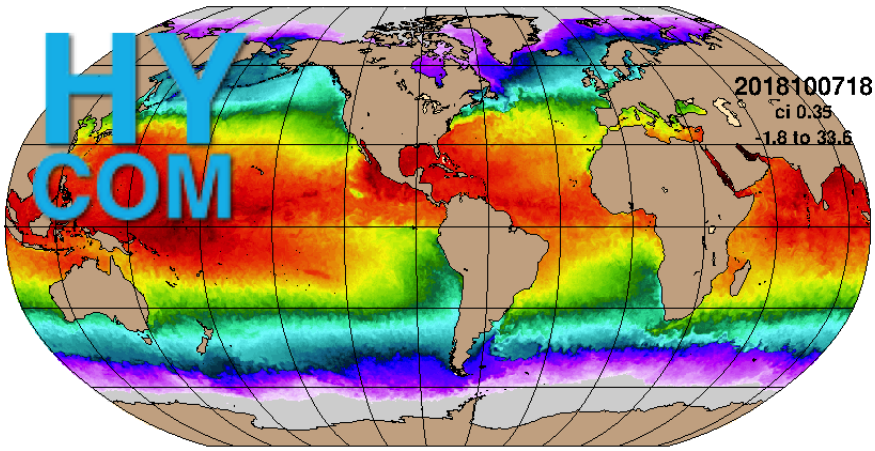
C++ NetCDF Interface for the HYCOM Global Ocean Model

User Manual

Blake Cole, Massachusetts Institute of Technology

August 22nd, 2018



Contents

1 Introduction

HYbrid Coordinate Ocean Model (HYCOM)

The HYCOM consortium is a multi-institutional effort sponsored by the National Ocean Partnership Program (NOPP), as part of the U. S. Global Ocean Data Assimilation Experiment (GODAE), to develop and evaluate a data-assimilative hybrid isopycnal-sigma-pressure (generalized) coordinate ocean model (called HYbrid Coordinate Ocean Model or HYCOM). The GODAE objectives of three-dimensional depiction of the ocean state at fine resolution in real time, provision of boundary conditions for coastal and regional models, and provision of oceanic boundary conditions for a global coupled ocean-atmosphere prediction model, are being addressed by a partnership of institutions that represent a broad spectrum of the oceanographic community. For more information, please visit the HYCOM website: `https://hycom.org/`.

Accessing HYCOM NetCDF Data

According to UCAR, netCDF (Network Common Data Form) is a set of software libraries and self-describing, machine-independent data formats that support the creation, access, and sharing of array-oriented scientific data. However, while netCDF effectively enables the efficient storage exceedingly large datasets, accessing these datasets – or small pieces of them – is not always an intuitive exercise.

The ***netcdf_hycom*** application is designed to extract a user-specified subset of spatio-temporal data from the HYCOM Global Ocean Forecast System *(GOFS 3.1)*, via a user-friendly C++ interface. The GOFS 3.1 model contains four 4D fields, including:

- ocean temperature [°C]
- salinity [psu]
- current x-velocity [m/s]
- current y-velocity [m/s]

It also includes five 3D fields, which are defined at only one depth, including:

- surface level [m]
- benthic ocean temperature [°C]
- benthic salinity [psu]
- benthic current x-velocity [m/s]
- benthic current y-velocity [m/s]

As of August 22nd, 2018, ***netcdf_hycom*** is only capable of extracting two of the available nine fields: *ocean temperature* and *salinity*. However, it could be expanded to read the other fields with truly minimal effort. Once the data is extracted from the HYCOM database, it is stored in 4D *c-style* arrays. If desired, the user can request that a new netCDF file be created, representing the desired subset of interest. This functionality could be particularly useful when conducting field work in areas without reliable internet access.

To ensure that the correct data has been written, the user may find it useful to inspect the new netCDF file using ***netcdf_hycom_readonly***. This auxiliary application looks for the netCDF file created by *netcdf_hycom*, in a fixed directory `[netcdf_hycom/data/salt_temp_4D.nc]`, inspects it, and prints its contents.

2 System Requirements

1. ***Linux Ubuntu 18.04.01 LTS***

   (a) Required:

      i. GNU Compiler (`g++`)

      ii. NetCDF Library for C++ (`libnetcdf-c++4-dev`)
      Dependencies:
      - `hdf5-helpers`
      - `libaec-dev`
      - `libcurl4-gnutls-dev`
      - `libhdf5-cpp-100`
      - `libhdf5-dev`
      - `libnetcdf-dev`

   (b) Recommended:

      i. NetCDF Viewer (`nvciew`)

      ii. NetCDF Operators (`nco`)

      iii. Climate Data Operators (`cdo`)

      iv. Optional netCDF C++ Libraries
      - `libnetcdf-c++4` (legacy)
      - `libnetcdf-c++4-1`
      - `libnetcdf13`

2. ***macOS High Sierra 10.13.5***

   (a) Required:

      i. GNU Compiler (`g++`)

      ii. NetCDF Library for C++ (`netcdf-cxx4`)
      Dependencies:
      - `hdf5`
      - `netcdf`

   (b) Recommended:

      i. NetCDF Viewer (`nvciew`)

      ii. NetCDF Operators (`nco`)

      iii. Climate Data Operators (`cdo`)

3 Download via GitHub

This program is available free of charge, under a Creative Commons License, via GitHub.

`https://github.com/blakecole/netcdf-hycom.git`

4 Compiling

The netCDF C++ libraries downloaded via *MacPorts* or *apt-get* will be located in non-default directories; accordingly, a few command line switches are required to compile correctly. The examples below are intended to be instructional; while they will work "out-of-the-box" for many users, they are provided for guidance only. It is important to remember that the relevant libraries and header files may exist in other non-default locations, depending on the user's operating system and build structure.

Linux Ubuntu 18.04.01 LTS

```
$ g++ -o netcdf_hycom netcdf_hycom.cpp -Wall -I/usr/include
-L/usr/lib/x86_64-linux-gnu -lnetcdf_c++4
```

macOS High Sierra 10.13.5

```
$ g++ -o netcdf_hycom netcdf_hycom.cpp -Wall -I/opt/local/include
-L/opt/local/lib -lnetcdf_c++4
```

Description of Command Line Switches:

1. Show (almost) all compiler warnings.
   `-Wall`
2. Augment the g++ "include" path (where the .h header file lives).
   `-I/..`
3. Augment the g++ "link" path (where the .a static library file lives).
   `-L/..`
4. Specify which library needs to be included (libnetcdf_c++4).
   `-lnetcdf_c++4`

5 Usage

Provided you have managed to successfully compile the program on your machine, you are ready to extract a chunk of HYCOM model data, at your leisure! Thankfully, ***netcdf_hycom*** interfaces directly with the HYCOM database via OPeNDAP, so you will not need to visit the HYCOM website. All you need is an internet connection!

5.1 Specifying the Bounding Box

The program allows users to specify a spatio-temporal bounding box via one of two methods:

1. **Command Line Switches**

`--tstart=[STRING]`	:	time: start time, `[YYYY:MM:DD]`
`--tstop=[STRING]`	:	time: end time, `[YYYY:MM:DD]`
`--depthmin=[FLOAT]`	:	depth: shallowest depth
`--depthmax=[FLOAT]`	:	depth: deepest depth
`--latmin=[FLOAT]`	:	latitude: southern edge
`--latmax=[FLOAT]`	:	latitude: northern edge
`--lonmin=[FLOAT]`	:	longitude: western edge

`--lonmax=[FLOAT]`	:	longitude: eastern edge
`--newfile=true`	:	write new netCDF file

If specifying a date range from the command line, using `--tstart=` and `--tstop=`, it is *essential* that the correct format be used: `[YYYY:MM:DD]`. Any and all other formats, or the accidental exclusion of brackets, will cause the program will fail.

It is not currently possible to specify start and stop hours via the command line switches. If this level of precision is required, simply run the program without command line switches, and enter the data when prompted during runtime (see 5.1.2: User Input During Runtime).

Unlike user input collected during runtime, command line switches are in no way quality-controlled by the program; it is assumed that the user is providing appropriate values. Inappropriate inputs will cause the program to fail.

2. **User Input During Runtime**

   The program will first look for a total of (8) parameters provided on the command line, to fully define a spatio-temporal bounding box: minimum and maximum values for time, depth, latitude, and longitude. In addition, the user must specify whether or not a new netCDF file will be written, using the `--newfile=true` command line switch. Use the `--help` command for a full list of acceptable command line inputs.

   If an insufficient number of command line switches are detected, the user will be prompted to enter each spatio-temporal boundary parameter individually. These inputs *are* quality-controlled; the program will continue to query the user until a valid entry is detected.

   This is the safest and most robust means of parameter entry, and it is recommended that most users avoid using command line switches. However, if running many queries repeatedly, it may be more convenient to use the command line switches; in this case, simply take care to ensure that the switches are correctly formatted.

Finally, it is important to understand that HYCOM model data is partitioned into one-year chunks, and stored in year-specific directories. Accordingly, if older model data is required (i.e. more than one year from present), one must edit the field *dataURL* (line 68) of ***netcdf_hycom.cpp*** to reference the appropriate directory.

For demonstration video, please visit: `https://vimeo.com/338568635`

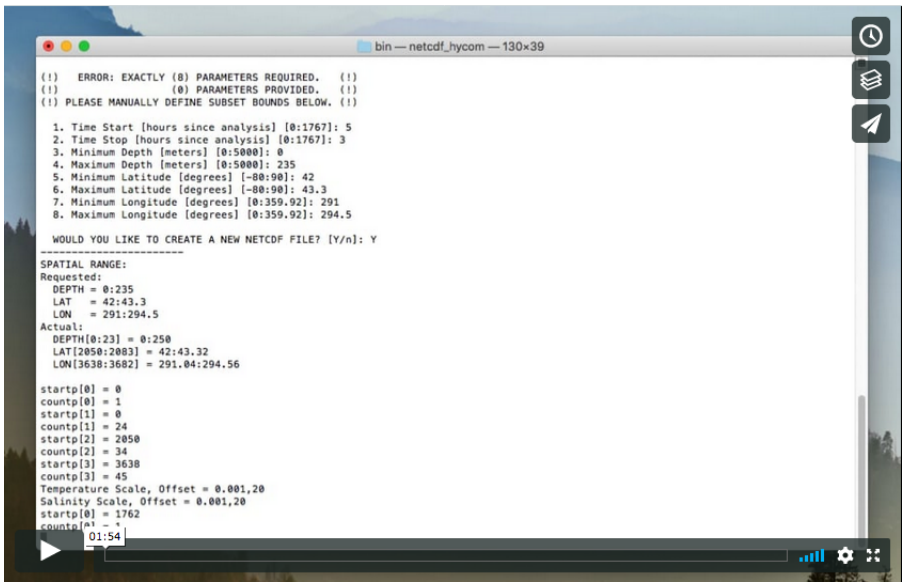


Figure 1: Screenshot from ***netcdf_hycom*** *demonstration video.*

6 Contact

For assistance with this program, please contact:

- **Blake Cole**
  `blerk@mit.edu`

- **Henrik Schmidt**
  henrik@mit.edu