CBIOMES-global Documentation

Release 0.1.0

Gael Forget

Mar 20, 2020

CONTENTS:

Here, you will learn about simple methods that are available to download, visualize, and compute *CBIOMES-global* solutions. The prototype solution, *CBIOMES-global (alpha version)*, is a global ocean state estimate that covers the period from 1992 to 2011. It is based on [FCH+15] for ocean physics and [DHJ+15] for marine biogeochemistry and ecosystems.

CHAPTER

ONE

DOWNLOADS

The following recipes allow users to download *CBIOMES-global* model output (Section 1.1), tools to manipulate this output (Section 1.2), and the *CBIOMES-global* model setup (Section 1.3).

1.1 Output

The *CBIOMES-global (alpha version)* monthly climatology has been interpolated to a $1/2 \times 1/2°$ grid (see *Interpolate Output*) and converted to Netcdf files (see *Convert to Netcdf*) that can readily be visualized using any *Netcdf*-enabled software (e.g., Panoply under *Windows*, *Linux*, or *macOS*). It is available via:

- The MIT-CBIOMES opendap (e.g. visit this page)
- The Simons CMAP data service (go to this page)

These servers can be accessed programatically. See e.g. notebooks/DarwinCmap* for cmap examples in Python and Julia. Or one can use opendap e.g. in Julia as follows:

```
using NCDatasets, Plots
function test_opendapp()
srv="http://engaging-opendap.mit.edu:8080/thredds/dodsC/las/id-fba1de9aef/"
fil="data_usr_local_tomcat_content_cbiomes_20200206_17_NutrientTendencies_gTr01.0001.
↪nc.jnl"
test=Dataset(srv*fil)
tmp=test["gTr01"][:,:,1,1]
tmp[findall(tmp.<=-0.99e34)].=NaN
return tmp
end
heatmap(test_opendapp())
```

The underlying climatology, on the native model grid, is in turn available in this dropbox folder which contains additional documentation like the *solution booklet* that extensively depicts *CBIOMES-global (alpha version)* and native grid output that allows for accurate computational analyses (see Section 1.2 and Section 2.3).

1.2 Tools

Accurate computational workflows often require native grid output rather than interpolated output. Manipulation of native grid output, which users may have either downloaded (Section 1.1) or generated themselves (Section 2.2), is facilitated by the gcmfaces toolbox [FCH+15]. This toolbox is available for Matlab (proprietary) and Octave (free); it can be downloaded as follows:

```
git clone https://github.com/MITgcm/gcmfaces
```

The Section 2.1 and Section 2.3 applications rely on gcmfaces along with *CBIOMES*-specific codes which can in turn be downloaded as follows:

```
git clone https://github.com/darwinproject/CBIOMES-Processing.m CBIOMES-tools
git clone https://github.com/gaelforget/CBIOMES CBIOMES-setup
```

1.3 Model

The Section 2.2 recipes to *Compile And Run* the ocean model allow users to reproduce *CBIOMES-global* solutions. This requires the ocean model *Code, Setup, And Input* (surface boundary conditions, initial conditions, grid, etc.). Provided scripts, *dowload_setup.sh* and *dowload_input.sh*, download these various elements along with the Section 1.2 tools and organize directories as expected by the Section 2.2 recipes (*Recommended Directory Organization*).

Code, Setup, And Input

```
git clone https://github.com/gaelforget/CBIOMES CBIOMES-setup
bash ./CBIOMES-setup/tools/shell/download_setup.sh
bash ./CBIOMES-setup/tools/shell/download_input.sh
```

Recommended Directory Organization

```
MITgcm/
  model/   (MITgcm core code)
  pkg/
    gud/   (bgc + ecosystem modules)
    + other subdirectories
  mysetups/
    CBIOMES/
      code/                 (compile-time settings)
      input/                (run-time settings)
      tools/                (shell and other scripts)
      inputs_drwn3/         (binary model input)
      forcing_baseline2/    (binary model input)
      inputs_baseline2/     (binary model input)
      + other subdirectories
  tools/
    genmake2        (makefile generation script)
    build_options   (various compiler options)
    + other subdirectories
  + other subdirectories
```

Note: *inputs_drwn3/* is not currently installed by *dowload_input.sh* even though it is needed for the biochemistry and ecology. Please contact us to get access to the corresponding dropbox folder and place its content as shown in *Recommended Directory Organization*.

CHAPTER TWO

COMPUTATIONS

This following recipes allow users to visualize CBIOMES-global (alpha version) (Section 2.1), reproduce it using the ocean model (Section 2.2), post-process native grid output (Section 2.3), and experiment with the ocean model (Section 2.4).

2.1 Visualization

Each *Netcdf* file generated via *Convert to Netcdf* recipes satisfies the so-called CF conventions. This is the case of all *Netcdf* files found in the *CBIOMES-global (alpha version)* monthly climatology folder, which can thus readily be ingested and visualized using software such as Panoply and many others. Furthermore, each *nctiles* collection generated via *Convert to Netcdf* recipes can be ingested by gcmfaces which provides several methods to visualize native grid output.

```
To be Continued ...
```

2.2 Model Run

The following recipe will *Compile And Run* the ocean model as needed to reproduce *CBIOMES-global (alpha version)*. Documentation of the model configuration is provided in [FCH+15] (global grid + physics) and in the Section 1.1 climatology folder (biochemistry + ecology).

Prerequisites

The following recipe assumes that *Code, Setup, And Input* have been installed as shown in *Recommended Directory Organization* (see Section 1.3). Running the model further requires a computer cluster equipped with gcc and gfortran, or alternative compilers, and MPI libraries for parallel computing (see MITgcm documentation). Netcdf libraries are also useful but not required.

Compile And Run

```
#1) go to model setup directory
cd MITgcm/mysetups/CBIOMES/

#2) compile model in build/
bash tools/shell/compile_model.sh

#3) prepare run/ directory
bash tools/shell/prep_rundir.sh

#4) run model on 360 cores
cd run
mpiexec -np 360 ./mitgcmuv
```

Note: Two modifications of this workflow can be needed depending on the computing environment. First, on most computer clusters, `mpiexec` (or `mpirun`) is called via a queuing system rather than directly from the command line as shown here. Second, *compile_model.sh* uses an *linux_amd64_ifort+mpi_ice_nas* option file that suits the *ifort* compiler on the *pleiades* computer. Option files that may better suit other compilers and computers can be found in the `MITgcm/tools/build_options/` subdirectory.

2.3 Processing

The following recipes have been used to post-process model output on its native grid – the so-called *LLC90* grid [FCH+15]. Covered topics include plotting, interpolating, and formatting.

Prerequisites

- Matlab (proprietary) or Octave (free) plus the Section 1.2 Toolboxes.
- Grid files from either this nctiles_grid/ or the Section 2.2 *run/* directory.

Plot And Analyze

The following recipe reads in native grid output in binary format (generated directly by *MITgcm*) or in tiled *Netcdf* format (created from *MITgcm* output via *Convert to Netcdf*). It decpicts various variables and compile all of the generated plots in the so-called *solution booklet*.

```
To be continued...
```

Interpolate Output

The following *Matlab* recipe reads in binary, native grid, output from *MITgcm* and interpolates it to a regular $1/2 \times 1/2°$ longitude-latitude grid. The *CBIOMES-global (alpha version)* interpolated climatology was created by feeding the results of this recipe to the *Convert to Netcdf* recipe.

```
p = genpath([pwd '/tools/matlab/']); addpath(p);
process2interp_driver([pwd '/run/']);
!mv run/diags/diags_interp_tmp run/diags_interp
```

Convert to Netcdf

The first recipe below takes the result of *Interpolate Output* and converts it into *Netcdf* files.

```
p = genpath([pwd '/tools/matlab/']); addpath(p);
interp2nctiles_driver([pwd '/run/']);
!mv run/diags_interp/nctiles_tmp run/diags_interp_netcdf
```

The next recipe below instead reads in native grid, binary output as generated by *MITgcm* directly and converts it to a tiled *Netcdf* format known as *nctiles* [FCH+15].

```
p = genpath([pwd '/tools/matlab/']); addpath(p);
process2nctiles_driver([pwd '/run/']);
!mv run/diags/nctiles_tmp run/diags_nctiles
```

Note: The above recipes generally extract variable names, descriptions, units, etc. directly from the *available_diagnostics.log* file generated by *MITgcm* with two potential exceptions: 1) in interpolated cases, vector fields should be converted to eastward / northward components, and variable names modified accordingly; 2) in all cases, passive tracer variables can be associated with more explicit names based on *PTRACERS_names* defined in *data.ptracers*.

2.4 Experiments

This section outlines methods that allow users to modify and experiment with model settings – this is often useful to better understand and improve upon existing solutions.

```
To be continued...
```

BIBLIOGRAPHY

[DHJ+15] S. Dutkiewicz, A. E. Hickman, O. Jahn, W. W. Gregg, C. B. Mouw, and M. J. Follows. Capturing optically important constituents and properties in a marine biogeochemical and ecosystem model. *Biogeosciences*, 12(14):4447–4481, 2015. URL: https://www.biogeosciences.net/12/4447/2015/, doi:10.5194/bg-12-4447-2015.

[FCH+15] G. Forget, J.-M. Campin, P. Heimbach, C. N. Hill, R. M Ponte, and C. Wunsch. ECCO version 4: an integrated framework for non-linear inverse modeling and global ocean state estimation. *Geoscientific Model Development*, 8(10):3071–3104, 2015. URL: http://www.geosci-model-dev.net/8/3071/2015/, doi:10.5194/gmd-8-3071-2015.