NCCOOS Baroclinic Near-operational Nowcast/Forecast Model System

Alfredo Aretxabaleta[1] et al

1. Abstract

This document describes the development of a baroclinic Nowcast/Forecast (NC/FC) system for the South Atlantic Bight region. The current status of the NC/FC system is described, including general procedure, model inputs and setup, and basic outputs. Some preliminary results and a simple comparison with observations is presented. Additionally, some future directions and developments are proposed.

2. Introduction

The South-East Atlantic Coastal Ocean Observing System (SEACOOS, www.seacoos.org) is a collaborative university partnership that collects, manages, and disseminates integrated regional ocean observations and information products for the coasts of North Carolina, South Carolina, Georgia, and Florida [*Seim et al.*, 2003]. Funding for this effort has been provided by the Office of Naval Research.

The SEACOOS effort includes the implementation of a model system for nowcasting and forecasting (NC/FC) the coastal ocean. The system extends from the Virginia-North Carolina border to the Mississippi River and is a collaboration of several institutions (U. of North Carolina, U. of Miami, U. of South Florida). The region of interest for the UNC component of the NC/FC system is the South Atlantic Bight (SAB) with a special focus on the North Carolina coast as part of the North Carolina Coastal Ocean Obeserving System (NCCOOS).

The focus of recent work has been to produce forecast simulations of circulation and mass fields for the SAB region. Two separate efforts were conducted to produce these simulations:
1) A barotropic simulation of the coastal ocean was developed [*Blanton et al.*, 2004a]. This simulation has been producing forecasts since 2003. The fields provided are water level and depth-averaged velocity.
2) A baroclinic simulation has been developed over the last year and the system is in the process of becoming automated. This simulation provides temperature, salinity, velocity, turbulence and water level fields.

This document explains the specific technical details of the current baroclinic simulation. Additional information about the basic setup of communications, resource locations and model parameters can be found in the barotropic model setup technical report [*Blanton et al.*, 2004a],

[1]University of North Carolina at Chapel Hill, NC

as a similar procedure is followed by the baroclinic system. In contrast with that report, this document focuses on the South Atlantic Bight and coastal North Carolina implementation as part of NCCOOS.

3. General procedure

The baroclinic implementation of the NCCOOS NC/FC system includes forcings by atmospheric wind stress and heat flux at the model surface, and tidal elevation, temperature, salinity and far-field elevation at the model open-ocean boundary.

The main goal of the NC/FC system is to provide circulation and mass field information for the SAB region available through a web interface. Both hindcast and forecast products are provided. Currently the hincast length is 8-9 days and the forecast is 14 days (Figure 1). The time availability of the forecast fields depends on the delay in the posting times of the input fields (atmospheric and basin-scale forcings). **Figure 1.**

The current implementation consists of a one-way nested system that uses initial and boundary conditions from the basin-scale operational HYCOM (HYbrid Coordinate Ocean Model) simulations for the North Atlantic. The atmospheric forcings are extracted from available National Center for Environmental Predictions (NCEP) mesoscale meteorological model products. A schematic of the procedure is presented in Figure 2. The specifics of the current setup are presented in the following section. **Figure 2.**

4. Model setup

4.1. Model

The model used for the regional-scale (SAB) ocean simulations is Quoddy [*Lynch and Werner*, 1991; *Lynch et al.*, 1996], which is a 3-D, nonlinear, prognostic, tide-resolving, finite element model with level 2.5 turbulence closure [*Mellor and Yamada*, 1982]. The model uses a higher order advection scheme developed by *Kliem* [2004]. The advection scheme is especially necessary for simulations for periods longer than a week, so that the over- and under-shooting and unrealistic smoothing effects can be reduced. The advection scheme (PCTNQ) uses the point-wise corrected transport advection with nodal quadrature to reduce the computational requirements.

4.2. Domains

The main requirement for the use of the PCTNQ advection scheme is the creation of two “concurrent” meshes: one fine mesh with linear basis functions and a second coarser mesh with quadratic basis functions (Figure 3). One quadratic element covers exactly four of the linear elements. The basic domain is a finite element mesh, that includes the SAB continental shelf from NE Florida to the Virginia-North Carolina border as well as the open ocean region west of 72W longitude. The finest horizontal resolution is 3 km near the coast and it increases to around 40 km in the open ocean. **Figure 3.**

4.3. Inputs

The initial and open boundary conditions are extracted from a basin-scale model simulation. The basin-scale (North Atlantic) model uses the HYbrid Coordinate Ocean Model (HYCOM, http://hycom.rsmas.miami.edu, *Bleck* [1998]; *Chassignet et al.* [2001]). HYCOM is run operationally by the Naval Research Laboratory as part of the Global Ocean Data Assimilation Experiment (GODAE). The HYCOM products have a temporal resolution of one day and a spatial resolution of $1/16^o$. The operational run does not include forcing by tides. A goal of the HYCOM/GODAE initiative is to provide initialization products to regional and limited-area coastal models, such as the one used in these simulations.

The heat flux and wind stress forcings are extracted from the NCEP operational meteorological solutions (NAM). The numerical weather prediction model currently used is ETA. The current setup of ETA on the AWIP12 output grid has a horizontal resolution of 12 km and 60 vertical layers on a domain covering most of North America, with a temporal output resolution of 6 h (four solutions daily). A complete description of the atmospheric product and its manipulation is available in *Blanton et al.* [2004a].

The tidal boundary conditions were obtained from the latest ADCIRC tidal database (TDB, *Blanton et al.* [2004b]). The TDB provides an important improvement from previous tidal estimates in this region because of the inclusion of the Estuaries and Tidal Inlet Complex (ETIC) in the SAB, which extends from the Florida-Georgia border to North Carolina. In the current implementation eight tidal constituents are used: M_2, S_2, N_2, K_2, O_1, K_1, P_1 and Q_1.

4.4. Outputs

The main NC/FC output is a NetCDF file posted in the NCCOOS web server (www.nccoos.org). The file includes hourly values of surface temperature, salinity, velocity and elevation; bottom velocity and depth-averaged velocity. The variables are interpolated to a regular grid, instead of being available on the native unstructured grid. Additionally the file includes 4 vertical transects across the shelf and shelf-break (Figure 4) and profiles of all the model variables at 4 locations: NDBC buoy 41008, SABSOON towers R2 and R6, and a point in open ocean over the Blake Plateau. The visualization of the model variables at the profile locations is still being developed. Eventually, the NetCDF files will contain the native model variables, so that direct representation without interpolation can be accomplish and so that the file can be used as hotstart files for future simulations. **Figure 4.**

4.5. Web output for status check

The production of status and diagnostic figures is done locally at reba.marine.unc.edu after the completion of the weekly simulation. An example of the matlab functions to produce the diagnosis figures can be found in ∼/opnml/scratch/hycom_oper .

There is a status website: http://www.unc.edu/∼hseim/sablam/seacoos_operat/seacoosrun.html that includes figures for several fields: surface temperature and depth averaged velocity (most current fields and animations), temperature along several transects both for the shelf and the open ocean, and comparisons between observed and model temperature at two locations: bottom temperature at the R2 tower and surface temperature at the Gray's Reef buoy (41008).

A very preliminary evaluation of the model simulations indicates a tendency for the solutions to develop unrealistic features near the northern and southern boundaries of the model domain (Figure 5). These features are more noticiable near the end of the forecast time. The features might be the result of inconsistencies between the internal mass and flow fields and the elevation imposed on the boundary. **Figure 5.**

Preliminary comparisons between in situ observations over the shelf and model temperatures reveal a tendency for the regional simulation to improve the predictive baroclinic skill when compared with the HYCOM estimate for that point over the shelf (Figure 6). The skill of the bottom temperature at R2 is generally better than the skill of the surface temperature at Gray's Reef (Figure 7). This is possibly associated with unrealistic heat flux being applied as surface boundary conditions. **Figure 6.**

Figure 7.

5. Runtime procedure

There are four different machines involved in the NC/FC system: 1) The local managing machine (reba.marine.unc.edu) that creates the model set-up, controls the runs and produces the status and diagnosis figures. 2) The machine where the atmospheric forcing files reside (eta.marine.unc.edu). 3) The computer cluster where the computations take place (happy.isis.unc.edu). 4) The server where the final output file resides and is accesible from the outside (nemo.isis.unc.edu) via OPeNDAP.

5.1. Current steps

The current procedure includes several non-automatic steps that are necessary in order for the simulation to be completed. At this point, there are 12 steps that eventually should be made automatic to achieve fully automated runs.

- Set up local machine (reba.marine.unc.edu) to see the atmospheric forcing files from eta.marine.unc.edu. To do that, as an administrator user (su) create and link NCEP directory.

>>mkdir /Volumes/NCEP
>>mount_nfs eta.marine.unc.edu:/dissert/NCEP /Volumes/NCEP

- Get HYCOM files from the Miami server:
  >> /afs/isis/depts/marine/opnml/matlab/local/hycom/get_hycom_output_reba.pl
  or
  >> ~/ws/get_hycom_output_reba.pl
  This command places the files in /Users/hycom/*cast
- Copy the operational directory to your working directory
  >> cp ~/opnml/scratch/hycom_oper .
  The source and input file directories need to be copied as well.
- Open the master version of the files to setup the simulations.
  >> co -l master_hycom_quoddy_prep.m
  >> co -l hycom_quoddy_setup.m
- Modify HOMEDIR in hycom_quoddy_setup.m
- Run master_hycom_quoddy_prep.m in MATLAB
- Copy entire simulation directory to happy.isis.unc.edu
- ssh to happy.isis.unc.edu
- Go to stage/diag directory and submit diagnostic job.
- When diagnostic run is finished, go to stage/prog directory and submit prognostic job.
- When prognostic run is finished, rename the output file to correct naming convention and copy it to server (nemo.isis.unc.edu) using the move2nemo function.
- Copy output file to local machine and produce status and diagnostic figures.

The files get_hycom_output_reba.pl, master_hycom_quoddy_prep.m and hycom_quoddy_setup.m are included in the Appendix.

5.2. Run sequence and timeline

The run sequence includes a 4-day diagnostic (frozen fields) run to spin-up the tides and a 3-week prognostic run. This is generally the case except when ocassionally the total length of time of the HYCOM product simulation changes.

The timeline for the different components of the simulation is presented in Figure 8. Under these conditions, the last available atmospheric fields are maintained constant in time until the end of the simulation. A better method would be to reproduce the entire last day of forcing, instead of the last file available, so that a more realistic heat flux cycle is included. **Figure 8.**

In fact, the real timeline often differs from the one presented in Figure 8. Over the last several weeks the HYCOM products are not available until almost the end of its valid forecast. This implies that the delay between the $Z = 0$ time and the processing time of the atmospheric fields (t_1) is almost 14 days. In this situation, there is no need to duplicate the final atmospheric file because there are available atmospheric fields for the entire simulation. This is shown in Figure 9. **Figure 9.**

Table 1 shows the availability of the different output files over the last month. There is usually some delay between HYCOM's $Z = 0$ and the posting time of the HYCOM solution. The posting delay of the Quoddy regional NC/FC is usually around 36 hours. **Table 1.**

6. Formats and namings

The NC/FC output file is a NetCDF file posted in nemo.isis.unc.edu. The location is /seacoos/data/nc-coos/model_data/quoddy/forecast/ and can be accessed through: http://nemo.isis.unc.edu/cgi-bin/nph-dods/data/nc-coos/model_data/quoddy/forecast/

The file name format is:
Q_HY_NAM_nq_yyyymmddThhmm_yyyymmddThhmm_yyyymmddThhmm.nc

-. Q is the identifier of the regional hydrographic model. In this case: a Quoddy simulation.

-. HY is the identifier of the basin scale hydrographyc forcing (initial condition and open-ocean boundary conditions). In this case: a HYCOM simulation.

-. NAM is the identifier of the atmospheric forcing. In this case: NAM (EDAS, ETA)

-. nq is the identifier of the native simulation grid. In this case: seacoos_coarse_sab_nq

-. The first time stamp (e.g., 20060220T0000) corresponds to the true initialization time. In this case it is the same as the initialization of the HYCOM hindcast. The format is yyyymmddThhmm

-. The second time stamp (e.g., 20060301T0000) corresponds to the nowcast time. In this case it is the same as the HYCOM nowcast time.

-. The third and final time stamp (e.g., 20060315T0000) corresponds to the final time of the simulation. In this case it is the same as the HYCOM final time.

7. Steps towards operationality

The entire model run should be executed in cron. The main steps that need to occur in cron are: 1) Implement the HYCOM file retreaval. 2) Create the entire model run directory. 3) Execute the model code. 4) Produce the status figures and place output file in server.

8. Future developments

Inclusion of local river discharge

Currently no river discharge is included in the NC/FC implementation. The model already deals with this field but issues with the consistency of a direct access to USGS discharge information forced a temporary no-river discharge implementation. Communication has been established with the NCEP group (Carlos Lozano) to share their real-time access to river discharge data from USGS.

Extension of northern boundary to Chesapeake Bay

In order to avoid some of the issues with the unrealistic flow in the vicinity of the northern boundary, the possibility of extending the domain northward to the mouth of Chessapeake Bay should be considered. This extended domain will allow a better (more realistic) comparison with the surface velocity observations from the NCCOOS HF radar system.

Additionally, the Chesapeake Bay fresh water discharge can be implemented by including a point source discharge extracted directly from the available HYCOM fields. This point discharge will assign the temperature, salinity and velocity from HYCOM to the regional domain boundary.

Addition of module to use COAMPS atmospheric forcing

The inclusion of an additional simultaneous model run that would use COAMPS atmospheric fluxes instead of NAM outputs should be considered. This additional run would allow comparisons between the two implementations. In the past, there have been concerns about the accuracy of NAM heat flux and wind stress, and a comparison with COAMPS forcings would provide the oportunity of evaluating its effects. Additionally, this new simulation would improve the robustness on the availability of model solutions by providing two simultaneous simulations. The daily COAMPS atmospheric fields are already being storaged on eta.marine.unc.edu. (/nopp/ldm/data/MET/FNMC)

Transition to new NCEP atmospheric model

A transition in the operational NCEP simulations from the current ETA model output to the WRF atmospheric model will occurr during June 2006. The new outputs will be provided in a different format (GRIB2) and tools need to be developed and implemented to maintain the NC/FC system.

Transition to NCEP-HYCOM

The oceanic modeling group at NCEP (Carlos Lozano's group) is in the final stages of developing an operational implementation of HYCOM with a focus on the East coast of the U.S. The validity of this operational product to produce initial and open-ocean boundary conditions for our regional NC/FC implementation needs to be evaluated.

Inclusion of sponge layer

Brian Blanton has been developing an implementation of a sponge layer for QUODDY. The inclusion of this development would likely help with the occasional appearence of unrealistic flow near the northern and southern boundaries of the regional domain.

Development of a coastal simulation

We should consider the development of an additional simulation on a coastal domain that would extend from the shelf-break to the coast and would resolve the complex estuarine system in our region of interest. This new domain would take advantage of the benefits of unstructured grids to appropriately resolve complex geometries. This implementation would provide an additional level of nesting, going from the basin-scale HYCOM simulation, to the regional simulation described above, to this new coastal estuary-resolving simulation. This run would allow for fully non-linear dynamics over the shelf, without the difficulties associated with steep density fields over steep topography in the shelf-break region.

9. APPENDIX

9.1. get_hycom_output_reba.pl

```
#!/afs/isis/pkg/isis/bin/perl -w
#/usr/bin/perl

use strict;
use warnings;
use Time::Local;
use File::Copy;
use File::Path;
use LWP::Simple;
use HTML::TokeParser;

my $hycomhost="asterix.rsmas.miami.edu";
my $srcurl="http://$hycomhost/datasets/ocean-prediction/atlantic/%s/%s/";
my $destdir="/Users/hycom/";
my $tempfile="/tmp/temptemptemp";
my ($set,@sets);
my ($time,@times);
my ($var,@vars);
my $hycomfilepattern="hycom_2.1_nat_1o12ml_%s_%s.nc";
my $status;
@vars=("ssh","mldpth","ubaro","vbaro","temp","salt","uvel","vvel","lthk");
@sets=("nowcast","forecast");
open(LOG,">-");

if (!-d $destdir){
   print LOG "$destdir DNE.  Create !!\n";
   exit 1; }

if (&check_hycom_server){ print LOG "$hycomhost is alive!!!\n"; }
else { print LOG "$hycomhost is dead!!!\n";
   exit 1; }

print LOG "Looping over nowcast times ...\n";
@times=&get_nc_times;
foreach $var (@vars){
   my $dir="$destdir/nowcast/$var";
   if (!-d $dir){ mkdir $dir; }
   foreach $time (@times){
      my $file=sprintf($hycomfilepattern,$var,$time);
```

```
      my $src=sprintf("$srcurl/$file.gz","nowcast",$var);
      my $dest=sprintf("$destdir/%s/%s/%s","nowcast",$var,$file);
      my $destgz=$dest . ".gz";
      my $com=sprintf("/sw/bin/wget --output-document=%s %s",$destgz,$src);
      if (!-e $dest){
         print LOG "$com\n";
         $status=‘$com‘;
         $status=‘gunzip $destgz‘;
      }
      else{ print LOG  "Already have $dest\n"; }
   }
}

print LOG "Looping over forecast times ...\n";
@times=&get_fc_times;
foreach $var (@vars){
   my $dir="$destdir/forecast/$var";
#    print "$dir\n";
   if (!-d $dir){ mkdir $dir; }
   foreach $time (@times){
      my $file=sprintf($hycomfilepattern,$var,$time);
      my $src=sprintf("$srcurl/$file.gz","forecast",$var);
      my $dest=sprintf("$destdir/%s/%s/%s","forecast",$var,$file);
      my $destgz=$dest . ".gz";
      my $com=sprintf("/sw/bin/wget --output-file=/tmp/wget-log --output-document=%s %s",$destgz,$src);
      if (!-e $dest){
         print LOG "$com\n";
         $status=‘$com‘;
         $status=‘gunzip $destgz‘;
      }
      else{ print LOG  "Already have $dest\n"; }
   }
}

exit 0;
sub get_fc_times{
   my $url="http://$hycomhost/datasets/ocean-prediction/atlantic/forecast/ssh/";
   my ($p,$text,$i,@times);
   getstore($url,$tempfile);
   $p = HTML::TokeParser -> new($tempfile);
   $i=-1;
   while (my $token = $p->get_token) {
      $text=$p->get_text();
      if ($text=˜/hycom/){
         $i++;
         $text=˜s/hycom_2.1_nat_1o12ml_ssh_//;
         $text=˜s/\.nc\.gz//;
         $times[$i]=$text;
      }
  }
  unlink($tempfile);
  return @times;
}

sub get_nc_times{
   my $url="http://$hycomhost/datasets/ocean-prediction/atlantic/nowcast/ssh/";
   my ($p,$text,$i,@times);
   getstore($url,$tempfile);
   $p = HTML::TokeParser -> new($tempfile);
   $i=-1;
   while (my $token = $p->get_token) {
      $text=$p->get_text();
```

```
      if ($text=~/hycom/){
         $i++;
         $text=~s/hycom_2.1_nat_1o12ml_ssh_//;
         $text=~s/\.nc\.gz//;
         $times[$i]=$text;
         #print "$i $text\n";
      }
  }
  unlink($tempfile);
  return @times;
}

sub check_hycom_server{
my ($f1,$f2,$url,$file);

$f1="index.html";
$f2="test.log";
system("rm -rf $f1 $f2");
$url="http://$hycomhost/datasets/ocean-prediction/atlantic/nowcast/ssh/";

system("/sw/bin/wget -o $f2 -O $f1 $url");

open(FIL,"<$f2") or die "Couldn’t open $f2\n";
undef $/;
$file=<FIL>;
close(FIL);
$/="\n";

system("rm -rf $f1 $f2");
if ($file=~/ERROR 404/){
   return 0;}
else{
   return 1;}
}
```

9.2. master_hycom_quoddy_prep.m

```
%function master_hycom_quoddy_prep(timenow)

% master control for nested quoddy/HYCOM simulations
% v1 : BOB : 2 Nov 2005
%%%%%
%%%%%
%%%%%
global stdoutfid stderrfid
stdoutfid=1;
stderrfid=2;

%%%%%
%%%%% initialize setup
%%%%%
hycom_quoddy_setup;

if exist(’timenow’)
else
   timenow=now;
end
[y,m,d,h,mn,s]=datevec(timenow);
current_day_of_week=datestr(timenow,8);
current_month=datestr(timenow,3);
```

```

%%%%%
%%%%% scan in "last_run" file
%%%%%
fid=fopen([HOMEDIR ’logs/last_run’],’r’);
if fid==-1, error(’Failed to open "last_run" file.’);,end
last_run_time=fscanf(fid,’%lf’,1);fgetl(fid);
last_run_osrc=fscanf(fid,’%s’,1); fgetl(fid);
last_run_fsrc=fscanf(fid,’%s’,1); fgetl(fid);
last_run_stat=fscanf(fid,’%s’,1); fgetl(fid);
fclose(fid);
last_run_day_of_week=datestr(last_run_time,8);

delta_t=timenow-last_run_time;
On_Log([’Number of days since last run : ’ num2str(delta_t)]);

if delta_t<7
   On_Err(’Not time to run.’);
else
   On_Log(’delta_t>7 OK to run’);
end

%if (delta_t<14 & strcmp(current_day_of_week,’Wed’))
%   On_Err(’Not time to run.’);
%else
%   On_Log(’delta_t<14 and day=Wed. OK to run’);
%end

current_day_of_week=’Wed’;
if strcmp(current_day_of_week,’Wed’)
   On_Log(’OK to run’);
else
   On_Err(’Wrong day of Week.’);
end

% check if HYCOM server is alive
tempurl=sprintf(’http://%s/datasets/ocean-prediction/atlantic/nowcast/ssh/’,HYCOM_HOST);
On_Log([’Checking for HYCOM_HOST=’ tempurl]);
err=is_hycom_server_alive(tempurl);
if ~err
   On_Log([tempurl ’ DNE.’]);
%   oceanextsrc(1)=[];  % remove HYCOM from list   (not being used right now)
   On_Log([’Using ’ oceanextsrc{1} ’ for IC/BC data’]);
else
   On_Log([’Using ’ oceanextsrc{1} ’ for IC/BC data’]);
end

%%%%%
%%%%%
%%%%%
%%%%% make run dirs in stage dir
%%%%%
[success,message,messageid] = mkdir(STAGEDIR,’hot’);
if ~success, On_Err(message),end
[success,message,messageid] = mkdir(STAGEDIR,’diag’);
if ~success, On_Err(message),end
[success,message,messageid] = mkdir(STAGEDIR,’prog’);
if ~success, On_Err(message),end
[success,message,messageid] = mkdir(STAGEDIR,’icq5s’);
if ~success, On_Err(message),end

switch oceanextsrc{1}
```

```
case ’CLIM’
   On_Log(’Building hotstart and bcs from CLIM ... ’);
   %%%%%
   %%%%% set model run dates
   %%%%%
   model_diag_start_date=datenum(timenow-dslag);
   [ydiagstart,mdiagstart,ddiagstart]=datevec(model_diag_start_date);
   model_hindcast_start_date=datenum(timenow-hclag);
   [yhindcaststart,mhindcaststart,dhindcaststart]=datevec(model_hindcast_start_date);
   model_forecast_end_date=datenum(timenow+fcastlength);
   [yforecastend,mforecastend,dforecastend]=datevec(model_forecast_end_date);

   climicq5=sprintf(’%s/F6.HOT.%s_nub.icq5’,CLIMDIR,current_month);
   system([’cp ’ climicq5 ’.bz2 ’,STAGEDIR,’hot/temp.icq5.bz2’]);
   system([’bunzip2 ’,STAGEDIR,’hot/temp.icq5.bz2’]);
   icq5=read_icq5([STAGEDIR,’hot/temp.icq5’]);
   % set start time in icq5 structure
   icq5.year=ydiagstart;
   icq5.month=mdiagstart;
   icq5.day=ddiagstart;
   err=write_icq5(icq5,[STAGEDIR,’hot/RESTART.icq5’]);
   err=write_icq5(icq5,[STAGEDIR,’icq5s/RESTART.icq5’]);

   On_Log(’Finished building hotstart  ... ’);

   hycflag=0;
   TSeta_extract(0,hycflag)

   On_Log(’Finished building bcs ... ’);

case ’HYCOM’
   On_Log(’Building hotstart and bcs from HYCOM ... ’);

   % get latest times from HYCOM server
   nc_times=get_nc_hycom_times;%nc_times(end-2:end,:)=[];
   fc_times=get_fc_hycom_times;

   % Get files from HYCOM server
   %system(’/afs/isis.unc.edu/depts/marine/opnml/matlab/local/hycom/get_hycom_output_reba.pl’);

   model_diag_start_date=datenum(nc_times(1,:))-dslag+hclag;
   [ydiagstart,mdiagstart,ddiagstart]=datevec(model_diag_start_date);
   model_hindcast_start_date=datenum(nc_times(1,:));
   [yhindcaststart,mhindcaststart,dhindcaststart]=datevec(model_hindcast_start_date);
   model_forecast_end_date=datenum(fc_times(end,:));
   [yforecastend,mforecastend,dforecastend]=datevec(model_forecast_end_date);

   % call the routines that get nc for 1 time to output icq5
   % loop over nc times.  codes should either return the icq5,
   % or write it ti disk.  Then, call routine that extracts bcs
   % from a spoecific icq5
   kdi1=datenum(nc_times(1,:));
   kdi2=datenum(nc_times(end,:));  nckd=kdi2;
   kdi=datenum(nc_times);
   ncflag=1;  % Nowcast flag for old format
   ncflag=0;  % Nowcast flag for format starting 10-Nov-05
   %hycom2quoddy(NC_URL,kdi1,kdi2,ncflag,0)
   %hycom2quoddy(NC_URL,kdi,ncflag,0)
   hycom2quoddy(NC_URL,kdi,ncflag,nckd)

   % Output RESTART icq5 for diagnostic run
   icq5=read_icq5([STAGEDIR,’icq5s/’,num2str(yhindcaststart),’_’,...
```

```
        num2str(mhindcaststart),’_’,num2str(dhindcaststart),’_HC.HOT.icq5’]);
   % set start time in icq5 structure
   icq5.year=ydiagstart;
   icq5.month=mdiagstart;
   icq5.day=ddiagstart;
   err=write_icq5(icq5,[STAGEDIR,’hot/RESTART.icq5’]);
   system([’cp ’,STAGEDIR,’icq5s/’,num2str(yhindcaststart),’_’,...
        num2str(mhindcaststart),’_’,num2str(dhindcaststart),’_hycom_elev.bcs.s2r ’...
        STAGEDIR,’hot/residual.bcs.s2r’]);

   % call the routines that get fc for 1 time to output icq5
   kdi1=datenum(fc_times(1,:));
   kdi2=datenum(fc_times(end,:));
   kdi=datenum(fc_times);
   ncflag=0;  % Nowcast flag
   %hycom2quoddy(FC_URL,kdi1,kdi2,ncflag,nckd)
   hycom2quoddy(FC_URL,kdi,ncflag,nckd)

   On_Log(’Finished building hotstart  ... ’);

   % call routines that get fc . same as above.
   hycflag=1;
   kdi=datenum([nc_times;fc_times]);%kdi(1:dslag-hclag)=[];
   TSeta_extract(kdi,hycflag)

   On_Log(’Finished building bcs ... ’);

otherwise
   On_Err([’Invalid oceanextsrc spec : ’ oceanextsrc{1}])
end

% write CONTROL file for diagnostic run
%    read CONTROL template file
diag_hotstart_name=[’../hot/RESTART.icq5’];
controlfile=textread([MATLABDIR ’/CONTROL.master’],’%s’,’delimiter’,’\n’,’whitespace’,’’);
controlfile=strrep(controlfile,’<gridname>’,quoddy_grid);
controlfile=strrep(controlfile,’<nnv>’,sprintf(’%d’,21));
delt=60;
controlfile=strrep(controlfile,’<delt>’,sprintf(’%d’,delt));
controlfile=strrep(controlfile,’<istride>’,sprintf(’%d’,3600/delt));
controlfile=strrep(controlfile,’<rivfile>’,’NONE’);
controlfile=strrep(controlfile,’<end_dd>’,sprintf(’%d’,dhindcaststart));
controlfile=strrep(controlfile,’<end_mm>’, sprintf(’%d’,mhindcaststart));
controlfile=strrep(controlfile,’<end_yyyy>’, sprintf(’%d’,yhindcaststart));
controlfile=strrep(controlfile,’<hotstart>’,diag_hotstart_name);
controlfile=strrep(controlfile,’<fluxfile>’,’NONE’);
controlfile=strrep(controlfile,’<massvar>’,[’THREE-DIAG’]);
controlfile=strrep(controlfile,’<rhoxy>’,[’RHOXY_FE5’]);
controlfile=strrep(controlfile,’<advscheme>’,[’PCTNQ’]);
%controlfile=strrep(controlfile,’<advscheme>’,[’ADVNQ’]);
controlfile=strrep(controlfile,’<nonlin>’,sprintf(’%d’,0));
%controlfile=strrep(controlfile,’<nbnd>’,sprintf(’%d’,146)); % sab_clim2
controlfile=strrep(controlfile,’<nbnd>’,sprintf(’%d’,147)); % ’seacoos_coarse_sabll_nq’
controlfile=strrep(controlfile,’<ihhbc>’,sprintf(’%d’,3));
controlfile=strrep(controlfile,’<tbcfile>’,’NONE’);
controlfile=strrep(controlfile,’<sbcfile>’,’NONE’);
controlfile=strrep(controlfile,’<reselevfile>’,[’../hot/residual.bcs.s2r’]);
controlfile=strrep(controlfile,’<hycelevfile>’,’NONE’);
controlfile=strrep(controlfile,’<i5file>’,[quoddy_grid,’.i5.nodes’]);
%controlfile=strrep(controlfile,’<tidebcsfile>’,[’sab_clim2_tides.bcs’]);
controlfile=strrep(controlfile,’<tidebcsfile>’,[’seacoos_coarse_sab_nqll.tides.bcs’]);
controlfile=strrep(controlfile,’<nprof>’,sprintf(’%d’,4));
```

```
controlfile=strrep(controlfile,'<profs>',sprintf('%d,%d,%d,%d',11574,10389,8776,6669));

fidc=fopen('CONTROL','w');
for i=1:length(controlfile)
  fprintf(fidc,'%s\n',controlfile{i});
end
fclose(fidc);
% copy diagnostic CONTROL file
system(['cp CONTROL ',STAGEDIR,'diag/.']);

% copy grid files
system(['cp ',GRIDDIR,quoddy_grid,'/',quoddy_grid,'* ',STAGEDIR,'diag/.']);

% copy executable
system(['cp ',EXECLOC,' ',STAGEDIR,'diag/quoddyhycom512']);
system(['cp ',GRIDDIR,quoddy_grid,'/',quoddy_grid,'.lev ',STAGEDIR,'diag/',quoddy_grid,'.lev']);
system(['cp ',COMMONDIR,'submit ',STAGEDIR,'diag/.']);

% write CONTROL file for prognostic run
%    read CONTROL template file
prog_hotstart_name=['../diag/RESTART.icq5'];
controlfile=textread([MATLABDIR '/CONTROL.master'],'%s','delimiter','\n','whitespace','');
controlfile=strrep(controlfile,'<gridname>',quoddy_grid);
delt=30;
controlfile=strrep(controlfile,'<delt>',sprintf('%0.3g',delt));
controlfile=strrep(controlfile,'<istride>',sprintf('%d',3600/delt));
controlfile=strrep(controlfile,'<nnv>',sprintf('%d',21));
controlfile=strrep(controlfile,'<rivfile>','NONE');
controlfile=strrep(controlfile,'<end_dd>',sprintf('%d',dforecastend));
controlfile=strrep(controlfile,'<end_mm>', sprintf('%d',mforecastend));
controlfile=strrep(controlfile,'<end_yyyy>', sprintf('%d',yforecastend));
controlfile=strrep(controlfile,'<hotstart>',prog_hotstart_name);
controlfile=strrep(controlfile,'<fluxfile>',['../hot/flux_fields.22']);
controlfile=strrep(controlfile,'<massvar>',['THREE-PROG']);
controlfile=strrep(controlfile,'<rhoxy>',['RHOXY_FE5']);
controlfile=strrep(controlfile,'<advscheme>',['PCTNQ']);
%controlfile=strrep(controlfile,'<advscheme>',['ADVNQ']);
controlfile=strrep(controlfile,'<nonlin>',sprintf('%d',0));
%controlfile=strrep(controlfile,'<nbnd>',sprintf('%d',146)); % sab_clim2
controlfile=strrep(controlfile,'<nbnd>',sprintf('%d',147)); % 'seacoos_coarse_sabll_nq'
controlfile=strrep(controlfile,'<ihhbc>',sprintf('%d',4));
controlfile=strrep(controlfile,'<tbcfile>',['../hot/Tbound.cbc']);
controlfile=strrep(controlfile,'<sbcfile>',['../hot/Sbound.cbc']);
controlfile=strrep(controlfile,'<reselevfile>','NONE');
controlfile=strrep(controlfile,'<hycelevfile>',['../hot/Zbound.cbc']);
controlfile=strrep(controlfile,'<i5file>',[quoddy_grid,'.i5.nodes']);
%controlfile=strrep(controlfile,'<tidebcsfile>',['sab_clim2_tides.bcs']);
controlfile=strrep(controlfile,'<tidebcsfile>',['seacoos_coarse_sab_nqll.tides.bcs']);
controlfile=strrep(controlfile,'<nprof>',sprintf('%d',4));
controlfile=strrep(controlfile,'<profs>',sprintf('%d,%d,%d,%d',11574,10389,8776,6669));

fidc=fopen('CONTROL','w');
for i=1:length(controlfile)
  fprintf(fidc,'%s\n',controlfile{i});
end
fclose(fidc);
% copy CONTROL file
system(['cp CONTROL ',STAGEDIR,'prog/.']);

% copy grid files
system(['cp ',GRIDDIR,quoddy_grid,'/',quoddy_grid,'* ',STAGEDIR,'prog/.']);
```

```
% copy executable
system(['cp ',EXECLOC,' ',STAGEDIR,'prog/quoddyhycom512']);
system(['cp ',GRIDDIR,quoddy_grid,'/',quoddy_grid,'.lev ',STAGEDIR,'diag/',quoddy_grid,'.lev']);
system(['cp ',COMMONDIR,'submit ',STAGEDIR,'prog/.']);


%%%% write move2nemo file
movefile=textread([MATLABDIR '/move2nemo.master'],'%s','delimiter','\n','whitespace','');
movefile=strrep(movefile,'<date1>',sprintf('%s',caldate(model_hindcast_start_date)));
movefile=strrep(movefile,'<date2>',sprintf('%s',caldate(datenum(nc_times(end,:)))));
movefile=strrep(movefile,'<date3>',sprintf('%s',caldate(model_forecast_end_date)));

fidc=fopen('move2nemo','w');
for i=1:length(movefile)
  fprintf(fidc,'%s\n',movefile{i});
end
fclose(fidc);
system(['chmod 774 move2nemo']);
system(['mv move2nemo ',STAGEDIR,'prog/.']);

%%%%%
%%%%% sort out flux files
%%%%%
% forget about COAMPS for now.
% times will depend on whether this is a hycom run or a clim run.
%make_daily_flux_field_22_awip12(round(model_diag_start_date),timenow-1);
make_daily_flux_field_22_awip12(round(model_hindcast_start_date),timenow-1);
make_daily_flux_field_22_awip12fc(timenow);

system(['cat flux_fields.0*.22 fc_flux_fields.0*.22 > flux_temp']);
system(['mv flux_temp ',STAGEDIR,'hot/flux_fields.22']);
system(['rm flux_fields.0*.22 fc_flux_fields.0*.22']);

% copy to happy e.g.
%scp ../stage/hot/* happy.isis.unc.edu:/netscr/alfredo/hycom_oper_copy/oper_110805/stage/hot/.

%%%%%
%%%%% clean up, wrap up, and write final logs
%%%%%

system(['rm *.mat']);
```

9.3. hycom_quoddy_setup.m

```
% operational definitions

% for a given date, determine the products available from the %4d%02d%02d
% best-estimate, nowcast, or forecast.
% check for best estimate
% old urls
%BE_URL='http://hycom.rsmas.miami.edu/cgi-bin/nph-dods/datasets/ocean-prediction/NAT/best-estimate/';
%NC_URL='http://hycom.rsmas.miami.edu/cgi-bin/nph-dods/datasets/ocean-prediction/NAT/nowcast/';

%THREDDS Aggregate server
%NC_URL='http://asterix.rsmas.miami.edu/thredds/dodsC/atl-ops-nowcast/';
%FC_URL='http://asterix.rsmas.miami.edu/thredds/dodsC/atl-ops-forecast/';

% "Snapshot" server
HYCOM_HOST='asterix.rsmas.miami.edu';
%NC_URL=['http://' HYCOM_HOST '/datasets/ocean-prediction/atlantic/nowcast/'];
```

```
%FC_URL=['http://' HYCOM_HOST '/datasets/ocean-prediction/atlantic/forecast/'];
NC_URL=['/Users/hycom/nowcast/'];
FC_URL=['/Users/hycom/forecast/'];


HOMEDIR= '/Users/alfredo/hycom_oper_copy/oper_nq_20060405/';
STAGEDIR=[HOMEDIR 'stage/'];
MATLABDIR=[HOMEDIR 'matlab/'];
CLIMDIR ='/afs/isis/home/h/s/hseim/public_html/sablam/climatology/baroclinic_sol/icq5s/';
%GRIDDIR ='/afs/isis/depts/marine/opnml/ARCHIVE/GRIDS/';
GRIDDIR ='/Users/alfredo/hycom_oper_copy/';
%EXECLOC ='/afs/isis/depts/marine/workspace/bblanton/HYCOM/quoddy_src_rcs/quoddyhycom512';
EXECLOC ='/Users/alfredo/hycom_oper_copy/src/quoddyhycom512';
COMMONDIR='/Users/alfredo/hycom_oper_copy/input_files/';
%AWIPDIR ='/dissert/NCEP/awip1200z';
AWIPDIR ='/Volumes/NCEP/awip1200z/';

% define some vars
oceanextsrc={'HYCOM','CLIM'};    % the order in which sources will be searched for initial/boundary data
metextsrc={'NAM','COAMPS'};      % the order in which sources will be searched for surf flux data

% days prior to "today" for hindcast run
hclag=8;
dslag=12;
fcastlength=4;
quoddy_grid='seacoos_coarse_sab_nq';
```

9.4. hycom2quoddy.m

```
function hycom2quoddy(URL,kdi,ncflag,nckd)
%function hycom2quoddy(URL,kdi1,kdi2,ncflag,nckd)
% extract Temperature Salinity and Elevation BC's
% from hycom fields

hycom_quoddy_setup;
load_constants;

g=loadgrid([GRIDDIR,quoddy_grid,'/',quoddy_grid,'ll']);
[ncyyyy,ncmm,ncdd]=datevec(nckd);

for ij=1:length(kdi)
% ij=1
  [yyyy,mm,dd]=datevec(kdi(ij));

  disp('HYCOM retrieval: Starting Stage 1 ...')
  fprintf('Getting HYCOM solution from: %s\n',URL);

  if ncflag
    uname=sprintf('%s%s/hycom_2.1_nat_1o12ml_%s_%4d%02d%02d.nc',...
      URL,'ssh','ssh',yyyy,mm,dd);
    mldpthnm=sprintf('%s%s/hycom_2.1_nat_1o12ml_%s_%4d%02d%02d.nc',...
      URL,'mldpth','mldpth',yyyy,mm,dd);
    ubaronm=sprintf('%s%s/hycom_2.1_nat_1o12ml_%s_%4d%02d%02d.nc',...
      URL,'ubaro','ubaro',yyyy,mm,dd);
    vbaronm=sprintf('%s%s/hycom_2.1_nat_1o12ml_%s_%4d%02d%02d.nc',...
      URL,'vbaro','vbaro',yyyy,mm,dd);
    uvelnm=sprintf('%s%s/hycom_2.1_nat_1o12ml_%s_%4d%02d%02d.nc',...
      URL,'uvel','uvel',yyyy,mm,dd);
    vvelnm=sprintf('%s%s/hycom_2.1_nat_1o12ml_%s_%4d%02d%02d.nc',...
      URL,'vvel','vvel',yyyy,mm,dd);
```

```
  tempnm=sprintf(’%s%s/hycom_2.1_nat_1o12ml_%s_%4d%02d%02d.nc’,...
    URL,’temp’,’temp’,yyyy,mm,dd);
  saltnm=sprintf(’%s%s/hycom_2.1_nat_1o12ml_%s_%4d%02d%02d.nc’,...
    URL,’salt’,’salt’,yyyy,mm,dd);
  lthknm=sprintf(’%s%s/hycom_2.1_nat_1o12ml_%s_%4d%02d%02d.nc’,...
    URL,’lthk’,’lthk’,yyyy,mm,dd);
else
  uname=sprintf(’%s/%s/hycom_2.1_nat_1o12ml_%s_b%4d%02d%02d_f%4d%02d%02d.nc’,...
    URL,’ssh’,’ssh’,ncyyyy,ncmm,ncdd,yyyy,mm,dd);
  mldpthnm=sprintf(’%s/%s/hycom_2.1_nat_1o12ml_%s_b%4d%02d%02d_f%4d%02d%02d.nc’,...
    URL,’mldpth’,’mldpth’,ncyyyy,ncmm,ncdd,yyyy,mm,dd);
  ubaronm=sprintf(’%s/%s/hycom_2.1_nat_1o12ml_%s_b%4d%02d%02d_f%4d%02d%02d.nc’,...
    URL,’ubaro’,’ubaro’,ncyyyy,ncmm,ncdd,yyyy,mm,dd);
  vbaronm=sprintf(’%s/%s/hycom_2.1_nat_1o12ml_%s_b%4d%02d%02d_f%4d%02d%02d.nc’,...
    URL,’vbaro’,’vbaro’,ncyyyy,ncmm,ncdd,yyyy,mm,dd);
  uvelnm=sprintf(’%s/%s/hycom_2.1_nat_1o12ml_%s_b%4d%02d%02d_f%4d%02d%02d.nc’,...
    URL,’uvel’,’uvel’,ncyyyy,ncmm,ncdd,yyyy,mm,dd);
  vvelnm=sprintf(’%s/%s/hycom_2.1_nat_1o12ml_%s_b%4d%02d%02d_f%4d%02d%02d.nc’,...
    URL,’vvel’,’vvel’,ncyyyy,ncmm,ncdd,yyyy,mm,dd);
  tempnm=sprintf(’%s/%s/hycom_2.1_nat_1o12ml_%s_b%4d%02d%02d_f%4d%02d%02d.nc’,...
    URL,’temp’,’temp’,ncyyyy,ncmm,ncdd,yyyy,mm,dd);
  saltnm=sprintf(’%s/%s/hycom_2.1_nat_1o12ml_%s_b%4d%02d%02d_f%4d%02d%02d.nc’,...
    URL,’salt’,’salt’,ncyyyy,ncmm,ncdd,yyyy,mm,dd);
  lthknm=sprintf(’%s/%s/hycom_2.1_nat_1o12ml_%s_b%4d%02d%02d_f%4d%02d%02d.nc’,...
    URL,’lthk’,’lthk’,ncyyyy,ncmm,ncdd,yyyy,mm,dd);
end

%  /afs/isis.unc.edu/depts/marine/opnml/matlab/local/hycom/get_hycom_output_reba.pl

nc_ssh=netcdf(uname);
query_cdf(uname)

Longitude=nc_ssh{’Longitude’}(:);
Latitude=nc_ssh{’Latitude’}(:);
sub_grid_def;

Longitude=Longitude(str2num(lon_range));
Latitude=Latitude(str2num(lat_range));

 % get grids
 % 2D
 fprintf(’\n\nGetting 2D grids ... ’);
 fprintf(’\nssh ... ’);
 ssh=nc_ssh{’ssh’}(:);ssh=ssh(str2num(lat_range),str2num(lon_range));
 %ssh_var  =[uname ’?ssh’ lat_range lon_range];
 %loaddods(ssh_var);
 ssh(find(ssh>1e10))=NaN;

 fprintf(’\nmldpth ... ’);

 fprintf(’\nubaro ... ’);
 nc_ubaro=netcdf(ubaronm);
 ubaro=nc_ubaro{’ubaro’}(:);ubaro=ubaro(str2num(lat_range),str2num(lon_range));
 ubaro(find(ubaro>1e10))=NaN;

 fprintf(’\nvbaro ... ’);
 nc_vbaro=netcdf(vbaronm);
 vbaro=nc_vbaro{’vbaro’}(:);vbaro=vbaro(str2num(lat_range),str2num(lon_range));
 vbaro(find(vbaro>1e10))=NaN;

 % 3D
 fprintf(’\n\nGetting 3D grids ... ’);
```

```
fprintf('\nuvel ... ');

nc_uvel=netcdf(uvelnm);
uvel=nc_uvel{'uvel'}(:);uvel=uvel(str2num(dep_range)+1,str2num(lat_range),str2num(lon_range));
uvel(find(uvel>1e10))=NaN; pack

fprintf('\nvvel ... ');
nc_vvel=netcdf(vvelnm);
vvel=nc_vvel{'vvel'}(:);vvel=vvel(str2num(dep_range)+1,str2num(lat_range),str2num(lon_range));
vvel(find(vvel>1e10))=NaN; pack

fprintf('\ntemp ... ');
nc_temp=netcdf(tempnm);
temp=nc_temp{'temp'}(:);temp=temp(str2num(dep_range)+1,str2num(lat_range),str2num(lon_range));
temp(find(temp>1e10))=NaN; pack

fprintf('\nsalt ... ');
nc_salt=netcdf(saltnm);
salt=nc_salt{'salt'}(:);salt=salt(str2num(dep_range)+1,str2num(lat_range),str2num(lon_range));
salt(find(salt>1e10))=NaN; pack

fprintf('\nlthk ... ');
nc_lthk=netcdf(lthknm);
lthk=nc_lthk{'lthk'}(:);lthk=lthk(str2num(dep_range)+1,str2num(lat_range),str2num(lon_range));
lthk(find(lthk>1e10))=NaN; pack

%clear nc_ssh nc_tmp
save matlab_stage1

disp('HYCOM retrieval: Starting Stage 2 ...')

 % layer depths
clear tmp
tmp(:,:,1)=ssh;
tmp(:,:,2:27)=shiftdim(lthk,1);
layer_depths=cumsum(tmp,3);

% hycom "z" coords
clear tmp
tmp(:,:,1)=ssh;
tmp(:,:,2:27)=-shiftdim(lthk,1);
hycom_z=cumsum(tmp,3);

izero=find(isnan(lthk));
lthk(izero)=0;
clear tmp

S=NaN*ones(141, 131, 27);
S(:,:,1)=shiftdim(salt(1,:,:),1);
S(:,:,2:end)=shiftdim(salt,1);

T=NaN*ones(141, 131, 27);
T(:,:,1)=shiftdim(temp(1,:,:),1);
T(:,:,2:end)=shiftdim(temp,1);

U3D=NaN*ones(141, 131, 27);
U3D(:,:,1)=shiftdim(uvel(1,:,:),1);
U3D(:,:,2:end)=shiftdim(uvel,1);

V3D=NaN*ones(141, 131, 27);
V3D(:,:,1)=shiftdim(vvel(1,:,:),1);
V3D(:,:,2:end)=shiftdim(vvel,1);
```

```
U2D=ubaro;
V2D=vbaro;

pack
% sum of layer thicknesses gives total bathymetric depth
depth=squeeze(sum(shiftdim(lthk,1),3));

% generate a FEM grid of the HYCOM grid, for interpolation purposes.
disp('Generating fake HYCOM FEM grid ...');
hc.e=elgen(length(Longitude),length(Latitude));
hc.x=repmat(Longitude,[1 length(Latitude)])';
hc.y=repmat(Latitude,[1 length(Longitude)]);
hc.bnd=detbndy(hc.e);
hc.name='Fake FEM grid for HYCOM Interp';
hc.z=depth(:);
hc.x=hc.x(:);
hc.y=hc.y(:);
hc=belint(hc);
hc=el_areas(hc);

% find FEM nodes in fake HYCOM FEM grid
if ij==1
   j=findelem(hc,[g.x g.y]);
end

% this plot gives back the SSH!!
%clf;pcolor(Longitude,Latitude,layer_depths(:,:,end)-depth);axeq;shading interp;colorbar

% interp SSH to FEM
g.ssh=interp_scalar(hc,ssh(:),g.x,g.y,j);

% find nearest fake HYCOM nodes (with defined SSH) to NaN's in g.ssh
disp('Finding nearest HYCOM nodes ...')
% inan is the index of nodes in g that will need fixing.
% these nodes are not in the "defined" HYCOM FEM grid
inan=find(isnan(g.ssh));

% issh is the index of HYCOM FEM nodes that have defined
% field values.
issh=find(isfinite(ssh(:)));

% inear contains the indices into the fake HYCOM FEM grid of the
% nearest defined nodes to the indefined quoddy nodes in inan
for i=1:length(inan)
   dx=(g.x(inan(i))-hc.x(issh)).^2;
   dy=(g.y(inan(i))-hc.y(issh)).^2;
   [a,b]=min(dx+dy);
   inear(i)=issh(b);
end

% this replaces NaNs in g.ssh with finite values from the
% fake HYCOM grid nearest to the inan nodes
if length(inan)~0
  g.ssh(inan)=ssh(inear);
end
%clf;colormesh2d(g,g.ssh);axeq;shading interp;colorbar

% now, loop over horizontal quoddy nodes to interpolate to
nnv=21;
dzbl=1.0;
g.Tm=NaN*ones(g.nn,nnv);
```

```
 g.S=g.Tm;
 g.U=g.Tm;
 g.V=g.Tm;

disp('Looping over Horizontal Quoddy Nodes ...')
for i=1:g.nn

 if rem(i,100)==0,disp([ int2str(i) '/' int2str(g.nn)]),end

 % if this is a "defined" quoddy location (i.e., inside the defined
 % but fake YCOM FEM grid) then proceed

 if ~any(i==inan)

    % Interpolate Horizontally to construct HYCOM vertical profiles
    jdx=j(i);
    x=g.x(i);
    y=g.y(i);

    % construct sigma grid at this location
    zquoddy=sinegrid(g.z(i),g.ssh(i),nnv,dzbl)';

    % get corresponding HYCOM vertical structure
    % we are in HYCOM element j(i)
    [i1,j1]=ind2sub([141 131],hc.e(jdx,1));
    [i2,j2]=ind2sub([141 131],hc.e(jdx,2));
    [i3,j3]=ind2sub([141 131],hc.e(jdx,3));

    z1=squeeze(hycom_z(i1,j1,:));
    z2=squeeze(hycom_z(i2,j2,:));
    z3=squeeze(hycom_z(i3,j3,:));

    s1=squeeze(S(i1,j1,:));
    s2=squeeze(S(i2,j2,:));
    s3=squeeze(S(i3,j3,:));

    t1=squeeze(T(i1,j1,:));
    t2=squeeze(T(i2,j2,:));
    t3=squeeze(T(i3,j3,:));

    u1=squeeze(U3D(i1,j1,:));
    u2=squeeze(U3D(i2,j2,:));
    u3=squeeze(U3D(i3,j3,:));

    v1=squeeze(V3D(i1,j1,:));
    v2=squeeze(V3D(i2,j2,:));
    v3=squeeze(V3D(i3,j3,:));

    U1=squeeze(U2D(i1,j1));
    U2=squeeze(U2D(i2,j2));
    U3=squeeze(U2D(i3,j3));

    V1=squeeze(V2D(i1,j1));
    V2=squeeze(V2D(i2,j2));
    V3=squeeze(V2D(i3,j3));

    % Add in barotropic velocity
%     u1=u1+U1;u2=u2+U2;u3=u3+U3;
%     v1=v1+V1;v2=v2+V2;v3=v3+V3;

    ARI=.5./hc.ar(jdx);
    A03 = hc.ar(jdx)-hc.A0(jdx,1) - hc.A0(jdx,2);
```

```
zhc=NaN*ones(27,1);
thc=NaN*ones(27,1);
shc=NaN*ones(27,1);
uhc=NaN*ones(27,1);
vhc=NaN*ones(27,1);

% vertical grid
for jj=1:27
   zz=[z1(jj) z2(jj) z3(jj)];
   if all(isfinite(zz))
      e123=interp_fxn(hc,zz,jdx,ARI,A03);
      zhc(jj) = sum([e123(1).*x, e123(2).*y, e123(3)]);
   elseif sum(isfinite(zz))==2
      idx=find(isfinite(zz));
      zz1=zz(idx(1));
      zz2=zz(idx(2));
      zhc(jj) = mean([zz1 zz2]);
   else
      idx=find(isfinite(zz));
      zhc(jj) = zz(idx);
   end          %  if all(isfinite(zz))
end             %  end for jj=1:27

%TS
for jj=1:27
   tt=[t1(jj) t2(jj) t3(jj)];
   ss=[s1(jj) s2(jj) s3(jj)];
   if all(isfinite(tt))
      e123=interp_fxn(hc,tt,jdx,ARI,A03);
      thc(jj) = sum([e123(1).*x, e123(2).*y, e123(3)]);
      e123=interp_fxn(hc,ss,jdx,ARI,A03);
      shc(jj) = sum([e123(1).*x, e123(2).*y, e123(3)]);
   elseif sum(isfinite(tt))==2
      idx=find(isfinite(tt));
      thc(jj) = mean([tt(idx(1)) tt(idx(2))]);
      shc(jj) = mean([ss(idx(1)) ss(idx(2))]);
   else
      idx=find(isfinite(tt));
      thc(jj) = tt(idx);
      shc(jj) = ss(idx);
   end          %  if all(isfinite(tt))
end             %  end for jj=1:27

% U3D
for jj=1:27
   uu=[u1(jj) u2(jj) u3(jj)];
   if all(isfinite(uu))
      e123=interp_fxn(hc,uu,jdx,ARI,A03);
      uhc(jj) = sum([e123(1).*x, e123(2).*y, e123(3)]);
   elseif sum(isfinite(uu))==2
      idx=find(isfinite(uu));
      uhc(jj) = mean([uu(idx(1)) uu(idx(2))]);
   else
      idx=find(isfinite(uu));
      uhc(jj) = uu(idx);
   end          %  if all(isfinite(uu))
end             %  end for jj=1:27

% V3D
for jj=1:27
   vv=[v1(jj) v2(jj) v3(jj)];
```

```
         if all(isfinite(vv))
            e123=interp_fxn(hc,vv,jdx,ARI,A03);
            vhc(jj) = sum([e123(1).*x, e123(2).*y, e123(3)]);
         elseif sum(isfinite(vv))==2
            idx=find(isfinite(vv));
            vhc(jj) = mean([vv(idx(1)) vv(idx(2))]);
         else
            idx=find(isfinite(vv));
            vhc(jj) = vv(idx);
         end          %  if all(isfinite(vv))
      end             %  end for jj=1:27

      % interpolate vertically from HC vert grid to QUODDY vert
      % grid (zz)
      for jj=nnv:-1:1
         [b2,n2,b1,n1]=basis1d1(-zhc,-zquoddy(jj));
         tquoddy(jj)=thc(n1)*b1+thc(n2)*b2;
         squoddy(jj)=shc(n1)*b1+shc(n2)*b2;
         uquoddy(jj)=uhc(n1)*b1+uhc(n2)*b2;
         vquoddy(jj)=vhc(n1)*b1+vhc(n2)*b2;
      end            % for jj=nnv:-1:1

%      if any(isnan(tquoddy))
%      end            %  if any(isnan(tquoddy))
      g.Tm(i,:)=tquoddy;
      g.S(i,:)=squoddy;
      g.U(i,:)=uquoddy;
      g.V(i,:)=vquoddy;
   else
   end                %  end  if ~any(i==inan)

  end                   % end for i=1:g.nn loop

  % check for nans in the "defined" portion of the quoddy grid
  ifinite=setdiff(1:g.nn,inan);
  t=g.Tm(ifinite,:)';

  if ~isempty(find(any(isnan(t))))
     disp('There are NaNs in the defined part of the grid.');
  end

  % now, fill the undefined grid with nearest defined nodes.
  disp('Filling undefined regions ... ')

  for i=1:length(inan)
     dx=(g.x(inan(i))-g.x(ifinite)).^2;
     dy=(g.y(inan(i))-g.y(ifinite)).^2;
     [a,b]=min(dx+dy);
     inear2(i)=ifinite(b);
  end

  for i=1:length(inan)
     ii=inan(i);
     zsrc=sinegrid(g.z(inear2(i)),g.ssh(inear2(i)),nnv,dzbl)';
     tsrc=g.Tm(inear2(i),:);
     ssrc=g.S(inear2(i),:);
     usrc=g.U(inear2(i),:);
     vsrc=g.V(inear2(i),:);
     zdst=sinegrid(g.z(ii),g.ssh(ii),nnv,dzbl)';
     for jj=nnv:-1:1
           [b2,n2,b1,n1]=basis1d1(zsrc,zdst(jj));
           tdst(jj)=tsrc(n1)*b1+tsrc(n2)*b2;
```

```
         sdst(jj)=ssrc(n1)*b1+ssrc(n2)*b2;
         udst(jj)=usrc(n1)*b1+usrc(n2)*b2;
         vdst(jj)=vsrc(n1)*b1+vsrc(n2)*b2;
   end            % for jj=nnv:-1:1
        g.Tm(ii,:)=tsrc;
        g.S(ii,:)=ssrc;
        g.U(ii,:)=usrc;
        g.V(ii,:)=vsrc;
end

t=g.Tm';
if ~isempty(find(any(isnan(t))))
   disp('There are NaNs in the defined part of the grid.');
end

% interp U2D,V2D
[g.u2d,jjj]=interp_scalar(hc,U2D(:),g.x,g.y);
inan=find(isnan(g.u2d));
ifinite=find(isfinite(g.u2d));
for i=1:length(inan)
   dx=(g.x(inan(i))-g.x(ifinite)).^2;
   dy=(g.y(inan(i))-g.y(ifinite)).^2;
   [a,b]=min(dx+dy);
   g.u2d(inan(i))=g.u2d(ifinite(b));
end

g.v2d=interp_scalar(hc,V2D(:),g.x,g.y,jjj);
inan=find(isnan(g.v2d));
ifinite=find(isfinite(g.v2d));
for i=1:length(inan)
   dx=(g.x(inan(i))-g.x(ifinite)).^2;
   dy=(g.y(inan(i))-g.y(ifinite)).^2;
   [a,b]=min(dx+dy);
   g.v2d(inan(i))=g.v2d(ifinite(b));
end

save matlab_stage2

pack

disp('HYCOM retrieval: Starting Stage 3 ...')
%  load matlab_stage2

%i5=load('sab_ec95ll.i5.nodes');
%i5=load('lam_coarse2ll.i5.nodes');
i5name=[GRIDDIR,quoddy_grid,'/',quoddy_grid,'.i5.nodes'];
i5=load(i5name);
xi5=g.x(i5);
yi5=g.y(i5);

zi5=interp2(Longitude,Latitude,ssh,xi5,yi5);
inan=find(isnan(zi5(1:round(length(zi5)/2))));
if length(inan)~0;zi5(inan)=zi5(inan(end)+1);end
l=length(zi5);
li=round(length(zi5)/2);
inan=find(isnan(zi5(l-li:l)));
if length(inan)~0;zi5(l-li+inan-1)=zi5(l-li+inan(1)-2);end
out=[i5 zi5 zeros(size(zi5))];

out=[i5 zi5];
s2rname=[STAGEDIR,'icq5s/',num2str(yyyy),'_',num2str(mm),'_',num2str(dd),'_hycom_elev.bcs.s2r']
%fid=fopen('hycom_elev.bcs.s2r','w');
```

```
  fid=fopen(s2rname,'w');
  fprintf(fid,'%s\n%s\n',quoddy_grid,'test hycom bcs');
  fprintf(fid,'%d %f\n',out');
  fclose(fid);

  % build icq5 for hotstart file

  %startdate for tides rampup is (yyyy,mm,dd)- days
  [syyyy,smm,sdd,shh,smn,sss]=datevec(datenum(yyyy,mm,dd));

  icq5.codename='QUODDY5';
  icq5.casename=quoddy_grid;
  icq5.inqfilename='todays.inq';
  icq5.initcondname='HYCOM-HOT';
  icq5.nn=g.nn;
  icq5.nnv=nnv;
  icq5.day=sdd;
  icq5.month=smm;
  icq5.year=syyyy;
  icq5.curr_seconds=0.;
  icq5.step_seconds=60.;

  icq5.ZETAMID=g.ssh;
  icq5.UMID=g.u2d;
  icq5.VMID=g.v2d;
  icq5.ZMID=sinegrid(g.z,icq5.ZETAMID,nnv,dzbl);

  icq5.ZETAOLD=icq5.ZETAMID;
  icq5.UOLD=icq5.UMID;
  icq5.VOLD=icq5.VMID;
  icq5.ZOLD=icq5.ZMID;

  icq5.UZMID=g.U;
  icq5.VZMID=g.V;
  icq5.WZMID=zeros(size(g.U));

  icq5.TMPMID=g.Tm;
  icq5.SALMID=g.S;
  icq5.Q2MID=0.000001*ones(size(icq5.TMPMID));
  icq5.Q2LMID=0.000001*ones(size(icq5.TMPMID));

  icq5name=[STAGEDIR,'icq5s/',num2str(syyyy),'_',num2str(smm),'_',num2str(sdd),'_HC.HOT.icq5']
  %err=write_icq5(icq5,'HC.HOT.icq5');
  err=write_icq5(icq5,icq5name);
  %system('bzip2 -f  HC.HOT.icq5');

  save matlab_stage3

end

return
```

References

Blanton, B., I. Bang, and R. He, Implementation of the SEACOOS Nowcast/Forecast Model System. Version 1.0 Barotropic Models and Skill, *Tech. rep.*, SEACOOS MPCC, pp 21., 2004a.

Blanton, B. O., F. E. Werner, H. Seim, R. A. Luettich, D. R. Lynch, K. W. Smith, G. Voulgaris, F. M. Bingham, and F. Way, Barotropic tides in the South Atlantic Bight, *J. Geophys. Res.*, *109*(C12024), doi:10.1029/2004JC002455, 2004b.

Bleck, R., Ocean modeling in isopycnic coordinates, in *Ocean Modeling and Parameterization*, edited by E. P. Chassignet and J. Verron, pp. 423–448, Kluwer Academic Publishers, 1998.

Chassignet, E. P., L. T. Smith, G. R. Halliwell, and R. Bleck, North Atlantic simulations with the Hybrid Coordinate Ocean Model (HYCOM): Impact of the vertical coordinate choice, reference pressure, and thermobaricity, *J. Phys. Oceanogr.*, *33*(12), 2504–2526, 2001.

Kliem, N., A transport corrected finite element advection scheme, *Ocean Modelling*, *7*(1-2), 1–19, 2004.

Lynch, D. R., and F. E. Werner, Three-dimensional hydrodynamics on finite elements. Part ii: Non-linear time-stepping model, *Int. J. Numer. Methods Fluids*, *12*, 507–533, 1991.

Lynch, D. R., J. T. C. Ip, C. E. Naimie, and F. E. Werner, Comprehensive coastal circulation model with application to the Gulf of Maine, *Cont. Shelf Res.*, *16*, 875–906, 1996.

Mellor, G., and T. Yamada, Development of a turbulence closure model for geophysical fluid problems., *Rev. of Geophys. Space Phys.*, *20*, 851–875, 1982.

Seim, H., et al., Sea-coos - a model for a multi-state, multi-institutional regional observation system., *MTS Journal*, *37*, 92–101, 2003.

Received ______________________

This manuscript was prepared with AGU's LaTeX macros v5, with the extension package 'AGU^{++}' by P. W. Daly, version 1.6b from 1999/08/19.

Figure Captions

Figure 1. Weekly operational timeline.

Figure 2. Operational procedure flowchart.

Figure 3. Model domains. The red mesh corresponds to the coarser, quadratic elements. The blue mesh represents the denser, lower order mesh. One quadratic element covers exactly four linear elements.

Figure 4. Location of the 4 available transects and position of the 4 profile locations: NDBC buoy 41008, SABSOON towers R2 and R6, and a point in open ocean over the Blake Plateau.

Figure 5. Surface temperature and depth-averaged velocity for 11-Apr-2006. This is the final time of the simulation for 29-Mar-2006.

Figure 6. Bottom temperature at R2 tower. The blue line represents climatological time series. The red line is the observations during 2006. The green dashed line is the temperature predicted by HYCOM. The black line is the temperature predicted by the current NC/FC system using QUODDY.

Figure 7. Surface temperature at Gray's Reef buoy (41008). The blue line represents climatological time series. The red line is the observations during 2006. The green dashed line is the temperature predicted by HYCOM. The black line is the temperature predicted by the current NC/FC system using QUODDY.

Figure 8. Timeline for the different components of the simulation under normal conditions.

Figure 9. Timeline for the different components of the simulation under long delay in posting times.

Tables

Run	Initial time	Final time	HYCOM posting time	Quoddy posting time
Mar 08	Feb 27	Mar 22	Mar 22	Mar 23
Mar 15	Mar 09	Mar 29	Mar 23	Mar 24
Mar 22	NO	NO	NO	NO
Mar 29	Mar 23	Apr 11	Apr 04 06:23	Apr 05 13:29
Apr 05	Mar 27	Apr 19	Apr 12 12:48	Apr 14 11:00
Apr 12	NO	NO	NO	NO

Table 1. Availability of output products for last month.

Figures

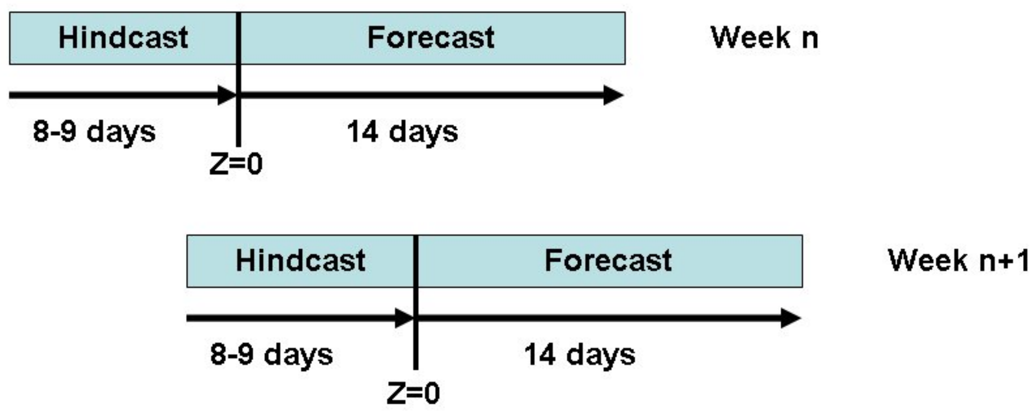

Figure 1. Weekly operational timeline.

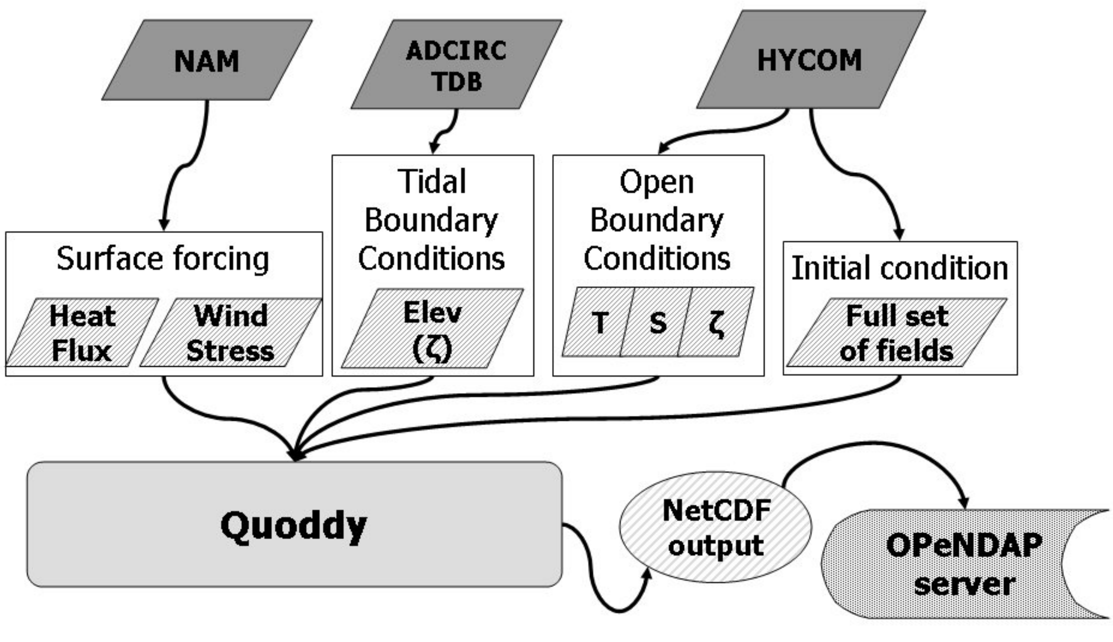

Figure 2. Operational procedure flowchart.

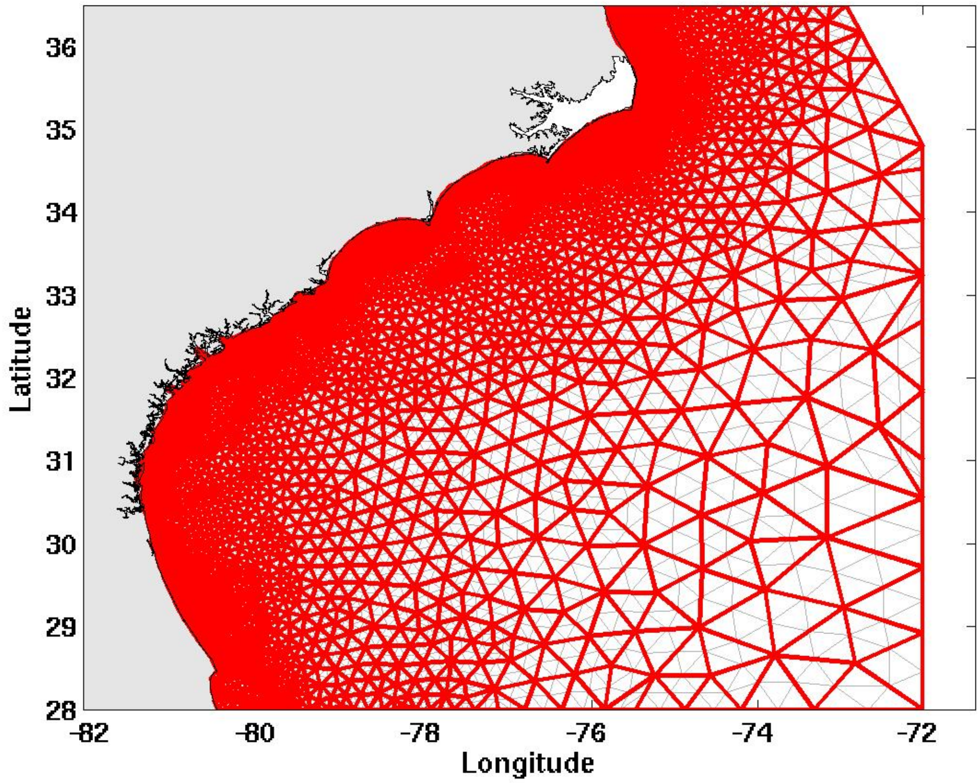

Figure 3. Model domains. The red mesh corresponds to the coarser, quadratic elements. The blue mesh represents the denser, lower order mesh. One quadratic element covers exactly four linear elements.

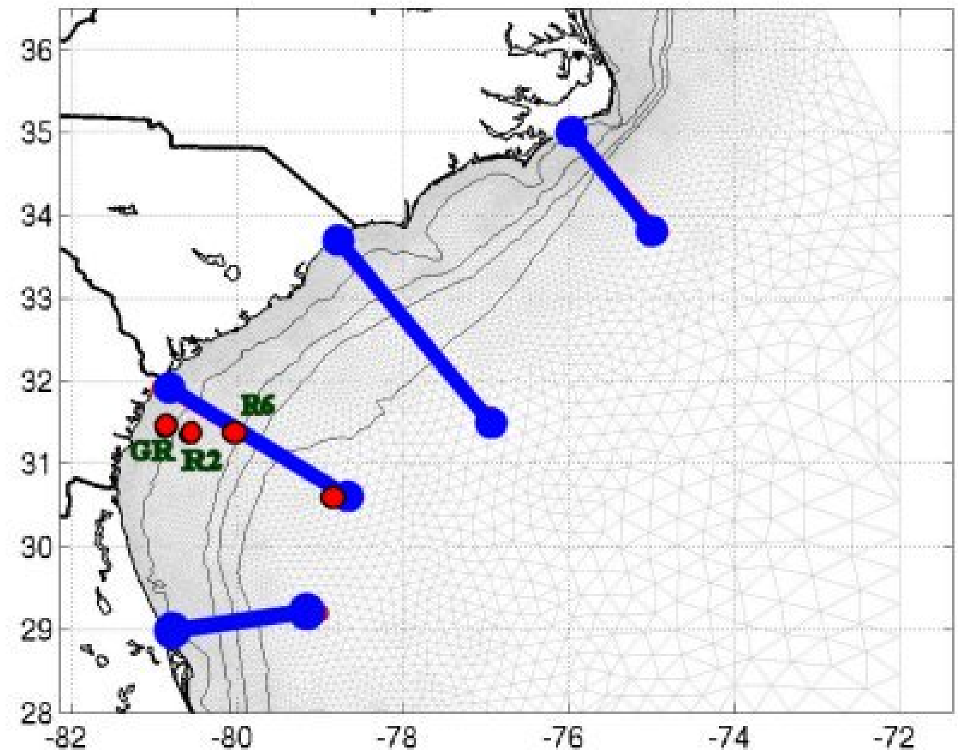

Figure 4. Location of the 4 available transects and position of the 4 profile locations: NDBC buoy 41008, SABSOON towers R2 and R6, and a point in open ocean over the Blake Plateau.

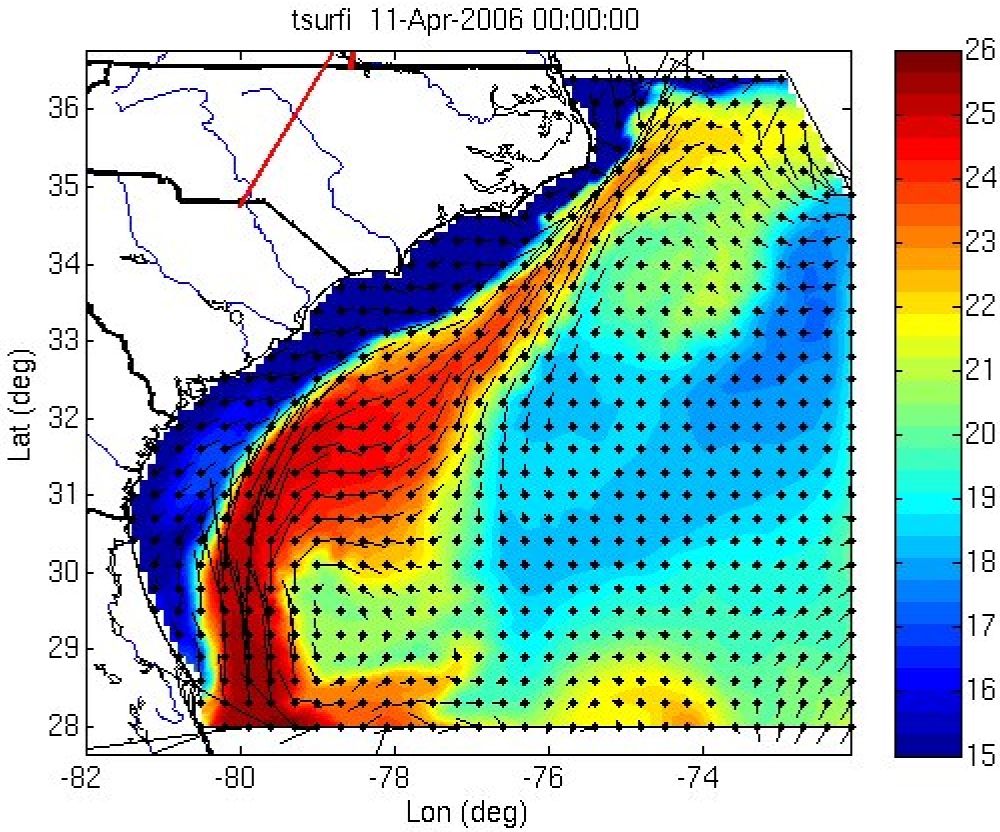

Figure 5. Surface temperature and depth-averaged velocity for 11-Apr-2006. This is the final time of the simulation for 29-Mar-2006.

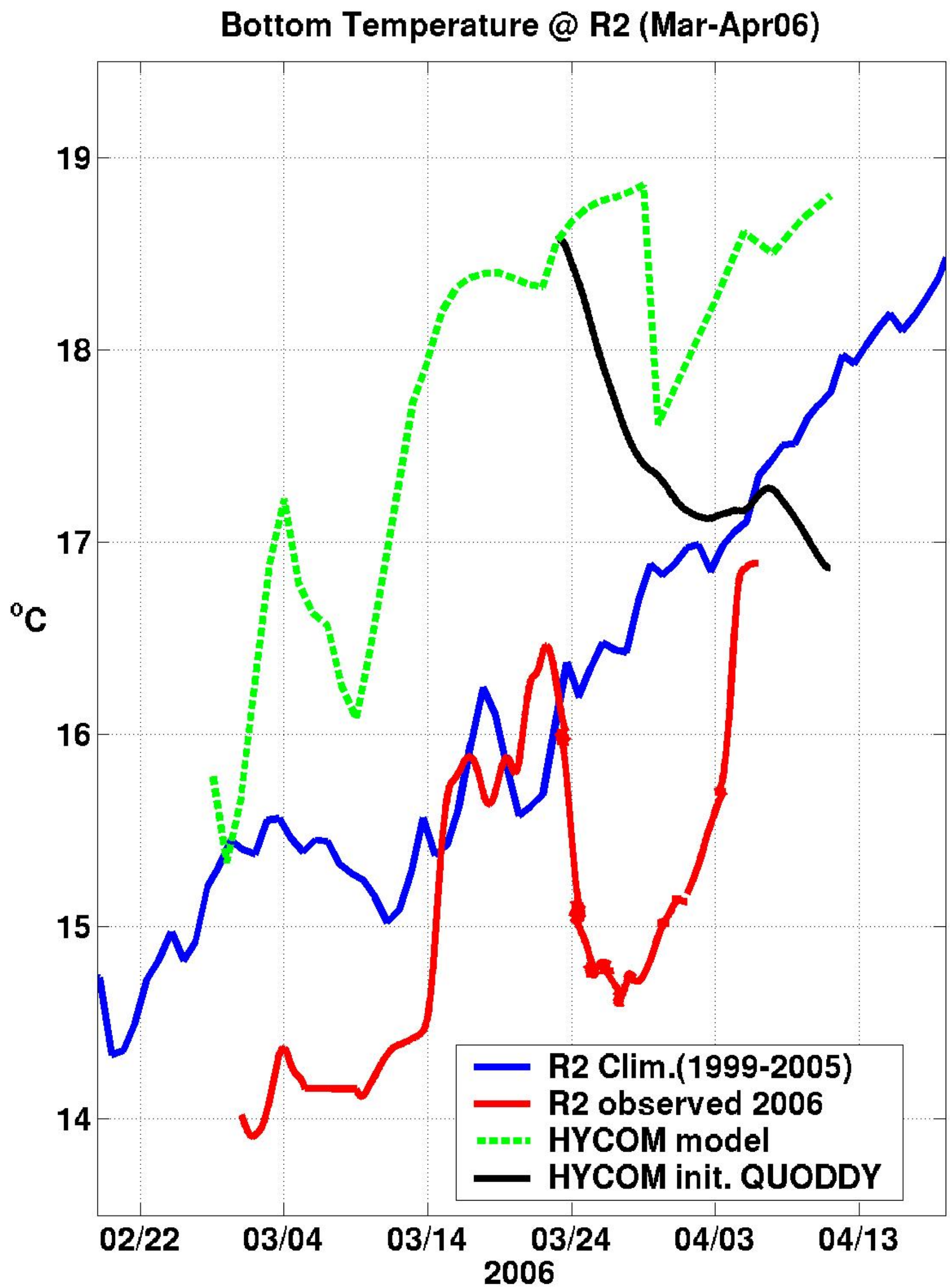

Figure 6. Bottom temperature at R2 tower. The blue line represents climatological time series. The red line is the observations during 2006. The green dashed line is the temperature predicted by HYCOM. The black line is the temperature predicted by the current NC/FC system using QUODDY.

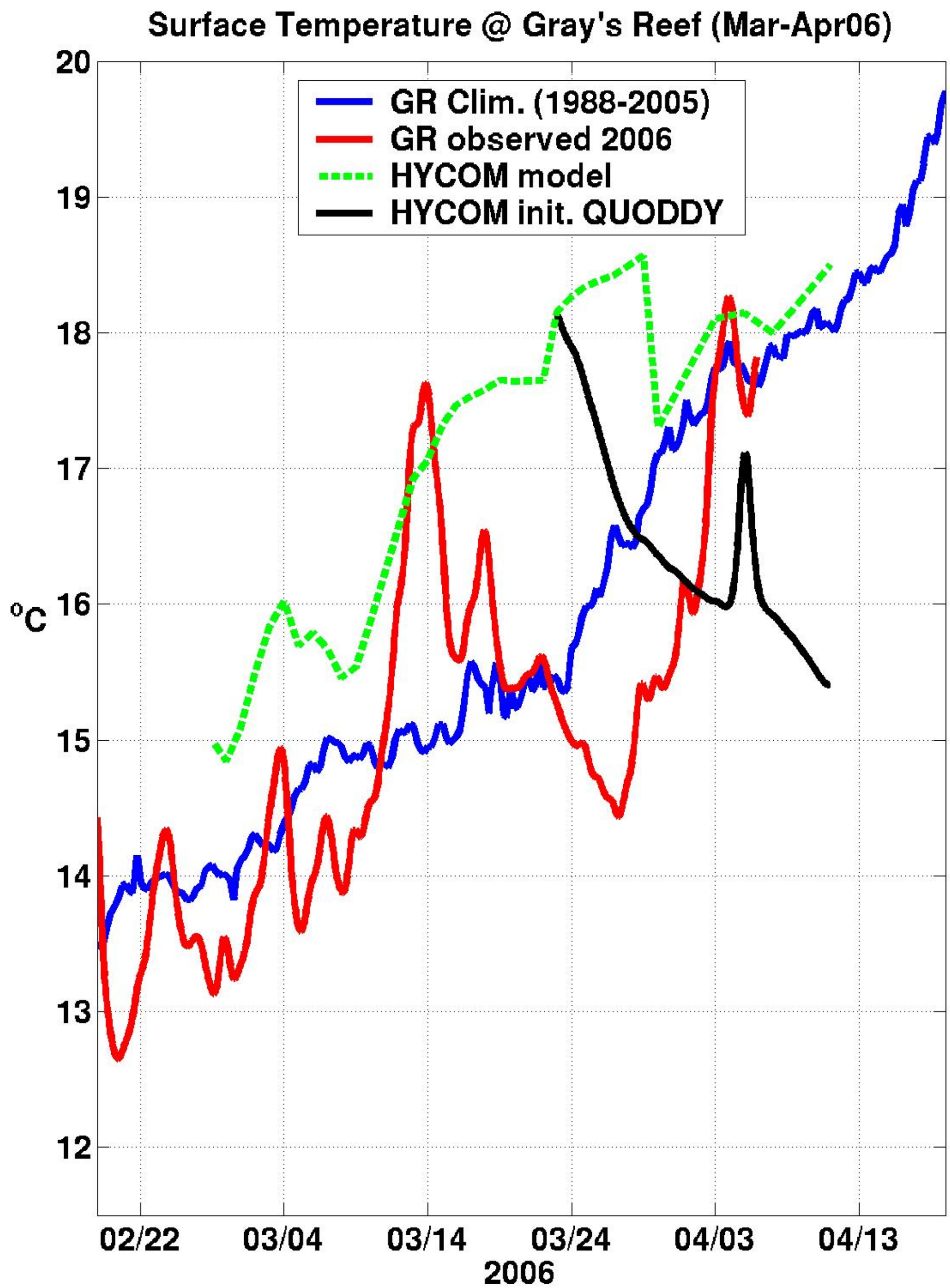

Figure 7. Surface temperature at Gray's Reef buoy (41008). The blue line represents climatological time series. The red line is the observations during 2006. The green dashed line is the temperature predicted by HYCOM. The black line is the temperature predicted by the current NC/FC system using QUODDY.

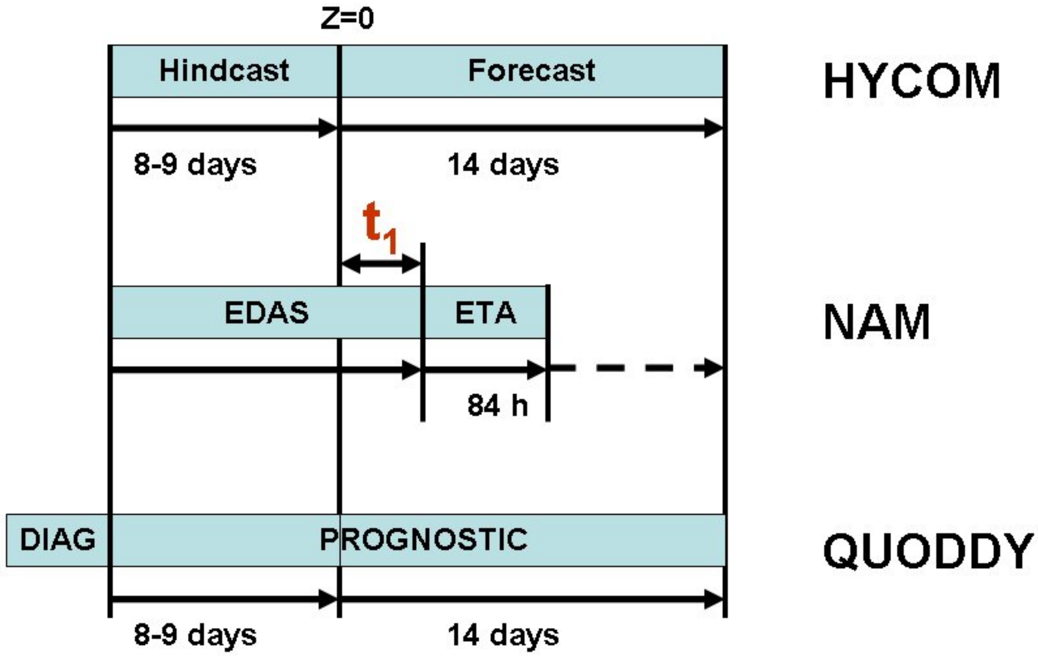

Figure 8. Timeline for the different components of the simulation under normal conditions.

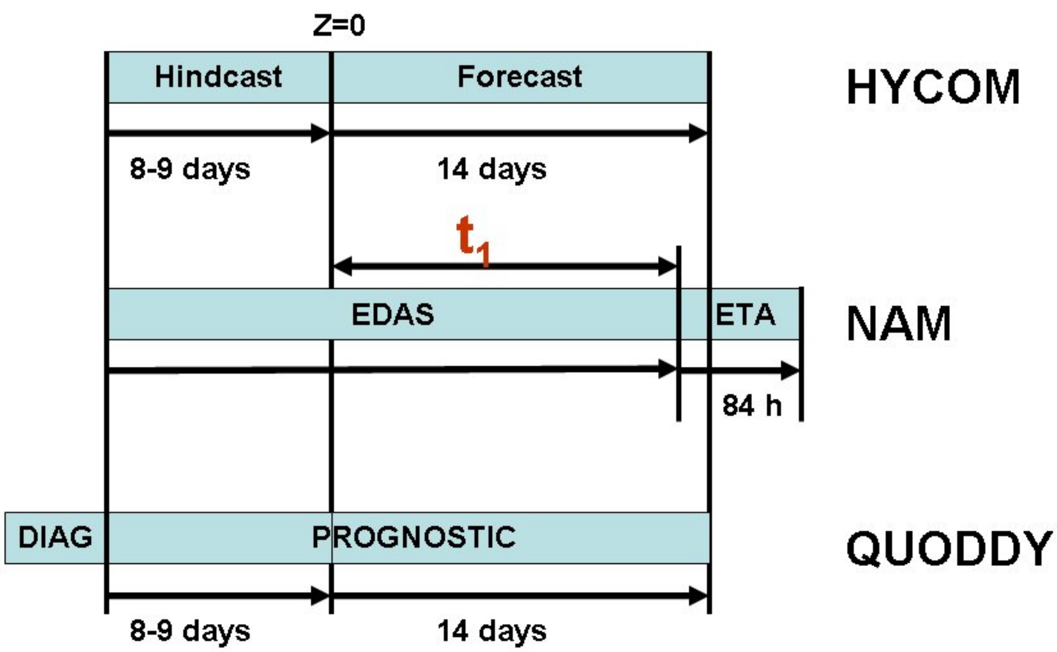

Figure 9. Timeline for the different components of the simulation under long delay in posting times.