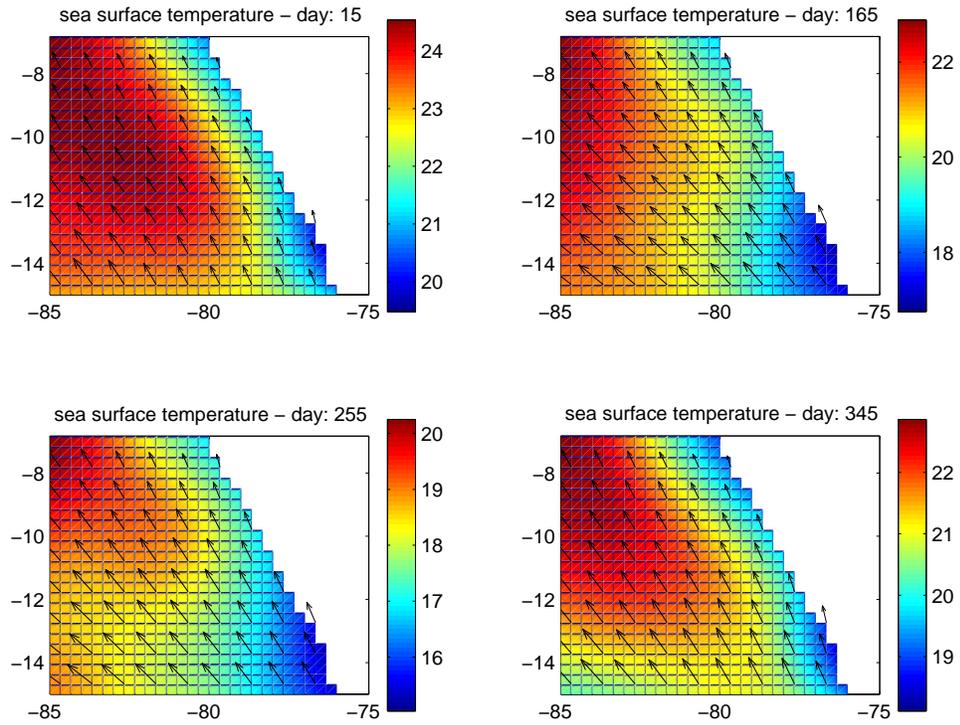


Figure 2: Result of make_forcing.m for the Peru example

```

%
makeclim=1; %1: process boundary data
makeoa=1; %1: process oa data
makeini=1; %1: process initial data
%
% Grid file name - Climatology file name
% Initial file name - OA file name
%
grdname='roms_grd.nc';
frcname='roms_frc.nc';
clmname='roms_clm.nc';
ininame='roms_ini.nc';
oaname='roms_oa.nc';
%
% Vertical grid parameters
% !!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!
% !!! WARNING WARNING WARNING WARNING WARNING !!!
% !!! THESE MUST IDENTICAL IN THE .IN FILE !!!
% !!! WARNING WARNING WARNING WARNING WARNING !!!
% !!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!
%
theta_s=7.;
theta_b=0.;
hc=5.;
N=20; % number of vertical levels (rho)
%
% Open boundaries switches

```

```

% !!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!
% !!! WARNING WARNING WARNING WARNING WARNING !!!
% !!! MUST BE CONFORM TO THE CPP SWITCHES !!!
% !!! WARNING WARNING WARNING WARNING WARNING !!!
% !!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!
%
%
obc=[1 0 1 1]; % open boundaries (1=open , [S E N W])
%
% Level of reference for geostrophy calculation
%
zref=-500;
%
% Day of initialization
%
tini=15;
%
% Set times and cycles: monthly climatology for all data
%
time=[15:30:345]; % time
cycle=360; % cycle
%
% Data climatologies file names:
%
% temp_month_data : monthly temperature climatology
% temp_ann_data : annual temperature climatology
% salt_month_data : monthly salinity climatology
% salt_ann_data : annual salinity climatology
%
temp_month_data='../WOA2001/temp_month.cdf';
temp_ann_data='../WOA2001/temp_ann.cdf';
insitu2pot=1; % transform in-situ temperature to potential temperature
salt_month_data='../WOA2001/salt_month.cdf';
salt_ann_data='../WOA2001/salt_ann.cdf';
%
%

```

Variables description :

- title='Climatology' : Any title is good.
- makeclim=1 : Switch to define if the climatology (boundary conditions) file is generated. should be 1.
- makeoa=1 : Switch to define if the OA (objective analysis) file is generated. This should be 1. The OA files are intermediate files where hydrographic data are stored on a ROMS horizontal grid but on a z vertical grid. The transformation to S-coordinate is done after. These files are not used by ROMS.
- makeini=1 : Switch to define if the initial file is generated. It should be 1.
- grdname='roms_grd.nc' : Grid file name previously generated.
- frcname='roms_frc.nc' : Surface forcing file name previously generated. We need to provide the surface forcing to compute the Ekman transport for the velocities of the lateral boundary conditions.
- clmname='roms_clm.nc' : Climatology file name.

- `iname='roms_ini.nc'` : Initial file name.
- `oaname='roms_oa.nc'` : OA intermediate file name.
- `theta_s=7.` : Vertical S-coordinate surface stretching parameter. When building the climatology and initial ROMS files, we have to define the vertical grid. Be cautious, the vertical grid parameters should be identical in this file and in the ROMS input file (i.e. `~/Roms_tools/Run/roms.in`). This is a serious cause of bug. The effects of `theta_s`, `theta_b`, `hc`, and `N` can be tested using the script :
`~/Roms_tools/Preprocessing_tools/test_vgrid.m`.
- `theta_b=0.` : Vertical S-coordinate bottom stretching parameter.
- `hc=5.` : Vertical S-coordinate H_c parameter. It gives approximately the transition depth between the horizontal surface levels and the bottom terrain following levels. It should be inferior to `hmin`.
- `N=20` : Number of vertical level. Be careful, this have to be also defined in the file :
`~/Roms_tools/Run/param.h` before compiling the model.
- `obc=[1 0 1 1]` : Switches to open (1) or close (0=wall) the lateral boundaries [South East North West]. This is used when the constraint of mass enforcement is applied. Be aware, this should be compatible with the open boundary CPP-switches in the file `~/Roms_tools/Run/cppdefs.h`.
- `zref=-500` : Depth in meters of no motion when computing the geostrophic velocities.
- `tini=15` : Day of initialization (15 = January 15).
- `time=[15:30:345]` : Time in days in the datasets (Matlab vector).
- `cycle=360` : This variable define that the model will be run forced by climatological data on a cyclic year of 360 days. If not, it should be 0.
- `temp_month_data='../WOA2001/temp_month.cdf'` : Name of the monthly climatology temperature data file.
- `temp_ann_data='../WOA2001/temp_ann.cdf'` : Name of the annual climatology temperature data file. Since the monthly climatology contains only the 1500 first meters, the annual climatology is used for the deeper levels.
- `insitu2pot=1` : Switch defined if it is in-situ temperature that is provided. In this case, in-situ temperature is converted into potential temperature.
- `salt_month_data='../WOA2001/salt_month.cdf'` : Name of the monthly climatology salinity data file.
- `salt_ann_data='../WOA2001/salt_ann.cdf'` : Name of the annual climatology salinity data file.

Save `make_clim.m` and run it in the Matlab session :

```
>> make_clim
```

This program can also take quite a long time to run.

Figure (3) presents 4 different sections of temperature in the initial condition file for the Peru example. The section are in the X-direction (East-West), the first section is for the South of the domain and the last one is for the North of the domain.

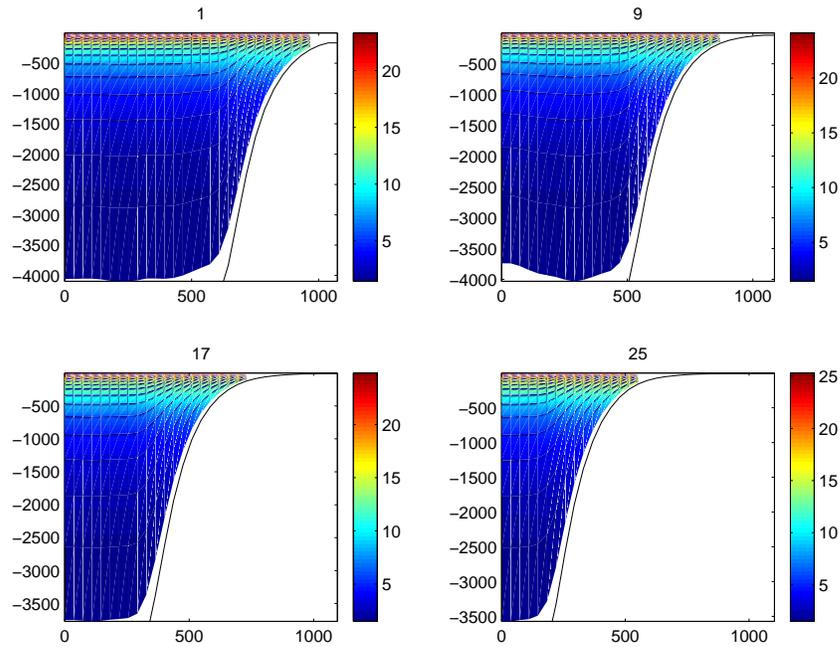


Figure 3: Result of make_clim.m for the Peru example

3.5 Compiling and running the model

Once all the netcdf data files are ready (i.e. XXX_grd.nc, XXX_frc.nc, XXX_ini.nc, and XXX_clm.nc), we can prepare ROMS for compilation. All is done in the `~/Roms_tools/Run/` directory. First edit the file `~/Roms_tools/Run/param.h`. The line that needs to be changed is:

```
# elif defined PERU
  parameter (LLm0=29, MMm0=24, N=20) ! < -- Peru Test Case
# else
```

These are the values of the model grid size: LLm0 points in the X direction, MM0m points in the Y direction and N vertical levels. LLm0 and MMm0 are given by running make_grid.m, and N is defined in make_clim.m.

The second file to change is `~/Roms_tools/Run/cppdefs.h`. This file defines the CPP keys that are used by the the C-preprocessor when compiling ROMS. Definitions of the CCP keys in cppdefs.h:

- BASIN : Must be defined for running the Basin Example.
- CANYON_A : Must be defined for running the Canyon_A Example.
- CANYON_B : Must be defined for running the Canyon_B Example.
- GRAV_ADJ : Must be defined for running the Gravitational Adjustment Example.
- INNERSHELF : Must be defined for running the Inner Shelf Example.
- OVERFLOW : Must be defined for running the Graviational/Overflow Example.
- SEAMOUNT : Must be defined for running the Seamount Example.
- SHELFROUNT : Must be defined for running the Shelf Front Example.
- SOLITON : Must be defined for running the Equatorial Rossby Wave Example.

- UPWELLING : Must be defined for running the Upwelling Example.
- REGIONAL_MODEL : Must be defined if running realistic regional simulations.
- PERU : Configuration Name, this is used in param.h.
- OPENMP : Activate Open-MP parallelization protocol.
- MPI : Activate MPI parallelization protocol.
- AGRIF : Activate the nesting capabilities
- SOLVE3D : Define if solving 3D primitive equations
- UV_COR : Activate Coriolis terms.
- UV_ADV : Activate advection terms.
- SSH_TIDES : Define for processing sea surface elevation tidal data at the model boundaries.
- UV_TIDES : Define for processing ocean current tidal data at the model boundaries.
- VAR_RHO_2D : Activate nonuniform density in barotropic mode pressure- gradient terms.
- FLAT_WEIGHTS : Use a more dissipative averaging for the baroclinic/barotropic coupling.
- CURVGRID : Activate curvilinear coordinate grid option.
- SPHERICAL : Activate longitude/latitude grid positioning.
- MASKING : Activate land masking in the domain.
- AVERAGES : Define if writing out time-averaged data.
- SALINITY : Define if using salinity.
- NONLIN_EOS : Activate the nonlinear equation of state.
- SPLIT_EOS : Activate to split the nonlinear equation of state in a adiabatic part and a compressible part.
- ZCLIMATOLOGY : Activate processing of sea surface height climatology.
- UCLIMATOLOGY : Activate processing of momentum climatology.
- ZNUDGING : Activate open boundary passive/active term + nudging layer for zeta.
- M2NUDGING : Activate open boundary passive/active term + nudging layer for u_{bar} and v_{bar}.
- SPONGE : Activate areas of enhanced viscosity/diffusion.
- QCORRECTION : Activate net heat flux correction.
- SFLX_CORR : Activate freshwater flux correction.
- DIURNAL_SRFLUX : Activate diurnal modulation of the short wave radiation flux
- TCLIMATOLOGY : Activate processing of tracer climatology.
- TNUDGING : Activate open boundary passive/active term + nudging layer for tracers.
- M3NUDGING : Activate open boundary passive/active term + nudging layer for u and v.
- ROBUST_DIAG : Activate strong tracer nudging in the interior for diagnostic simulations.

- ANA_SSFLUX : Define if using analytical surface salinity flux.
- ANA_SST : Define if using analytical SST and dQdSST.
- ANA_SSS : Define if using analytical sea surface salinity.
- ANA_SRFLUX : Define if using analytical surface shortwave radiation flux.
- ANA_STFLUX : Define if using analytical surface temperature flux.
- ANA_BSFLUX : Define if using analytical bottom salinity flux.
- ANA_BTFLUX : Define if using analytical bottom temperature flux.
- ANA_TCLIMA : Activate analytical tracer climatology.
- ANA_SMFLUX : Define if using analytical surface momentum stress.
- ANA_UCLIMA : Activate analytical momentum climatology.
- VIS_GRID : Activate viscosity coefficient scaled by grid size.
- UV_VIS2 : Activate Laplacian horizontal mixing.
- TS_DIF2 : Activate Laplacian horizontal mixing.
- DIF_GRID : Activate diffusion coefficient scaled by grid size.
- MIX_GP_TS : Activate mixing on geopotential (constant Z) surfaces.
- MIX_GP_UV : Activate mixing on geopotential (constant Z) surfaces.
- CLIMAT_TS_MIXH : Activate mixing of T-Tclim instead of tracers.
- BODYFORCE : Define if applying stresses as bodyforces.
- BVF_MIXING : Activate Brunt-Vaisala frequency mixing scheme.
- PP_MIXING : Activate Pacanowsky/Philander mixing scheme.
- LMD_MIXING : Activate Large/McWilliams/Doney mixing (LMD-KPP closure).
- LMD_SKPP : Activate surface boundary layer KPP mixing (LMD-KPP closure).
- LMD_BKPP : Activate bottom boundary layer KPP mixing (LMD-KPP closure).
- LMD_RIMIX : Activate shear instability interior mixing (LMD-KPP closure).
- LMD_CONVEC : Activate convection interior mixing (LMD-KPP closure).
- LMD_DDMIX : Activate double diffusion interior mixing (LMD-KPP closure).
- LMD_BOTEK : Activate a bottom Ekman layer parametrisation.
- LMD_NONLOCAL : Activate nonlocal transport (LMD-KPP closure).
- OBC_EAST : Open eastern boundary (should be consistent with make_clim.m).
- OBC_WEST : Open western boundary (should be consistent with make_clim.m).
- OBC_NORTH : Open northern boundary (should be consistent with make_clim.m).
- OBC_SOUTH : Open southern boundary (should be consistent with make_clim.m).
- OBC_VOLCONS : Activate mass conservation enforcement at open boundaries.

- OBC_M2ORLANSKI : Activate 2D radiation open boundary conditions for ubar and vbar.
- OBC_M2FLATHER : Activate Flather open boundary conditions for ubar and vbar.
- OBC_TORLANSKI : Activate 2D radiation open boundary conditions for tracers.
- OBC_M3ORLANSKI : Activate 2D radiation open boundary conditions for u and v.
- OBC_TSPECIFIED : Activate specified open boundary conditions for tracers.
- OBC_M2SPECIFIED : Activate specified open boundary conditions for ubar and vbar.
- OBC_M3SPECIFIED : Activate specified open boundary conditions for u and v.
- AGRIF_STORE_BAROT_CHILD : Store ubar and vbar during the parent step for the child boundary conditions (Nesting).
- AGRIF_FLUX_BC : Apply parent/child barotropic boundary conditions has fluxes (Nesting).
- AGRIF_POLY_DUAVG : Apply a third order polynomial temporal interpolation for DU_avg1 and DU_avg2 (Nesting).
- AGRIF_CORRECT_LOCAL_FLUXES : Apply a local barotropic flux conservation enforcement at the child boundaries (Nesting).
- AGRIF_RAD2D : Radiate child barotropic boundary conditions (Nesting).
- AGRIF_RAD3D : Radiate child 3d momentum boundary conditions (Nesting).
- AGRIF_RADT : Radiate child tracer boundary conditions (Nesting).
- AGRIF_2WAY : 2-way nesting (Nesting).
- BIOLOGY : Activate the biogeochemical module.
- FLOATS : Activate floats.
- FLOATS_GLOBAL_ATTRIBUTES
- RANDOM_WALK
- PASSIVE_TRACER : Add a passive tracer
- SEDIMENT : Activate the sediment module.
- ANA_SEDIMENT
- BED_ARMOR
- ANA_SPFLUX
- ANA_BPFLUX
- LINEAR_CONTINUATION
- NEUMANN
- ROUSE
- BBL : Activate the bottom boundary layer module.
- ANA_WWAVE
- ANA_BSEDIM

- Z0_BL
- Z0_RIP
- Z0_BIO

In the case of standard regional modeling, only open boundary conditions switches should be changed in `cppdefs.h` (to be conformed to `make_clim.m`). Once done, ROMS can be compiled by running the UNIX script `~/Roms_tools/Run/jobcomp`. The script `jobcomp` is able to recognize your system. It has been tested with Linux, IBM and Compaq computers. On Linux PCs, the default compiler is the GNU `g77`, but it is possible to uncomment specific lines in `jobcomp` to use the Intel Fortran Compiler (IFC). The latter is mandatory when using AGRIF and/or OPEN_MP. When changing the compiler you should provide a corresponding netcdf library. Once compiled you should obtain a new executable (`roms`) in the `~/Roms_tools/Run` directory.

Now, edit the input parameter file: `~/Roms_tools/Run/roms.in`. The vertical grid parameters (`THETA_S`, `THETA_B`, `HC`) should be identical to the ones in `make_clim.m`. Otherwise, the other default values should not be changed. The meaning of all the input variable is given at the start of each ROMS simulation. To run the model, type in directory `~/Roms_tools/Run/` : `roms roms.in`.

3.6 Getting the results

Once the model has run, or during the simulation, it is possible to easily visualize the model outputs using a Matlab graphic user interface : `roms_gui`. Launch `roms_gui` in the Matlab session (in `~/Roms_tools/Run/` directory):

```
>> roms_gui
```

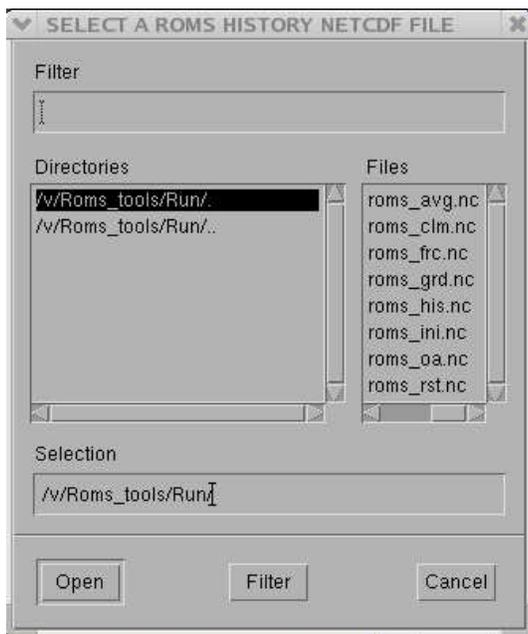


Figure 4: Entrance window of `roms_gui`

A window pops up, asking for a ROMS history netcdf file (Figure 4). You should select `roms_his.nc` (history file) or `roms_avg.nc` (average file) and click "open".

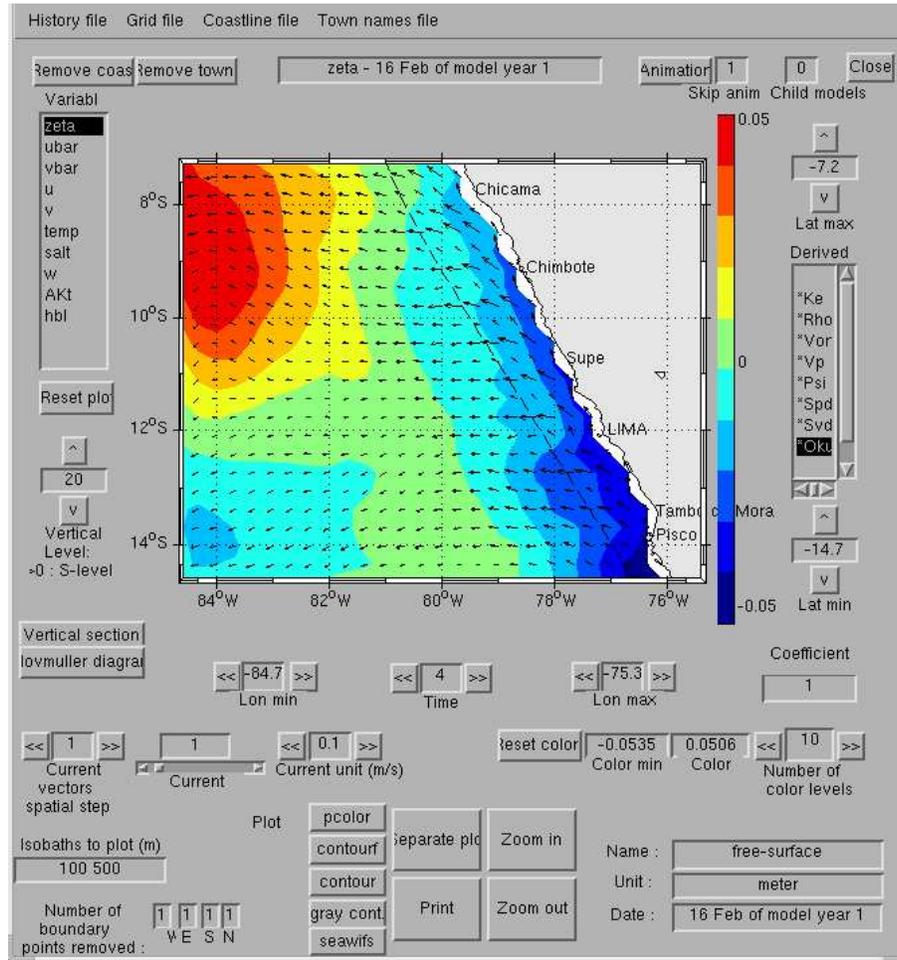


Figure 5: roms_gui

The main window appears, variables can be selected to obtain an image such as (Figure 5). The left box gives the available variable names. the right box presents the derived variables. These are variables diagnosed from ROMS outputs :

- Ke : Horizontal slice of kinetic energy: $0.5(u^2 + v^2)$.
- Rho : Horizontal slice of density using the non-linear equation of state for seawater of Jackett and McDougall (1995).
- Vor : Horizontal slice of vorticity: $\frac{\partial v}{\partial x} - \frac{\partial u}{\partial y}$.
- Vp : Horizontal slice of the vertical component of Ertel's potential vorticity: $\frac{\partial \lambda}{\partial z} \left[f + \left(\frac{\partial v}{\partial x} - \frac{\partial u}{\partial y} \right) \right]$. In our case, $\lambda = \rho$.
- Psi : Horizontal slice of stream function: $\nabla^2 \psi = \frac{\partial v}{\partial x} - \frac{\partial u}{\partial y}$. This routine might be costly since it inverts the Laplacian of the vorticity (using a successive over relaxation solver).
- Spd : Horizontal slice of the ocean currents velocity : $\sqrt{u^2 + v^2}$.
- Svd : Horizontal slice of the transport stream function : $\nabla^2 S_{vd} = \frac{\partial \bar{v}}{\partial x} - \frac{\partial \bar{u}}{\partial y}$.

- Oku : Horizontal slice of the Okubo-Weiss parameter : $\Lambda^2 = \left(\frac{\partial u}{\partial x} - \frac{\partial v}{\partial y}\right)^2 + \left(\frac{\partial v}{\partial x} + \frac{\partial u}{\partial y}\right)^2 - \left(\frac{\partial v}{\partial x} - \frac{\partial u}{\partial y}\right)^2$.

It is possible to add arrows for the horizontal currents by increasing the "Current vectors spatial step". It is also possible to obtain vertical sections and Hovmüller diagrams by clicking on the corresponding targets in roms_gui.

4 Tides

Using the methods described by Flather (1976), ROMS is able to propagate the different tidal constituents from its boundaries. Simply define the cpp keys SSH_TIDES and UV_TIDES and recompile the model using jobcomp.

4.1 Preparing the tides boundary conditions

The tidal components are added to the forcing file (XXX_frc.nc) by the Matlab program make_tides.m. Edit the file : ~/Roms_tools/Preprocessing_tools/make_tides.m. The part of the file that you should change is :

```
%
% TPXO file name
%
tidename='../Tides/TPXO6.nc';
%
% ROMS file names
%
gname='roms_grd.nc';
fname='roms_frc.nc';
%
% Number of tides component to process
%
Ntides=5;
%
% Set start time of simulation
%
year=2000;
month=1;
day=15;
hr=0.;
minute=0.;
second=0.;
```

The variables that you can change :

- tidename='../Tides/TPXO6.nc'; : Location of the netcdf tidal dataset. This file is derived from the Oregon State University global model of ocean tides TPXO.6 (Egbert and Erofeeva, 2002). Data sources can be found at <http://www.oce.orst.edu/po/research/tide/global.html>.
- gname='roms_grd.nc' : Name of the ROMS netcdf grid file.
- fname='roms_frc.nc' : Name of the ROMS netcdf forcing file. This is where the tidal informations will be added.
- Ntides=5; : Number of tidal components to process. Be careful ! This value should be identical to the value of the parameter Ntides in param.h: "parameter (Ntides=5)".

- year=2000; month=1; day=15; hr=0.; minute=0.; second=0.; : This is the starting time of simulation. A procedure correct phases and amplitudes for real time runs. It employs parts of a post-processing code from Egbert and Erofeeva (2002) TPXO model.

4.2 Run the model

Save `make_tides.m` and run it in the Matlab session. The model is now ready to run with tides. To work correctly, the model should use the Flather (1976) open boundary radiation scheme (cpp key `OBC_M2FLATHER` defined).

5 Embedding

5.1 Introduction

To address the challenge of bridging the gap between near-shore and offshore dynamics, a nesting capability has been added to ROMS and tested for the California Upwelling System (Penven et al., 2004). The method chosen for embedded gridding takes advantage of the AGRIF (Adaptive Grid Refinement in Fortran) package (Blayo and Debreu, 1999; Debreu, 2000; Debreu and Blayo, 2003; Debreu and Vouland, 2003). AGRIF is a Fortran 95 package for the inclusion of adaptive mesh refinement (AMR) features within a finite difference numerical model. The package is based on the use of pointers which minimizes the changes in the original numerical model. Regardless of the possibility of adaptive refinement (i.e the automatic generation and evolution of high-resolution subdomains in response to dynamical criteria), one of the major advantages of AGRIF in static-grid embedding is the ability to manage an arbitrary number of fixed grids and an arbitrary number of embedding levels.

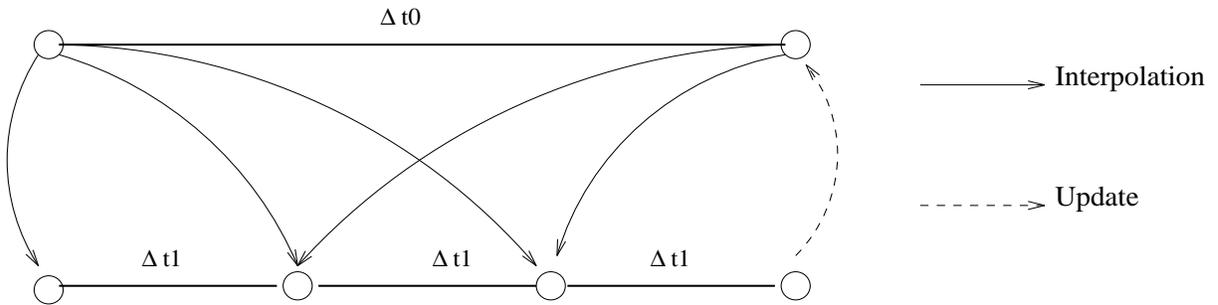


Figure 6: Temporal coupling between a parent and a child grid for a refinement factor of 3. The coupling is done at the baroclinic time step.

A recursive integration procedure manages the time evolution for the child grids during the time step of the parent grids (Fig. 6). ROMS resolves explicitly the external mode. In order to preserve the CFL criterion, for a typical coefficient of refinement (say, a factor of 3 for a 5 km resolution grid embedded in a 15 km grid), for each parent time step the child must be advanced using a time step divided by the coefficient of refinement as many time as necessary to reach the time of the parent (Fig. (6)). For simple 2-level embedding, the procedure is as follows:

1. Advance the parent grid by one parent time step.
2. Interpolate the relevant parent variables in space and time to get the boundary conditions for the child grid.
3. Advance the child grid by as much child time steps as necessary to reach the new parent model time.

4. Update point by point the parent model by averaging the more accurate values of the child model (in the case of 2-way embedding).

The recursive approach allows the specification of any number of embedding level.

5.2 Prepare the child model

To run an embedded model, the user must provide the grid, the surface forcing and the initial conditions. To name the different files AGRIF employs a specific strategy: if the parent file names are of the form: XXX.nc, the first child names will be of the form: XXX.nc.1, the second: XXX.nc.2, etc... This convention is also applied to the roms.in files.

A graphic user interface (NestGUI) facilitates the generation of the different netcdf files. Launch nestgui in the Matlab session (in `~/Roms_tools/Run/` directory):

```
>> nestgui
```

A window pops up, asking for a "PARENT GRID" netcdf file (Figure 7). In our Peru test

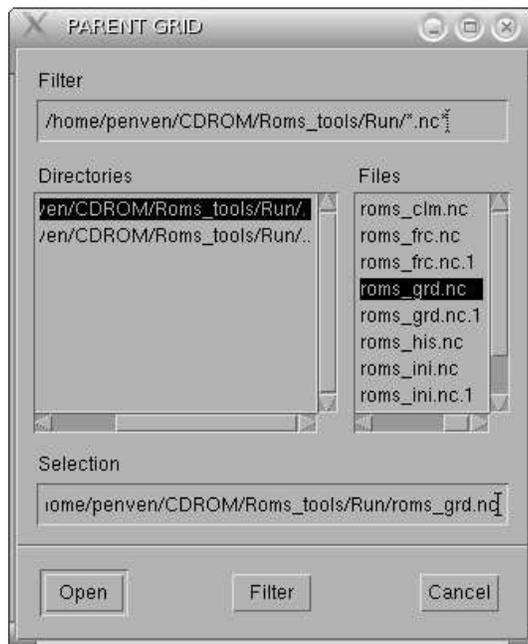


Figure 7: Entrance window of NestGUI

case, you should select `roms_grd.nc` (grid file) and click "open". The main window appears (Figure 8).

1. To define the child domain, click "Define child" and click on 2 opposite corners of the child domain in the main window. The size of the grid child (L_{child} and M_{child}) are now visible. This operation can be redone until you are happy with the position of the child domain.
2. Click "Interp child". You will be asked if you want to use a new topography. If the answer is "yes", you will have to give the location of the topography file: `~/Roms_tools/Topo/etopo2.nc`. The bathymetry will be smoothed until it fits the r factor (Beckmann and Haidvogel, 1993) that is defined in the NestGUI window. It will be connected to the parent topography following the relation: $h_{new} = \alpha \cdot h_{child} + (1 - \alpha) \cdot h_{parent}$, where α is going from 0 to 1 in n points from the boundary (also defined in the NestGUI window). If the answer is "no", the parent topography is simply interpolated on the child grid.

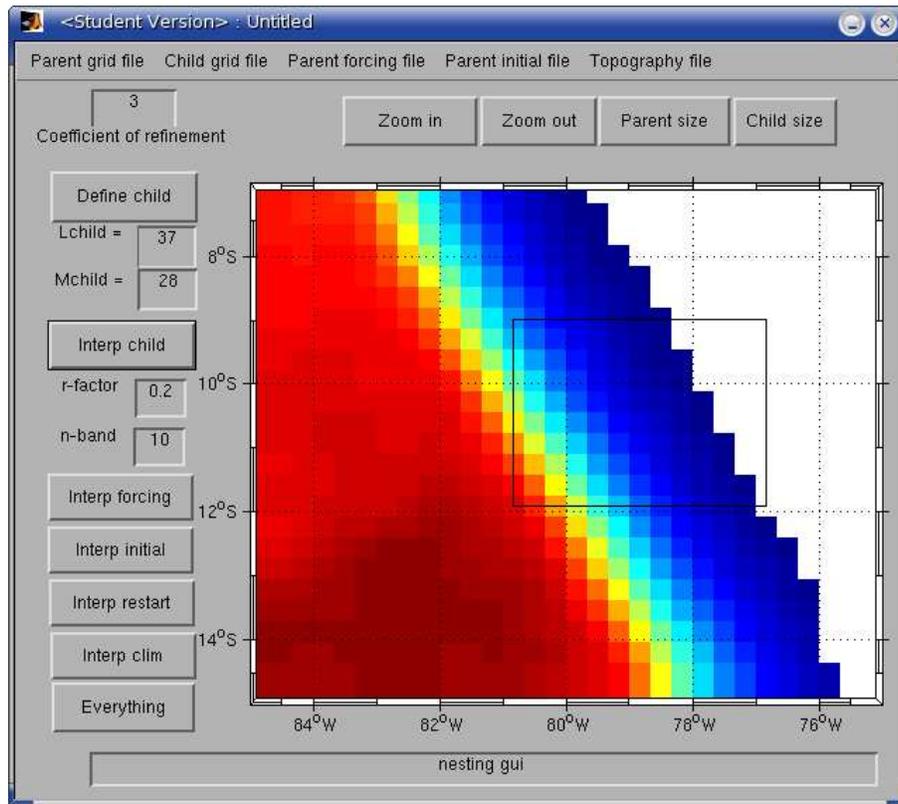


Figure 8: The NestGUI main window

3. Click "Interp forcing". The parent surface fluxes are interpolated on the child grid.
4. Click "Interp initial". The parent initial conditions are interpolated on the child grid. In case of different topographies between the parent and the child grids, the child initial conditions are vertically re-interpolated.

"Interp clim" might be useful to generate boundary conditions to test the child model alone. "Interp restart" is used to generate a child restart file. This a child domain can be "hot started" after the spin-up of the parent model.

5.3 Compile and Run

The ROMS nesting procedure needs a Fortran 95 compiler. For Linux PCs, the Intel Fortran 95 Compiler (IFC) is available for free at <http://www.intel.com/software/products/compilers/flin/noncom.htm>. To be able to compile ROMS with IFC, you should change the corresponding comments in jobcomp. Define AGRIF in `~/Roms_tools/Run/cppdefs.h`. Other cppkeys are related to AGRIF:

- AGRIF_STORE_BAROT_CHILD : Store ubar and vbar during the parent step for the child boundary conditions (Nesting).
- AGRIF_FLUX_BC : Apply parent/child barotropic boundary conditions has fluxes (Nesting).
- AGRIF_POLY_DUAVG : Apply a third order polynomial temporal interpolation for DU_avg1 and DU_avg2 (Nesting).
- AGRIF_CORRECT_LOCAL_FLUXES : Apply a local barotropic flux conservation enforcement at the child boundaries (Nesting).

- AGRIF_RAD2D : Radiate child barotropic boundary conditions (Nesting).
- AGRIF_RAD3D : Radiate child 3d momentum boundary conditions (Nesting).
- AGRIF_RADT : Radiate child tracer boundary conditions (Nesting).
- AGRIF_2WAY : 2-way nesting (Nesting).

The default definitions should give satisfaction for most of the applications. Edit the file `AGRIF_FixedGrids.in`. This file contains the child grid positions (i.e. `imin,imax,jmin,jmax`) and coefficients of refinement. A first line gives the number of children grids per parent (if `AGRIF_STORE_BAROT_CHILD` is defined, only one child grid can be defined per parent grid). A second line gives the relative position of each grid and the coefficient of refinement for each dimension. Edit the input files `roms.in.1`, `roms.in.2`, etc... to define correctly the file names and the time steps. To run the model, simply type at the prompt: `roms roms.in`. To visualize the different grid levels, change the value of the "child models" box in `roms_gui`.

6 Diagnostics

In the case of long simulations, it appeared more practical to run month by month simulations. The `csh` script : `~/Roms_tools/Run/run_roms.csh` is restarting automatically `roms` each simulated month (or year). To analyze these long simulations, a few scripts have been added in the directory: `~/Roms_tools/Diagnostic_tools`:

- `roms_anim.m` : Create an animation from the monthly history or average files.
- `get_Mmean.m` : Get the mean for each month.
- `get_Smean.m` : Get the seasonal and annual mean.
- `get_Meddy.m` : Get the variance for each month.
- `get_Meddy_ns.m` : Get the non-seasonal (i.e., the seasonal variation are filtered) variance for each month.
- `get_Seddy.m` : Get the seasonal and annual variance.
- `get_Seddy_ns.m` : Get the seasonal and annual non-seasonal variance.

The mean or variance files can be visualized by `roms_gui`.

7 ROMSTOOLS users

This toolbox and/or `roms_AGRIF` have been now used by several researchers in a variety of applications (Table 1).

User	Institute	Country	Application domain
Olivier Aumont	IRD	France	Humboldt Current
Moacyr Araújo	UFPE	Brasil	Brazilian Current
Jean-Michel Baey		France	
Arnaud Bernard	UBO	France	Iberic Peninsula
Meinte Blaas	UCLA	USA	California Current
Moncef Boukthir	IPEIT	Tunisia	Gulf of Gabes
Xavier Capet	UCLA	USA	California Current
Gildas Cambon	UBO	France	Mer d'Iroise
Laurent Debreu	INRIA	France	California Current
Emanuele Di Lorenzo	SCRIPPS	USA	California Current
Changming Dong	UCLA	USA	California Current
Vincent Échevin	IRD	France	Humboldt Current
Pierre Florenchie	UCT	South Africa	Anguola-Benguela Front
Hartmut Frenzel	UCLA	USA	California Current
Bruno Galand	BOOST	France	Mediterranean Sea
Nicolas Guillou	CETMEF	France	Mer d'Iroise
Steven Herbette	JRC	Italy	Canary Current
Ashley Johnson	M&CM	South Africa	Benguela Current
Vamara Kone	IRD	South Africa	Benguela Current
Eric Machu	IRD	South Africa	Benguela Current
Sergio Major	IMARPE	Peru	Peru Current
Patrick Marchesiello	IRD	France	California and Canary Currents
Alejandro Pares Sierra	CICESE	Mexico	Mexican-Caribbean
Jose Pasapera	IMARPE	Peru	Peru Current
Álvaro Júdece Peliz	University of Aveiro	Portugal	Iberic Peninsula
Pierrick Penven	IRD	France	Benguela and Peru Currents
Hyodae Seo	UCSD	USA	
Alain Serpette	SHOM	France	Bay of Biscay
Julio Sheinbaum	CICESE	Mexico	Mexican-Caribbean
Bamol Ali Sow	LPA	Senegal	Canary Current
Sabrina Speich	UBO	France	Iberic Peninsula, Agulhas Current, Bay of Biscay
Jorge Tam	IMARPE	Peru	Peru Current
Frederic Vandermeirch	IFREMER	France	Bay of Biscay
Doris Veleda	UFPE	Brasil	Brazilian Current
Kelin Zhuang	CGS	China	China Sea

Table 1: Toolbox users.

References

- Beckmann, A., Haidvogel, D.B., 1993. Numerical simulation of flow around a tall isolated seamount. Part I: Problem formulation and model accuracy. *Journal of Physical Oceanography* 23, 1736-1753.
- Blanke, B., Roy, C., Penven, P., Speich, S., McWilliams, J.C., Nelson, G., 2002. Linking wind and upwelling interannual variability in a regional model of the southern Benguela, *Geophysical Research Letters* 29, 2188-2191.
- Blayo, E., Debreu, L., 1999. Adaptive mesh refinement for finite-difference ocean models: First experiments. *Journal of Physical Oceanography* 29, 1239-1250.
- Conkright, M.E., R.A. Locarnini, H.E. Garcia, T.D. O'Brien, T.P. Boyer, C. Stephens, J.I.

- Antonov, 2002. World Ocean Atlas 2001: Objective Analyses, Data Statistics, and Figures, CD-ROM Documentation. National Oceanographic Data Center, Silver Spring, MD, 17 pp.
- Da Silva, A.M., Young, C.C., Levitus, S. 1994. Atlas of surface marine data 1994, Vol. 1, algorithms and procedures, NOAA Atlas NESDIS 6, U. S. Department of Commerce, NOAA, NESDIS, USA, 74 pp.
- Debreu, L., 2000. Raffinement adaptatif de maillage et méthodes de zoom - application aux modèles d'océan, 2000, Ph.D. thesis, Université Joseph Fourier, Grenoble.
- Debreu, L., Blayo, E., 2003. AGRIF: Adaptive Grid Refinement In Fortran. *submitted to ACM Transactions on Mathematical Software - TOMS*.
- Debreu, L., Vouland, C., 2003. AGRIF: Adaptive Grid Refinement In Fortran. [Available online <http://www-lmc.imag.fr/IDOPT/AGRIF/index.html>].
- Di Lorenzo, E., Miller, A.J., Neilson, D.J., Cornuelle, B.D., Moisan, J.R., 2003. Modeling observed California Current mesoscale eddies and the ecosystem response . International Journal of Remote Sensing, in press.
- Egbert, G., Erofeeva, S., 2002. Efficient inverse modeling of barotropic ocean tides, Journal of Atmospheric and Oceanic Technology 19, 183-204.
- Flather, R.A., 1976. A tidal model of the northwest European continental shelf. Mémoires de la Société Royale des Sciences de Liège, 10, 141-164.
- Haidvogel, D.B., Arango, H.G., Hedstrom, K. , Beckmann, A., Malanotte-Rizzoli, P., Shchepetkin, A.F., 2000. Model Evaluation Experiments in the North Atlantic Basin: Simulations in Nonlinear Terrain-Following Coordinates. Dynamics of Atmospheres and Oceans 32 , 239-281.
- Haney, R.L., 1991. On the pressure force over steep topography in sigma coordinate ocean models. Journal of Physical Oceanography 21, 610-619.
- Jackett, D.R., McDougall, T.J., 1995. Minimal Adjustment of Hydrostatic Profiles to Achieve Static Stability. Journal of Atmospheric and Oceanic Technology 12, 381-389.
- Large, W.G., McWilliams, J.C., Doney, S.C., 1994. Oceanic vertical mixing: a review and a model with a nonlocal boundary layer parameterization. Reviews in Geophysics 32, 363-403.
- MacCready, P. M., R. D. Hetland, W. R. Geyer, Long-Term Isohaline Salt Balance in an Estuary. Continental Shelf Research, 22, 1591-1601.
- Marchesiello, P., McWilliams, J.C., Shchepetkin, A., 2001. Open boundary condition for long-term integration of regional oceanic models. Ocean Modelling 3, 1-21.
- Marchesiello, P., McWilliams, J.C., Shchepetkin, A., 2003. Equilibrium structure and dynamics of the California Current System. Journal of Physical Oceanography 33, 753-783.
- Penven, P., Roy C., Lutjeharms, J.R.E., Colin de verdière, A., Johnson, A., Shillington, F., Fréon, P., Brundrit, G., 2001. A regional hydrodynamic model of the Southern Benguela. South African Journal of Science 97, 472-476.
- Penven, P., Debreu, L., Marchesiello, P., McWilliams, J.C., 2004. Application of the ROMS embedding procedure for the Central California Upwelling System. Ocean Modelling, In revision.
- Shchepetkin, A.F., McWilliams, J.C., 1998. Quasi-monotone advection schemes based on explicit locally adaptive dissipation. Monthly Weather Review 126, 1541-1580.
- Shchepetkin, A.F., McWilliams, J.C., 2003. A method for computing horizontal pressure-gradient force in an ocean model with a non-aligned vertical coordinate. Journal of Geophysical Research 108.

- Shepikin, A.F., McWilliams, J.C., 2004. Regional Ocean Model System: a split-explicit ocean model with a free-surface and topography-following vertical coordinate. Submitted to Journal of Computational Physics.
- Smith, W.H.F., Sandwell, D.T., 1997. Global seafloor topography from satellite altimetry and ship depth soundings. *Science* 277, 1957-1962.