The MASNUM SurfaceWave Model

Zhenya Song

Key Laboratory of Marine Science and Numerical Modelling (MASNUM), First Institute of Oceanography (FIO), SOA, Qingdao China

Laboratory of Regional Oceanography and Numerical Modeling, Qingdao National Laboratory for Marine Science and Technology, Qingdao China

January 26, 2016, Beijing

Outline

- **Surface Wave Model**

- Background

- Future prospects

- **MASNUM surface wave model**

- Brief introduction

- Steps to run

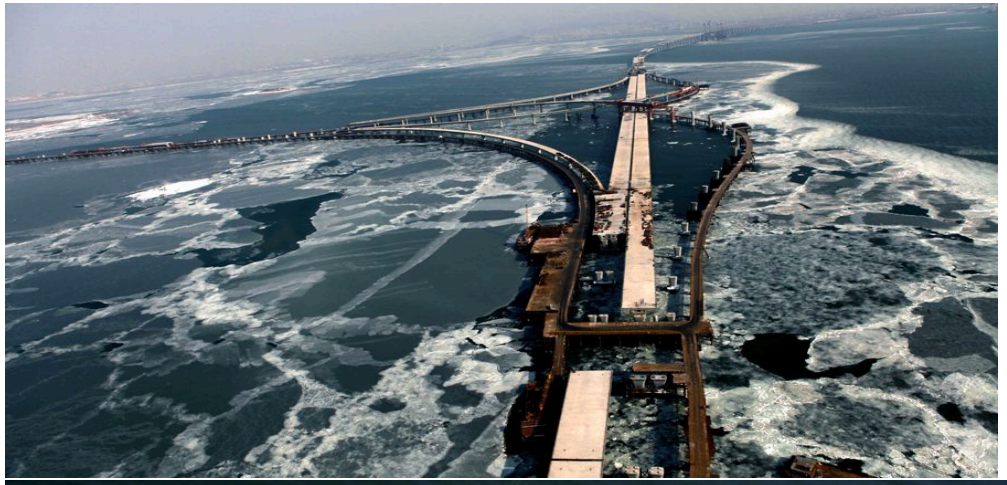
- **Suggestions**

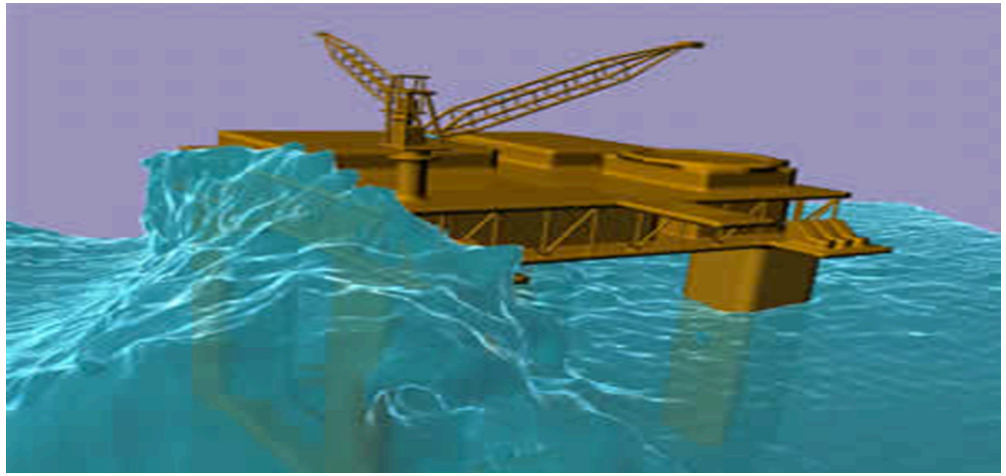
Surface Wave Model

What is surface wave

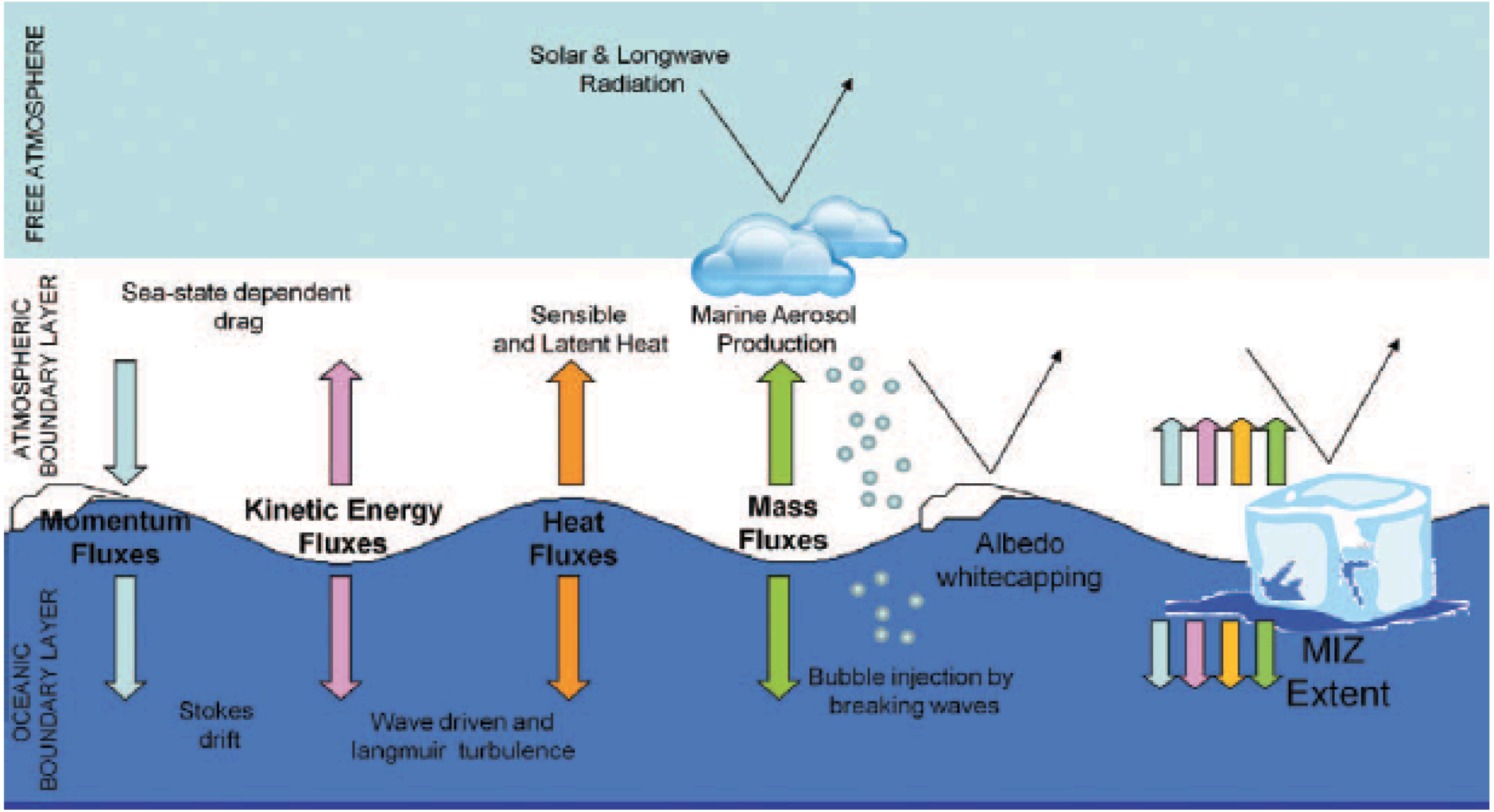
How important to study

How important to study





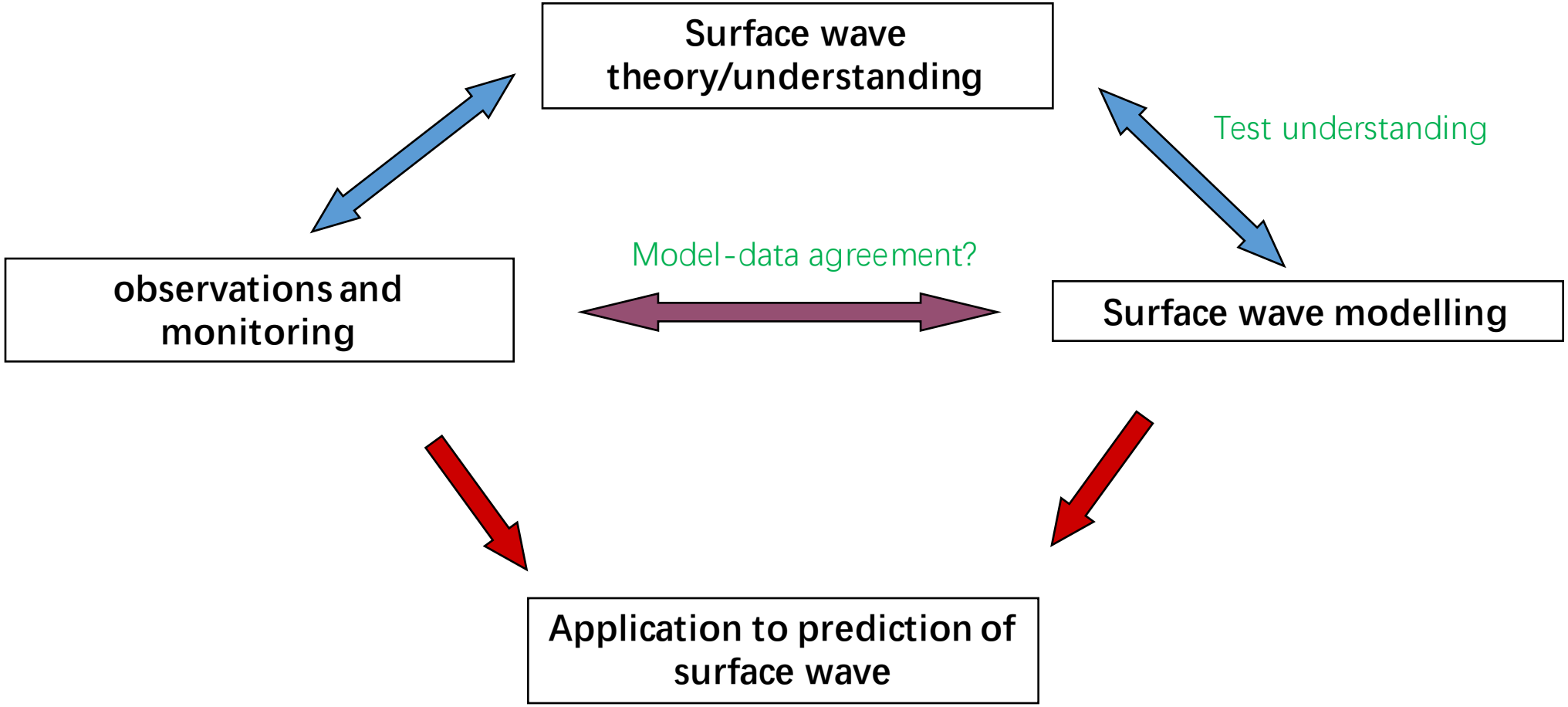
How important to study



Methods for surface wave study

- Theoretical research
- Physical simulation
- Observation
- Numerical simulation

Numerical Simulation--Model



Understanding and Prediction

Model

- Scientific model
- Wiki definition:

  A scientific model is a simplified abstract view of a complex reality. A scientific model represents empirical objects, phenomena, and physical processes in a logical way. Attempts to formalize the principles of the empirical sciences use an interpretation to model reality, in the same way logicians axiomatize the principles of logic. The aim of these attempts is to construct a formal system for which reality is the only interpretation. The world is an interpretation (or model) of these sciences, only insofar as these sciences are true.

Statistical vs Physical Models

- **Statistical Model**

- Based on observations, you identify a relationship between two variables. You do not necessarily understand the reason why this relationship exists.

- **Physical Model**

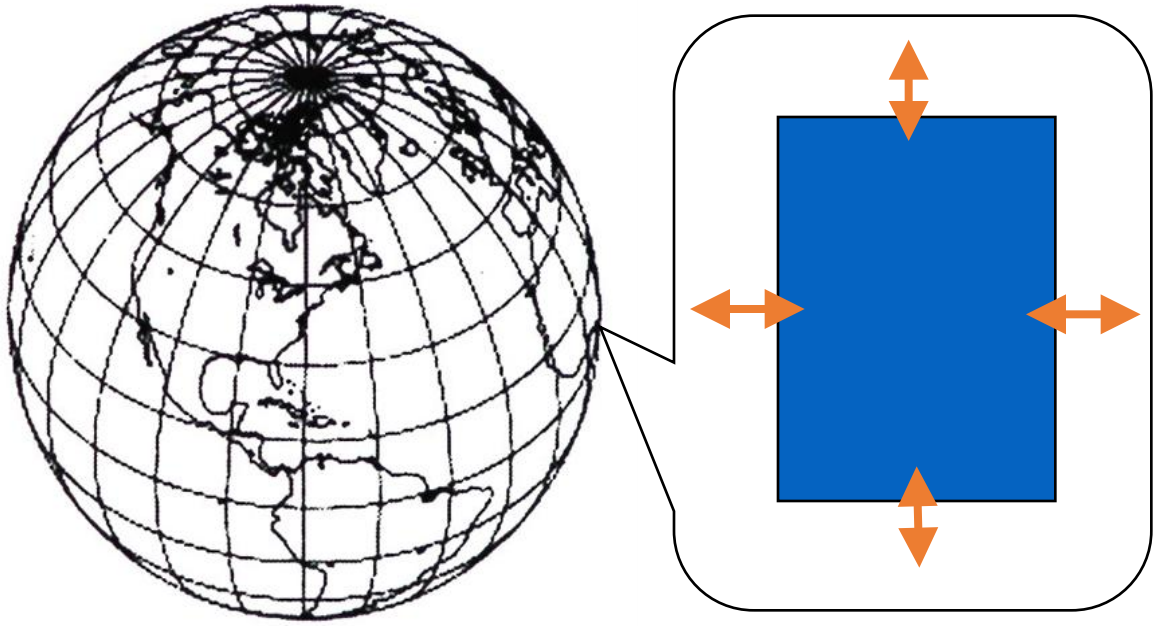
- Based on the rules of physics, you construct a model that describes the relationships between different physical phenomena.

What is a surface wave model?

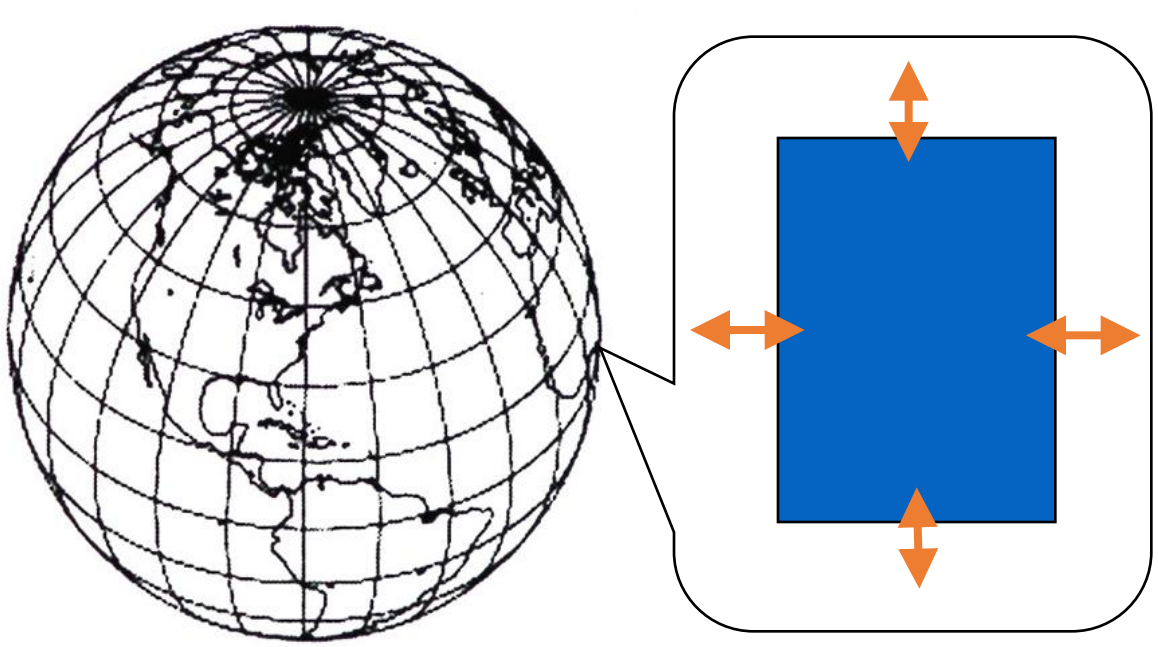
- **A mathematical representation of the many processes that make up the surface wave**
- **Requires**

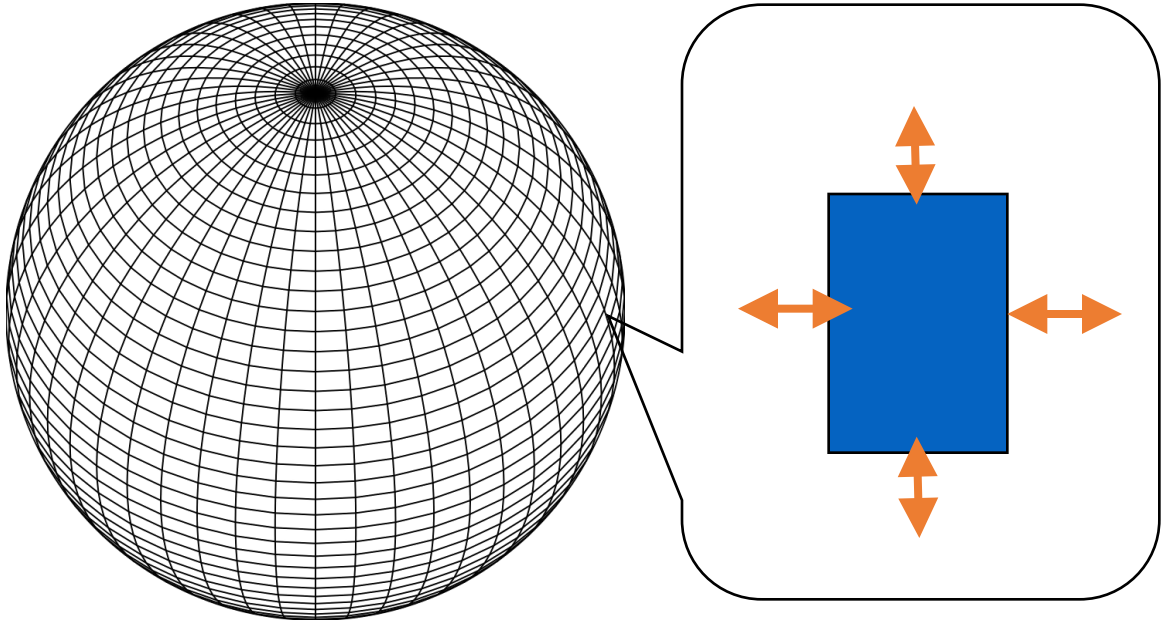
- Knowledge of the physical laws that govern surface wave
- Mathematical expressions for those laws
- Numerical methods to solve the mathematical expressions on a computer
- A computer of adequate size to carry out the calculations

Resolution



Resolution





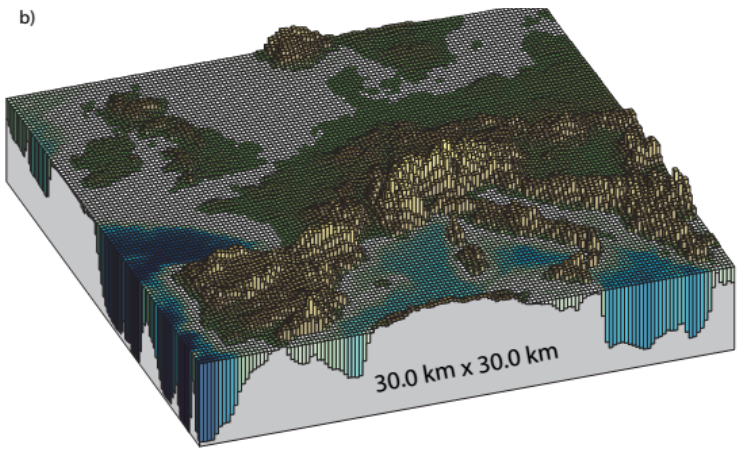
Parameterization

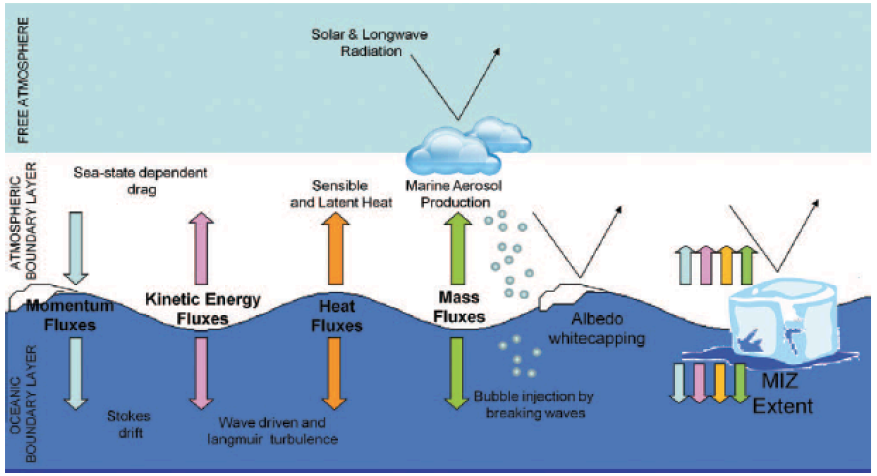
- **Important processes smaller than a grid box**
- **We need to include them by means of parameterization (a largely statistical description of what goes on "inside" the box)**
- **What is a model to do?**

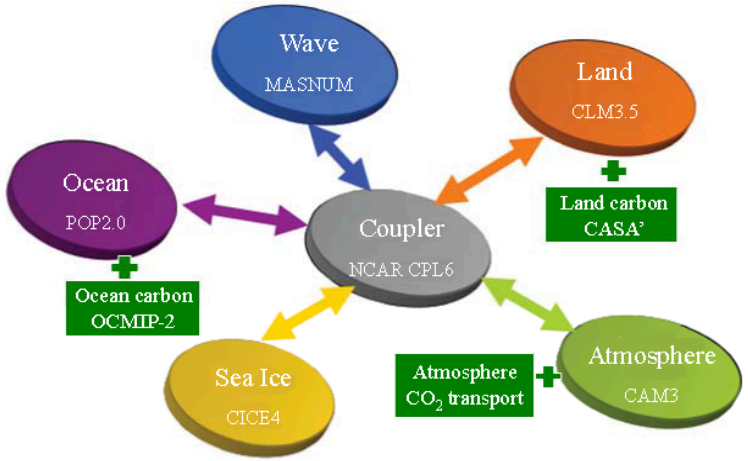
- Parameterization: Represent the effects of the unresolved processes on the grid. Assume that unresolved processes are at least partly driven by the resolved climate.
- Similar idea to molecules being summarized statistically by temperature and pressure, but much more complex!

Future prospects

- Higher resolution
- More physical process
- Coupled models







MASNUM
Surface Wave Model

MASNUM surface wave model

- Developed by Key Laboratory of Marine Science and Numerical Modelling (MASNUM)
- Once called LAGFD (Laboratory of Geophysical Fluid Dynamics) wave model
- third-generation surface wave model

Model	Version	Period	Coordinate	Scale	Assimilation	FORTRAN
LAGFD -WAM	1.1	1990s	Orthogonal	Regional	No	F77
	1.2	2000	Orthogonal	Regional	Yes	F77
MASNUM-WAM	2.1	2005 -2009	Spherical	Regional /Global	No	F77
	2.2	2009 - now	Spherical	Regional /Global	No	F90 Modularized

MASNUM surface wave model

Basic equations:

$$\frac{\partial E}{\partial t}+\left(\frac{C_{g\lambda}+U_{\lambda}}{R\cos\phi}\right)\frac{\partial E}{\partial \lambda}+\left(\frac{C_{g\phi}+U_{\phi}}{R}\right)\frac{\partial E}{\partial \phi}-\frac{(C_{g\phi}+U_{\phi})\tan\phi}{R}E=SS(E)$$

Where $E=E(K,\lambda,\phi,t)$ is the wave-number spectrum as a function of the vector for wave-number $\vec{K}=(k_{\lambda},\mathrm{k}_{\phi})$, latitude ϕ and longitude λ; $\vec{U}=(U_{\lambda},U_{\phi})$ is background current velocity; and $\vec{C}_g=(C_{g\lambda},C_{g\phi})$ represents the group velocity. The right term of *SS(E)* consists of the wind input source function，the wave breaking dissipation function，the bottom friction dissipation function, the wave-wave weak nonlinear interaction function and the wave-current interaction. Detailed descriptions of these source functions can be found in the following section.

MASNUM surface wave model

Basic equations:

$$\frac{\partial E}{\partial t}+\left(\frac{C_{g\lambda}+U_{\lambda}}{R\cos\phi}\right)\frac{\partial E}{\partial \lambda}+\left(\frac{C_{g\phi}+U_{\phi}}{R}\right)\frac{\partial E}{\partial \phi}-\frac{(C_{g\phi}+U_{\phi})\tan\phi}{R}E=SS(E)$$

In spherical coordinates:

$$\frac{d\lambda}{dt}=\frac{C_{g\lambda}+U_{\lambda}}{R\cos\phi}\qquad \frac{d\phi}{dt}=\frac{C_{g\phi}+U_{\phi}}{R}$$

$$\frac{\partial K}{\partial t}+\frac{C_{g\lambda}+U_{\lambda}}{R\cos\phi}\frac{\partial K}{\partial \lambda}+\frac{C_{g\phi}+U_{\phi}}{R}\frac{\partial K}{\partial \phi}+\left(U_{\lambda}\sin\theta_1-U_{\phi}\cos\theta_1\right)\tan\phi R^{-1}K\cos\theta_1$$
$$=-\frac{\cos\theta_1}{R\cos\phi}\left(\frac{\partial\sigma}{\partial D}\frac{\partial D}{\partial \lambda}+K\cos\theta_1\frac{\partial U_{\lambda}}{\partial \lambda}+K\sin\theta_1\frac{\partial U_{\phi}}{\partial \lambda}\right)$$
$$-\frac{\sin\theta_1}{R}\left(\frac{\partial\sigma}{\partial D}\frac{\partial D}{\partial \phi}+K\cos\theta_1\frac{\partial U_{\lambda}}{\partial \phi}+K\sin\theta_1\frac{\partial U_{\phi}}{\partial \phi}\right)$$

$$\frac{\partial \theta_1}{\partial t}+\frac{C_{g\lambda}+U_{\lambda}}{R\cos\phi}\frac{\partial \theta_1}{\partial \lambda}+\frac{C_{g\phi}+U_{\phi}}{R}\frac{\partial \theta_1}{\partial \phi}+\left(U_{\lambda}\cos\theta_1+U_{\phi}\sin\theta_1\right)\tan\phi R^{-1}\cos\theta_1$$
$$+C_g\tan\phi R^{-1}\cos\theta_1$$
$$=\frac{\sin\theta_1}{R\cos\phi}\left(\frac{1}{K}\frac{\partial\sigma}{\partial D}\frac{\partial D}{\partial \lambda}+\cos\theta_1\frac{\partial U_{\lambda}}{\partial \lambda}+\sin\theta_1\frac{\partial U_{\phi}}{\partial \lambda}\right)$$
$$-\frac{\cos\theta_1}{R}\left(\frac{1}{K}\frac{\partial\sigma}{\partial D}\frac{\partial D}{\partial \phi}+\cos\theta_1\frac{\partial U_{\lambda}}{\partial \phi}+\sin\theta_1\frac{\partial U_{\phi}}{\partial \phi}\right)$$

MASNUM surface wave model

Basic equations:

$$\frac{\partial E}{\partial t}+\left(\frac{C_{g\lambda}+U_{\lambda}}{R\cos\phi}\right)\frac{\partial E}{\partial \lambda}+\left(\frac{C_{g\phi}+U_{\phi}}{R}\right)\frac{\partial E}{\partial \phi}-\frac{(C_{g\phi}+U_{\phi})\tan\phi}{R}E=SS(E)$$

Source functions:

$$SS=S_{in}+S_{ds}+S_{bo}+S_{nl}+S_{cu}$$

S_{in}: Wind input source function

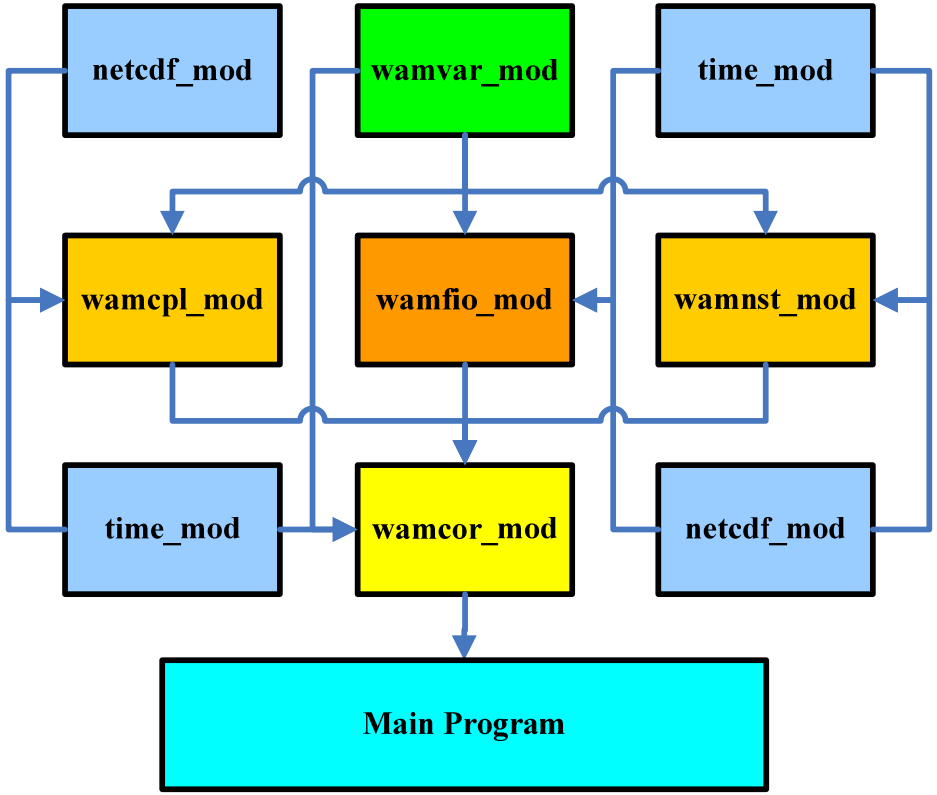
S_{ds}: The dissipation source function

S_{bo}: The bottom dissipation source function

S_{nl}: The nonlinear wave-wave interaction source function

S_{cu}: The wave-current interaction source function

MASNUM surface wave model



The dependence between different modules

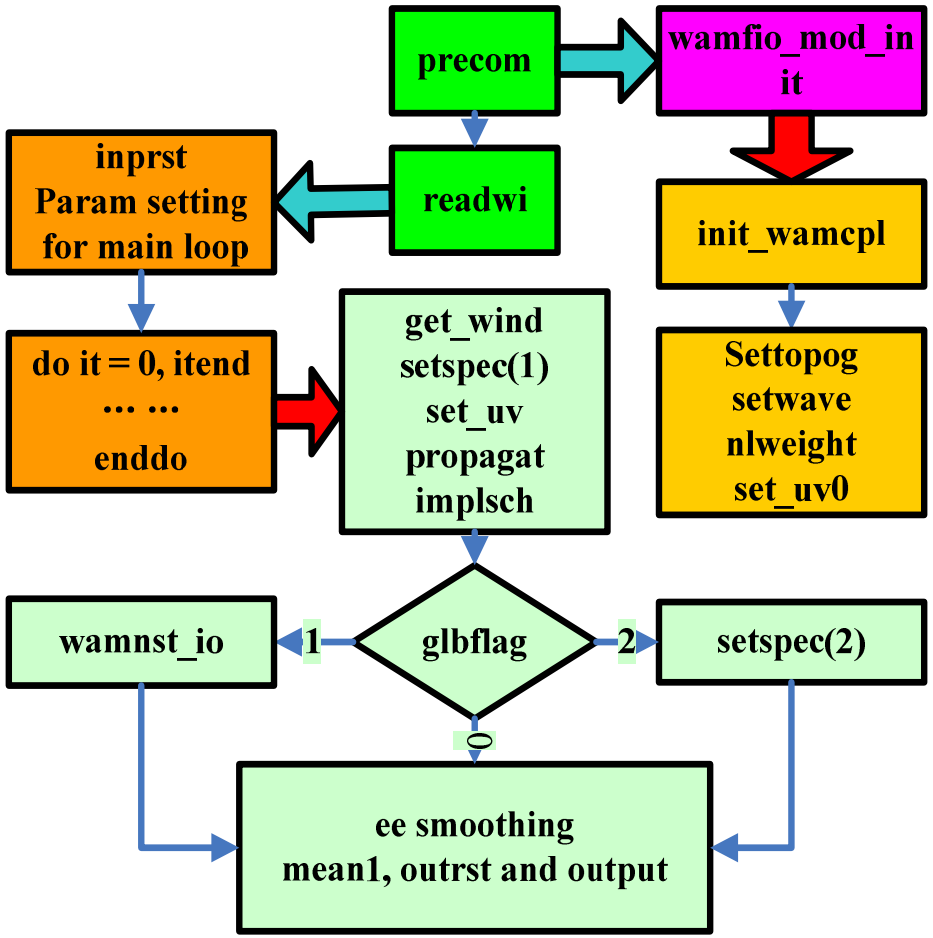
(1)	**time_mod**	Used to deal with the time.
(2)	**netcdf_mod**	Used to input/output data through NetCDF format.
(3)	**wamvar_mod**	Include all the global variables used in this model.
(4)	**wamfio_mod**	Subroutines for I/O data or model results.
(5)	**wamcpl_mod**	Subroutines for coupling w/current model.
(6)	**wamcor_mod**	The core subroutines of this model.
(7)	**wamnst_mod**	The subroutines for model nesting.

Independent modules: *netcdf_mod, time_mod*

MASNUM surface wave model

```
main program_____precom_____settopog
              |          |__setwave
              |          |__nlweight_____jafu
              |__readwi_____setspec
                         |__propagat_____inter
                         |__implsch _____mean2
                         |....... glb / setspec / nest
                         |__mean1
                         |__output ...
```

The structure of the MASNUM surface wave model



The flowchart of the main program

How can we setup the MASNUM surface wave model experiments?

- Setup the computation environment
- Modify necessary parameters in “makefile”
- Compile
- Modify necessary parameters in exp1_run.csh and exp2_run.csh
- Run
- Validate the results

Setup the computation environment

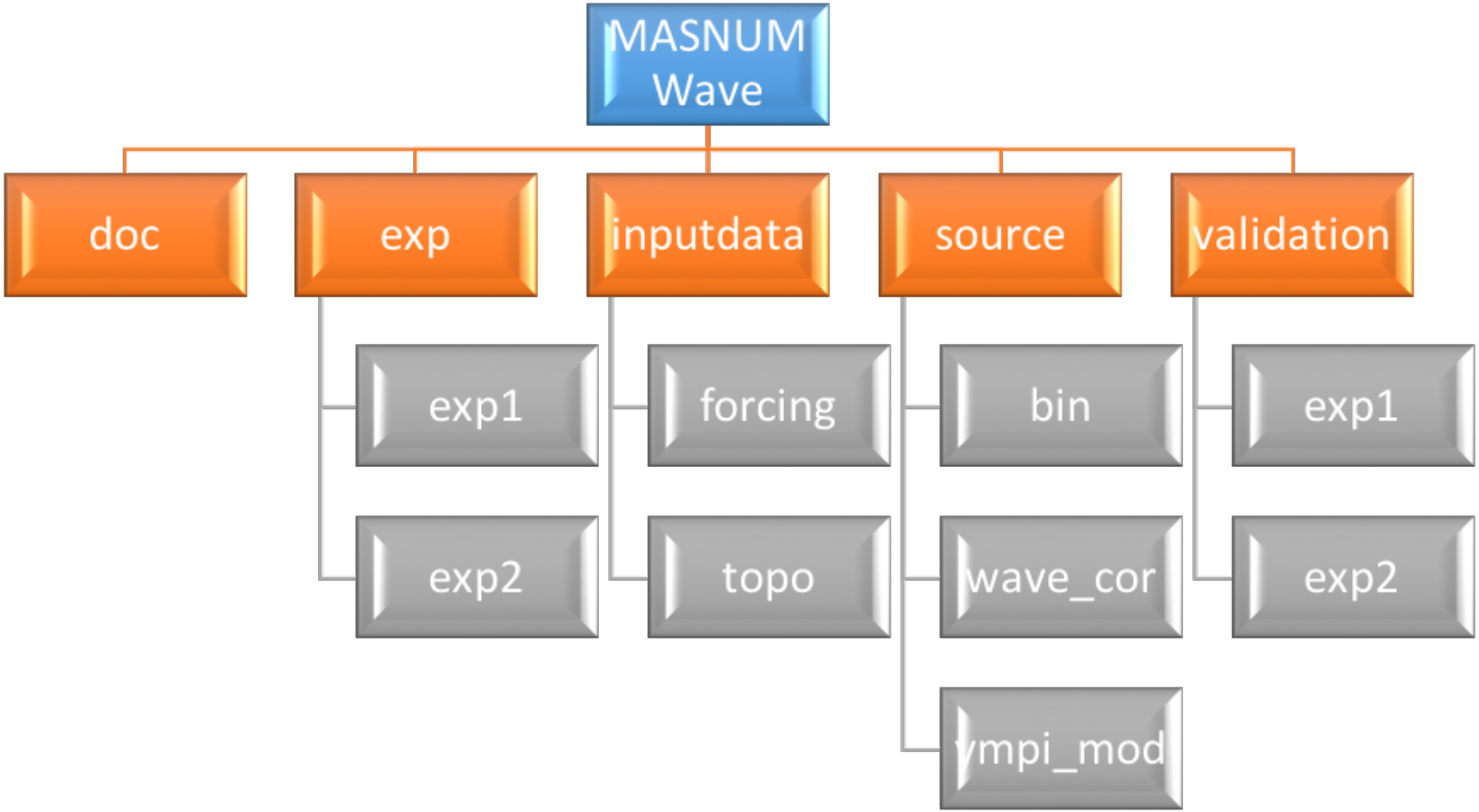
- UNIX/Linux/OSX operation system
- Scripting languages: csh
- Compilers: pgi/intel/xlf, support Fortran 90
- MPI: intel mpi/openmpi/mpich
- Other library: NetCDF (version 3.6.2/3.6.3)

Setup the computation environment

NetCDF version 3.6.3

- **Download:**
- wget ftp://ftp.unidata.ucar.edu/pub/netcdf/netcdf-3.6.3.tar.gz
- **Install**
- tar -zxvf netcdf-3.6.3.tar.gz
- cd netcdf-3.6.3
- FC=ifort CC=icc CXX=icpc ./configure –prefix=$HOME/tools/netcdf_3.6.3
- make
- make check all
- make install

Structure of MASNUM surface wave model for ASC



Compiling procedures

- Enter the directory masnum_wave/source/bin
- Modify the necessary parameters in the “makefile”

Compi

- Enter the directory masr
- Modify the necessary pa

```
F77 = mpif90
LF77OPTS =
FFLAGS   =

EXE = masnum.wam.mpi

NETCDF_PATH = /WORK/home/fio_climate1/tools/netcdf/netcdf-3.6.2

LIBS = -L$(NETCDF_PATH)/lib -lnetcdf -L/usr/local/lib

INC_PATH = -I$(NETCDF_PATH)/include

SRC = \
netcdf_mod.f90 time_mod.f90  wamvar_mod.f90 wamcpl_mod.f90 \
wamfio_mod.f90 wamcor_mod.f90  ympi_mod.f90 wammpi_mod.f90 \
masnum_wam_mpi.f90

INC = \
get_wind.inc  mixture.inc     precom.inc    wamfio_mod_init.inc   \
mixture_limit.inc             mixture_wit.inc                     \
implsch.inc   nlweight.inc    propagat.inc                        \
intact.inc    outmix.inc      readwi.inc                          \
inter.inc     output.inc      setspec.inc                         \
io_rest.inc   outwav.inc      settopog.inc                        \
mean1.inc     out_wavmix.inc  set_uv.inc                          \
mean2.inc     outwav_t.inc    setwave.inc  outmix_wit.inc

QSRC=

VPATH= ../wave_cor/:../ympi_mod/

OBJS = ${SRC:.f90=.o}
QOBJS= ${QSRC:.f90=.o}

.SUFFIXES:
.SUFFIXES: .f90 .o
.f90.o:
        $(F77) $(INC_PATH) $(FFLAGS) -c $<


$(EXE)  :$(INC) $(OBJS)  $(QOBJS)
        $(F77) $(LF77OPTS) -o $@  $(OBJS) $ $(QOBJS) $(LIBS)

$(OBJS) $(QOBJS) : $(INC)

clean:
        rm -f  $(EXE) *.o *.mod

cleano:
        rm -f  *.o *.mod
```

Compiling procedures

- Enter the directory masnum_wave/source/bin
- Modify the necessary parameters in the “makefile”

```
F77 = mpif90
LF77OPTS =
FFLAGS   =

EXE = masnum.wam.mpi

NETCDF_PATH = /WORK/home/fio_climate1/tools/netcdf/netcdf-3.6.2

LIBS = -L$(NETCDF_PATH)/lib -lnetcdf -L/usr/local/lib

INC_PATH = -I$(NETCDF_PATH)/include

```

Compiling procedures

- Enter the directory masnum_wave/source/bin
- Modify the necessary parameters in the “makefile”
- Command: make
- Check the executable file “masnum.wam.mpi”

Setup the experiment 1 (exp1)

- Enter the directory masnum_wave/exp/exp1
- Modify the necessary parameters in the “exp1_run.csh”

```
#!/bin/tcsh -f
#-----------------------------------------------------------------------------
# ************ Part I. Set parameters for the MASNUM_WAVE model. **************
#-----------------------------------------------------------------------------

# --- If need to make the executable file, set needmake as "YES", else "NO".
set needmake = "YES"

# --- Set number of processors for MPI run.
set nproc = 24

# --- Path for wave model running.
set masnum_home = $HOME/szhy/masnum_wave

# --- File of depth for wave model.
set depfile = $masnum_home/inputdata/topo/topo_pacific_p15.nc

# --- File of ice mask for wave model.
#set icefile = $masnum_home/inputdata/topo/ice_clim_mask.nc

# --- Path for code, w/subpath: scripts, wave_cor & pre_time
set src_path = $masnum_home/source

# --- Path for wind & model setting.
#     NOTE: end with slash; keep it agree with windtype.
set wind_path = $masnum_home/inputdata/forcing/

#-----------------------------------------------------------------------------

set title      = "pac_ncep"    # --- Symbal for model output.
set istime     = 20090101      # --- Integral start time
set ietime     = 20090301      # --- Integral end time
set cools_days = 0             # --- The time (days) for cool start.
set delttm     = 7.5           # --- Length of integral time step, in minutes.
                               # --- Maximum value is 7.5 in this example

set wndfreq = 6   # --- The frequence of wind data (hours).
set wndtype = 3   # --- The wind type:
                  #  0  for wind in the same grid with model, files by monthly.
                  #  1  for GFS wind (0.5 * 0.5), no interp.
                  #  2  for QuikSCAT BLN wind (0.5 * 0.5), interp.
                  #  3  for NCEP re-anal wind, with interp.
set outflag = 3   #  output wave variables into file multi-records,
                  #  1    : one file every year,
                  #  2    : one file every month,
                  #  3    : one file every day,
                  #  else : one file every run.
set wiofreq = 24  # --- The output frequence for wave results (hour).
set ciofreq = 24  # --- The output frequence for current coef.s (hour).
set rstfreq = 24  # --- The output frequence for model restart (hour).

#-----------------------------------------------------------------------------
# ************ PART II. Prepare work directory, executable files, **************
# @@@  NOTE: The following part is not necessary to change, just keep them.  @@@
#-----------------------------------------------------------------------------

set BIN = $masnum_home/source/bin
set EXP = `pwd`
if ($needmake == "YES")then
  cd $BIN
  make -f makefile clean
  make -f makefile
endif
cd $EXP

#-----------------------------------------------------------------------------

rm -rf masnum.wam.mpi wamyyz.nc ice_clim_mask.nc wave_rest.nc
cp $BIN/masnum.wam.mpi masnum.wam.mpi
ln -s $depfile            wamyyz.nc
#ln -s $depfile            ice_clim_mask.nc

#-----------------------------------------------------------------------------

cat > ctlparams << EOF
&CTLPARAMS
data_path   = ""          ,
wind_path   = "$wind_path",
TITLE       = "$title"    ,
CISTIME     = $istime     ,
CIETIME     = $ietime     ,
COOLS_DAYS  = $cools_days ,
DELTTM      = $delttm     ,
WNDFREQ     = $wndfreq    ,
WNDTYPE     = $wndtype    ,
OUTFLAG     = $outflag    ,
WIOFREQ     = $wiofreq    ,
CIOFREQ     = $ciofreq    ,
RSTFREQ     = $rstfreq
/
EOF

#-----------------------------------------------------------------------------

mpirun -np $nproc ./masnum.wam.mpi > out.qrunout

#-----------------------------------------------------------------------------

exit

#-----------------------------------------------------------------------------
# ***************************** THE END ****************************************
```

```
1  #!/bin/tcsh -f
2  #-------------------------------------------------------------------------------
3  # ************ Part I. Set parameters for the MASNUM_WAVE model. **************
4  #-------------------------------------------------------------------------------
5
6  # --- If need to make the executable file, set needmake as "YES", else "NO".
7  set needmake = "YES"
8
9  # --- Set number of processors for MPI run.
10 set nproc = 24
11
12 # --- Path for wave model running.
13 set masnum_home = $HOME/szhy/masnum_wave
14
15 # --- File of depth for wave model.
16 set depfile = $masnum_home/inputdata/topo/topo_pacific_p15.nc
17
18 # --- File of ice mask for wave model.
19 #set icefile = $masnum_home/inputdata/topo/ice_clim_mask.nc
20
21 # --- Path for code, w/subpath: scripts, wave_cor & pre_time
22 set src_path = $masnum_home/source
23
24 # --- Path for wind & model setting.
25 #     NOTE: end with slash; keep it agree with windtype.
26 set wind_path = $masnum_home/inputdata/forcing/
27
28 #-------------------------------------------------------------------------------
29
30 set title      = "pac_ncep"    # --- Symbal for model output.
31 set istime     = 20090101      # --- Integral start time
32 set ietime     = 20090301      # --- Integral end time
33 set cools_days = 0             # --- The time (days) for cool start.
34 set delttm     = 7.5           # --- Length of integral time step, in minutes.
35                                # --- Maximum value is 7.5 in this example
36
37 set wndfreq = 6   # --- The frequence of wind data (hours).
38 set wndtype = 3   # --- The wind type:
39                   #  0  for wind in the same grid with model, files by monthly.
40                   #  1  for GFS wind (0.5 * 0.5), no interp.
41                   #  2  for QuikSCAT BLN wind (0.5 * 0.5), interp.
42                   #  3  for NCEP re-anal wind, with interp.
43 set outflag = 3   #  output wave variables into file multi-records,
44                   #  1    : one file every year,
45                   #  2    : one file every month,
46                   #  3    : one file every day,
47                   #  else : one file every run.
48 set wiofreq = 24  # --- The output frequence for wave results (hour).
49 set ciofreq = 24  # --- The output frequence for current coef.s (hour).
50 set rstfreq = 24  # --- The output frequence for model restart (hour).
51
52 #-------------------------------------------------------------------------------
53 # ************ PART II. Prepare work directory, executable files, **************
54 # @@@  NOTE: The following part is not necessary to change, just keep them.  @@@
55 #-------------------------------------------------------------------------------
56
57 set BIN = $masnum_home/source/bin
58 set EXP = `pwd`
59 if ($needmake == "YES")then
60   cd $BIN
61   make -f makefile clean
62   make -f makefile
63 endif
64 cd $EXP
65
66 #-------------------------------------------------------------------------------
67
68 rm -rf masnum.wam.mpi wamyyz.nc ice_clim_mask.nc wave_rest.nc
69 cp $BIN/masnum.wam.mpi masnum.wam.mpi
70 ln -s $depfile            wamyyz.nc
71 #ln -s $depfile            ice_clim_mask.nc
72
73 #-------------------------------------------------------------------------------
74
75 cat > ctlparams << EOF
76 &CTLPARAMS
77 data_path   = ""          ,
78 wind_path   = "$wind_path",
79 TITLE       = "$title"    ,
80 CISTIME     = $istime     ,
81 CIETIME     = $ietime     ,
82 COOLS_DAYS  = $cools_days ,
83 DELTTM      = $delttm     ,
84 WNDFREQ     = $wndfreq    ,
85 WNDTYPE     = $wndtype    ,
86 OUTFLAG     = $outflag    ,
87 WIOFREQ     = $wiofreq    ,
88 CIOFREQ     = $ciofreq    ,
89 RSTFREQ     = $rstfreq
90 /
91 EOF
92
93 #-------------------------------------------------------------------------------
94
95 mpirun -np $nproc ./masnum.wam.mpi > out.qrunout
96
97 #-------------------------------------------------------------------------------
98
99 exit
100
101 #-------------------------------------------------------------------------------
102 # ***************************** THE END ****************************************
```

Very important
Control file

Setup the experiment 1 (exp1)

- **Enter the directory masnum_wave/exp/exp1**
- **Modify the necessary parameters in the "exp1_run.csh"**
- **Submit the shell "exp1_run.csh"**
- **Validate the results**

- Compare the variable "hs" in the output files

Setup the experiment 1 (exp1)

- **Enter the directory masnum_wave/exp/exp1**
- **Modify the necessary parameters in the "exp1_run.csh"**
- **Submit the shell "exp1_run.c**
- **Validate the results**

- Compare the variable "hs"

```
Macintosh:exp1 songzhenya$ ./compare_exp1
 compare HS between pac_ncep_wav_20090228.nc and
 pac_ncep_wav_20090228_stardard.n
 Step 1: Open file pac_ncep_wav_20090228.nc
 Step 1 Success
 Step 2: Open file pac_ncep_wav_20090228_stardard.n
 Step 2 Success
 Step 3
 Step 3.1 Compare missing_value
 Step 3.1 Success
 Step 3.2 Compare scale_factor
 Step 3.2 Success
 Step 3.3 Compare HS
 Step 3.3 Success
 Compare Success
Macintosh:exp1 songzhenya$
```

Setup the experiment 1 (exp1)

- **Enter the directory masnum_wave/exp/exp1**
- **Modify the necessary parameters in the "exp1_run.csh"**
- **Submit the shell "exp1_run.csh"**
- **Validate the results**

- Compare the variable

```
Macintosh:exp1 songzhenya$ ./compare_exp1
 compare HS between pac_ncep_wav_20090228.nc and
 pac_ncep_wav_20090228_stardard.n
 Step 1: Open file pac_ncep_wav_20090228.nc
 Step 1 Success
 Step 2: Open file pac_ncep_wav_20090228_stardard.n
 Step 2 Success
 Step 3
 Step 3.1 Compare missing_value
 Step 3.1 Success
 Step 3.2 Compare scale_factor
 Step 3.2 Success
 Step 3.3 Compare HS
 Error! HS do not match at (i,j) =            1 ,            1
Macintosh:exp1 songzhenya$
```

Suggestions

Suggestion for optimization

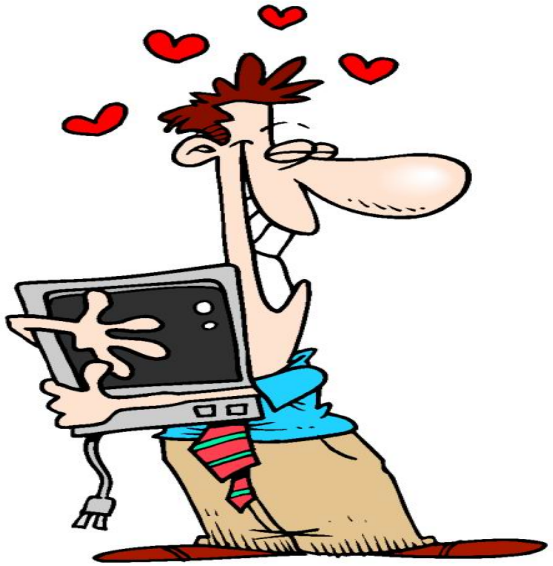
Program analysis

- Load balance
- Hotspot function

Optimization in one node

- compiler flags
- scalar optimization
- vectorization optimization
- parallelization (MPI/OpenMP)

Optimization in many nodes

- MPI
- OpenMP
- Load balance

Thank You!