FUNWAVE-TVD WORKSHOP TUTORIAL

July 25-27, 2017
Embassy Suites, Newark DE

COASTAL &
HYDRAULICS
LABORATORY

COASTAL AND HYDRAULICS LABORATORY

Prerequisites for the FUNWAVE Workshop

- Your laptop is able to use WiFi to access to the internet
- You can use SSH Secure Shell Client (usually Putty or Cygwin on Windows)
- If your laptop uses Windows OS, it is recommended that you download the latest version of cygwin (http://www.cygwin.com). Cygwin is a bash shell/unix emulation program and contains many of the tools such as tar, gzip/gunzip, and cpp, which will be useful for installation and compilation of FUNWAVE-TVD.
- Having a postprocessing toolbox on your machine (laptop). Both Matlab and Python post-processing script are provided with practice examples. Many participants already use Matlab, but if you do not, it is recommended that you install a Python package (information below).

Downloading and Installing a Python Package (Anaconda)

The best and most comprehensive FREE package for the Python language, along with most tools and modules (e.g., NumPy, Matplotlib, etc.) is distributed by the Continuum Analytics under the Anaconda package. It is available for Linux, Mac OS X, and Windows machines. You do **NOT** need administrator privileges to install the Anaconda package, you can do so as a standard user on all three platforms listed above.

For the Anaconda package go to: https://www.continuum.io/downloads

Pick the **appropriate platform** (Linux, Mac OS X, Windows) by clicking the correct tab and get the Anaconda distribution that comes with **Python 3.6** (not 2.7). You can either download the **Graphical Installer** (recommended), of if you are comfortable with the terminal in the Linux/Mac OS X environment, you can also download it through the command line.

For the complete list of packages/modules included in the Anaconda Python distribution see: https://docs.continuum.io/anaconda/packages/pkg-docs

Training session # 1 (Tuesday, lead: Matt Malej)

Topics

- FUNWAVE-TVD and Parallel Computing (MPI) - Documentation Wiki

  Wiki ⇒ https://fengyanshi.github.io/build/html/index.html

- Latest Version (3.0) - current capabilities and those in development

  Tag/Version Release version 3.0 ⇒
  https://github.com/fengyanshi/FUNWAVE-TVD/releases/tag/v3.0

- Where do I get the code? - Version Control (Github)

  Full Repository ⇒ https://github.com/fengyanshi/FUNWAVE-TVD

- Sandbox for USACE and DoD member with GUI [private on US Army side]

  Link (need access) ⇒ https://github.com/malej/funwaveGUI

- How to build (compile/link) and install FUNWAVE-TVD on different machines for parallel computation?

  https://fengyanshi.github.io/build/html/setup.html#compile-and-setup

Practice

1) Log into ==> ***mills***

   ```
   > ssh your_user_id@mills.hpc.udel.edu
   > input your password
   > mkdir your_funwave_folder
   ```
   (optional, otherwise all will be in $HOME)

   NOTE: **$HOME** will be **/home/funwave_tvd_workshop/tvdguest##**

   We suggest using **sftp** or **scp** to transfer data to/from **mills**

   ```
   > sftp your_user_id@mills.hpc.udel.edu
   ```
   or
   ```
   > scp your_user_id@mills.hpc.udel.edu:/home/your_user_id/test.txt  ./
   > input your password
   ```

2) **Clone** FUNWAVE-TVD Package

```
> cd your_funwave_folder (if you created in step 1)
> git clone https://github.com/fengyanshi/FUNWAVE-TVD.git
```

3) Compile the source code

```
> cd FUNWAVE-TVD/src
> emacs Makefile-Mills    (if you want to modify the source file)
```

NOTE: Modify the Makefile if needed. Emacs/vi/gedit are text editors. You can use any other editor you are familiar with on the Unix/Linux system. To exit emacs editor press (Ctrl X then Ctr C).

```
EXEC = funwave_surface_wave
FLAG_1 = -DDOUBLE_PRECISION
FLAG_4 = -DCARTESIAN
FLAG_2 = -DPARALLEL (if you want to run with parallel mode)
FC = mpif90 (on mills)
```

The compiled/linked executable file will be ***funwave_surface_wave*** *inside the* ***src*** *directory*

- Compile the code for surface wave applications

```
> vpkg_require openmpi  (load mpi libraries if you run with parallel)
> make clean (or 'make clobber' if you want to remove the executable)
> make -f Makefile-Mills
```

- Compile the code for ship-wake applications **[different executable]**

  Besides FLAGs used for surface wave applications, add/change

```
FLAG_12 = -DVESSEL
EXEC = funwave_ship_wake
```

```
> vpkg_require openmpi  (load mpi libraries if you run with parallel)
> make clean (or 'make clobber' if you want to remove the executable)
> make -f Makefile-Mills-Vessel
```

- Compile the code for spherical coordinates

  Use FLAGs for surface wave applications, but **remove**

```
FLAG_4 = -DCARTESIAN
```

```
> vpkg_require openmpi  (load mpi libraries if you run with parallel)
> make clean
> make -f Makefile-Mills-Spherical
```

Training session # 2 (Tuesday, lead: **Matt Malej)**

Topics

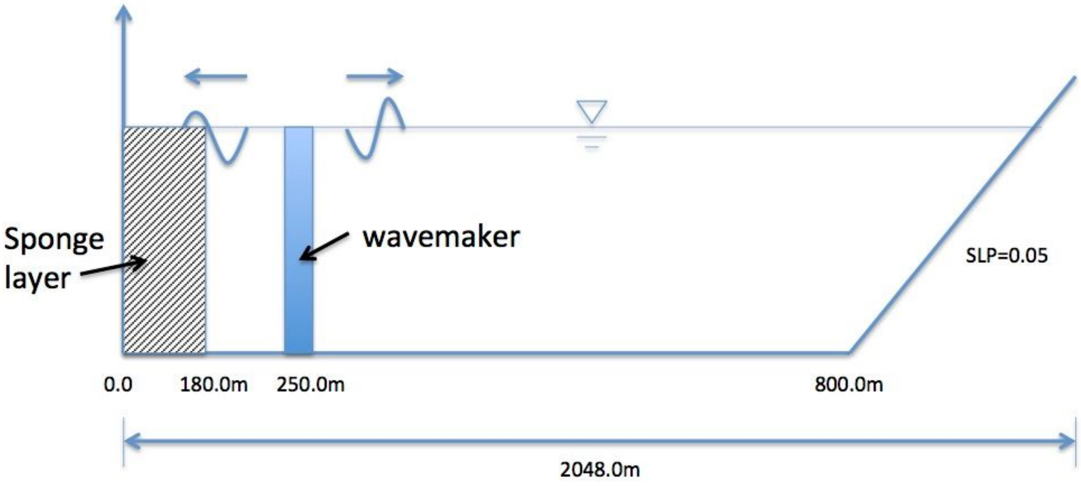
- How to run FUNWAVE-TVD? Navigating the basic sections within the INPUT file for different simulations (numerics, physics, input, output, etc.).

  Wiki Direct Link ⇒ https://fengyanshi.github.io/build/html/definition.html

- Setting up (Linux/Mac OS X and HPC machines with PBS scheduler), running, and post-processing your first FUNWAVE-TVD simulation (1D beach runup or levee overtopping with shoaling and wetting/drying).

Practice

1) Surface Waves on 1D Sloped Beach



Model configuration: Grid dimensions: 1024X3. Grid sizes: DX=DY=2m. Depth at flat bottom: 10 m, Beach slope: 1/20.

- Go to the directory of the 1D case

```
> cd FUNWAVE-TVD/simple_cases/surface_wave_1d
```

- Create a **work** directory

```
> mkdir work
```

- Copy and rename one of input the files into a work folder (regular wave as an example)

```
> cd work
> cp ../input_files/input_reg.txt  input.txt
```

- Copy compiled executable into the **work** directory

```
> cp ../../../src/funwave_mills  ./
```

- Check and Modify `input.txt`

The following statements are necessary in the `input.txt`

```
Parallel (if applicable)
   PX = 4
   PY = 1

Depth
   DEPTH_TYPE = SLOPE
   DEPTH_FLAT = 10.0
   SLP = 0.05
   Xslp = 800.0

Dimensions
   Mglob = 1024
   Nglob = 3

Time
   TOTAL_TIME = 200.0
   PLOT_INTV = 10.0
   SCREEN_INTV = 10.0

Grid sizes
   DX = 1.0
   DY = 1.0

Add wavemaker
   WAVEMAKER = WK_REG
   DEP_WK = 10.0
   Xc_WK = 250.0
   Yc_WK = 0.0
   Tperiod = 12.0
   AMP_WK = 0.5
   Delta_WK = 1.0 ! the default is 0.5, set a larger number for nonlinear waves

Add sponge layer
   FRICTION_SPONGE = T
   DIRECT_SPONGE = T
   Sponge_west_width = 180.0
   Sponge_east_width = 0.0
   Sponge_south_width = 0.0
   Sponge_north_width = 0.0

Breaking scheme (default: SWE breaker)
   VISCOSITY_BREAKING = T
   Cbrk1 = 0.65
   Cbrk2 = 0.35

Wetting and Drying
MinDepth=0.01
```

Output

```
RESULT_FOLDER = output/
ETA = T
MASK = T
```

- **Run the Model**

```
> cp ~/FUNWAVE-TVD/simple_cases/YourNamePBS.qs ./run_script.qs
> qsub run_script.qs
```

NOTE: make sure run_script.qs is in the current work directory.

- **Post-Process Your Results**

Download your results to your laptop**(eta_#####, mask_#####, dep.out)** and use the provided Matlab or Python scripts (using either **sftp** or **scp**).

```
> cd directory_where_you_want_to_download_your_results
> scp -r your_user_id@mills.hpc.udel.edu:/home/your_user_id/FUNWAVE-TVD/simple_cases/surface_wave_1d/work  ./
> enter your password
```

NOTE: You will need to modify the output folder name in Matlab/Python scripts to load your results.

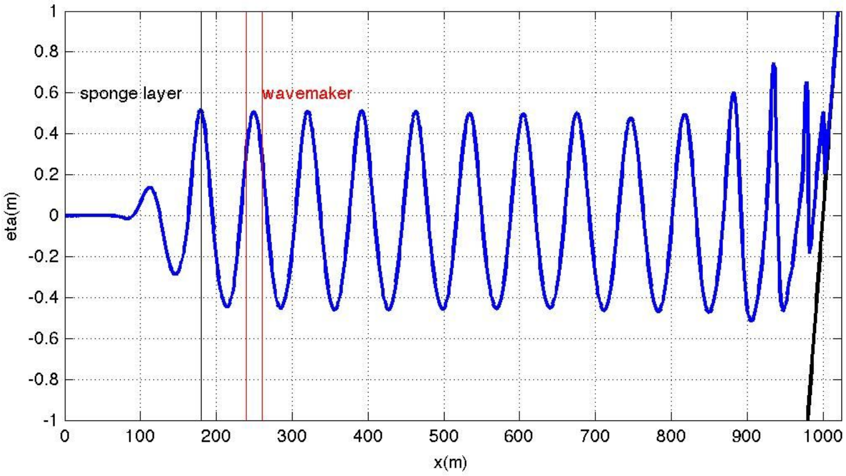
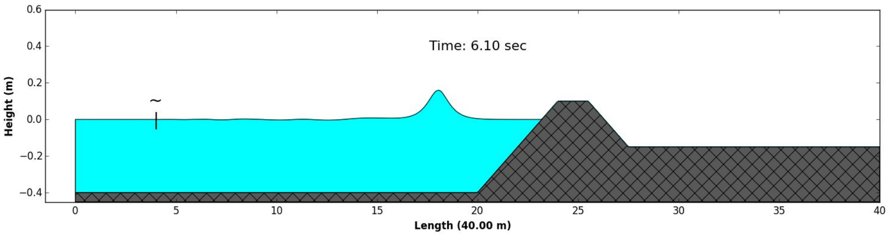


Figure: an example of plot using matlab **plot_wave.m**

2) EXTRA CHALLENGE - PROGRESSION

For those participants who were able to to complete the above task and want an additional challenge, try to set up the following

a) Modify the **input.txt** to force the wavemakers with a **Solitary Wave** as input (**hint: WAVEMAKER=INIT_SOL**), of **0.16 meter amplitude**.
b) Use the supplied bathymetry (**depth_levee.txt**) as the **'DATA'** depth type with dimensions of [500 x 3] points.
c) The depth at the wavemaker is **0.4** meters and it is located **4.0** meters from the left boundary.
d) Set the **TOTAL_TIME** to **30.0** seconds with a **PLOT_INTERVAL** of **0.1** seconds.
e) Activate the sponges (**DIRECT_SPONGE** only) on the **west** (2.0m) and **east** (1.0m) side.
f) Set the spatial discretization to **dx=dy=0.08** meters, wetting/drying to 1 millimeter (**MinDepth=0.001**), **FroudeCap=2.0**, and **CFL** condition to **0.1**.
g) Output the **depth**, surface elevation (**eta**), and **mask**.



Training session # 3 (Wednesday, lead: Fengyan Shi)

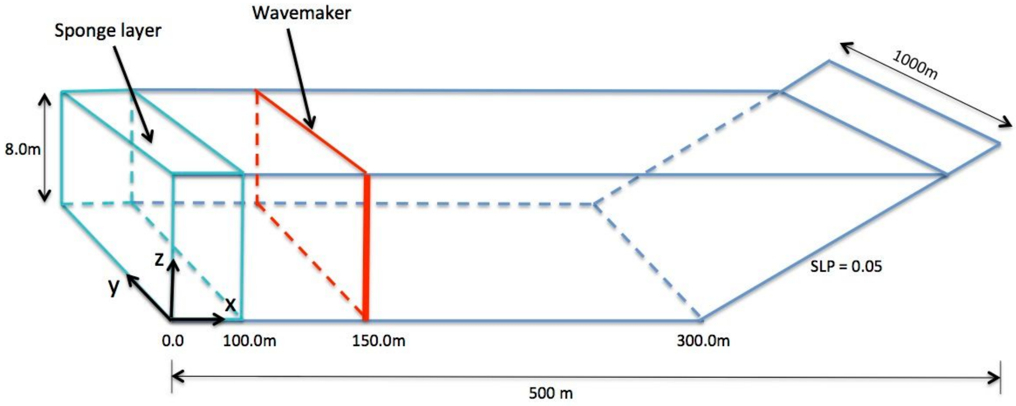
Topics

2D plane beach case with various wave conditions (different wavemakers)

1) Monochromatic waves
2) Irregular waves (TMA spectrum)

Practice

1) Waves on 2D plane beach



Model configuration: Grid dimensions: 250X500. Grid sizes: DX=DY=2m. Depth at flat bottom: 8 m, Beach slope: 1/20.

- Go to directory `FUNWAVE-TVD/simple_cases/beach_2d`
- Create a work directory

```
> mkdir work
```

- Copy and rename one of input files into the work directory (regular wave as an example)

```
> cd work
> cp ../input_files/input_reg.txt  input.txt
```

- Check and Modify `input.txt`

  The following statements are necessary in input.txt

```
Slope bed
  DEPTH_TYPE = SLOPE
  DEPTH_FLAT = 8.0
  SLP = 0.05
  Xslp = 300.0

Output folder
  RESULT_FOLDER = output/

Dimensions
  Mglob = 250
  Nglob = 500

Grid sizes
  DX = 2.0
  DY = 2.0

Add wavemaker
  WAVEMAKER = WK_REG
  DEP_WK = 8.0
  Xc_WK = 150.0
  Yc_WK = 0.0
  Tperiod = 8.0
  AMP_WK = 0.5
  Theta_WK = 30.0
  Delta_WK = 3.0

Add periodic boundary condition
  PERIODIC = T

Sponge layer
  DIFFUSION_SPONGE = F
  FRICTION_SPONGE = T
  DIRECT_SPONGE = T
  Csp = 0.0
  CDsponge = 1.0
  Sponge_west_width = 100.0
  Sponge_east_width = 0.0
  Sponge_south_width = 0.0
  Sponge_north_width = 0.0

Wave breaking
  VISCOSITY_BREAKING = T
  Cbrk1 = 0.65
  Cbrk2 = 0.35

Wave average property
  T_INTV_mean = 100.0
  STEADY_TIME=100.0

Output
  ETA = T
  Umean = T
  Vmean = T
  ETAmean = T
  MASK = T
  WaveHeight = T
```

- Use the same procedures as in the previous sessions to run the model and post-process results

2) Try different **`wavemakers`**, **`sponge layers`**, **`wave breaking scheme`**, **`non periodic boundary condition`**, `etc.`

3) EXTRA CHALLENGE - PROGRESSION: analysis of wave-averaged properties such as the significant wave height (**`Hsig`**) and wave-induced currents. You can also try the rip current case in /simple_cases/rip_2d/

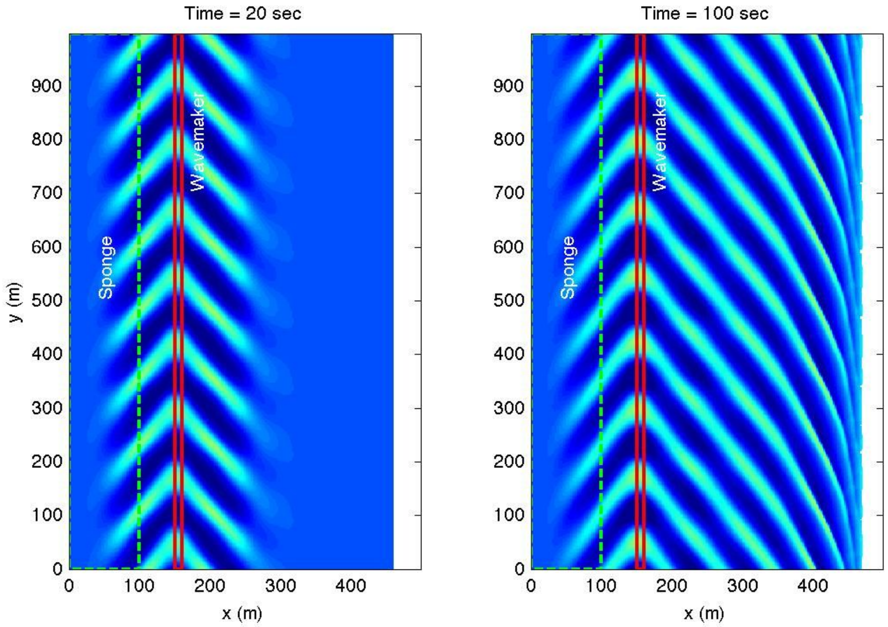


*Figure: an example of plot using **`plot_wave.m`**, regular obliquely incident waves.*

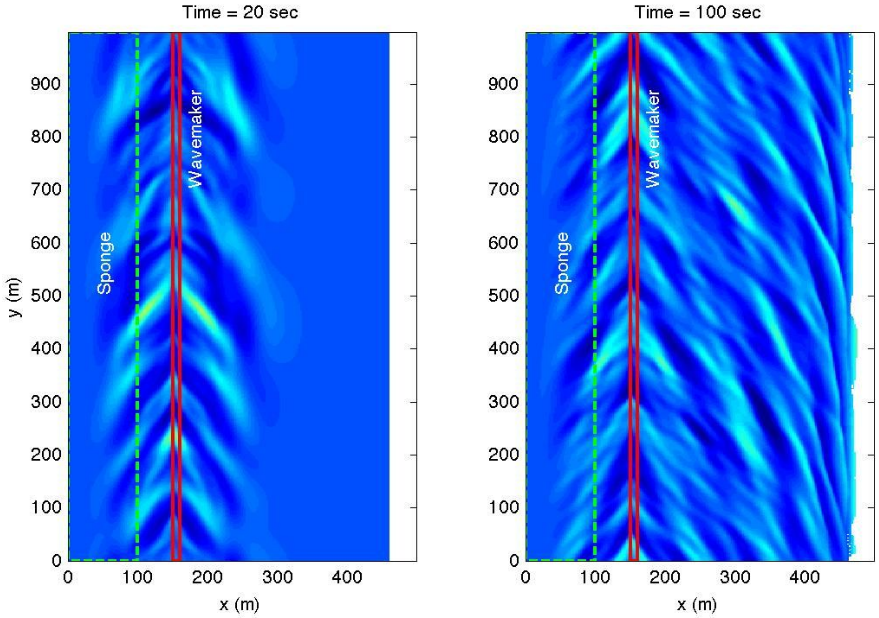


*Figure: an example of plot using **`plot_wave.m`**, irregular obliquely incident waves.*

Training Session # 4 (Wednesday, lead: Fengyan Shi)

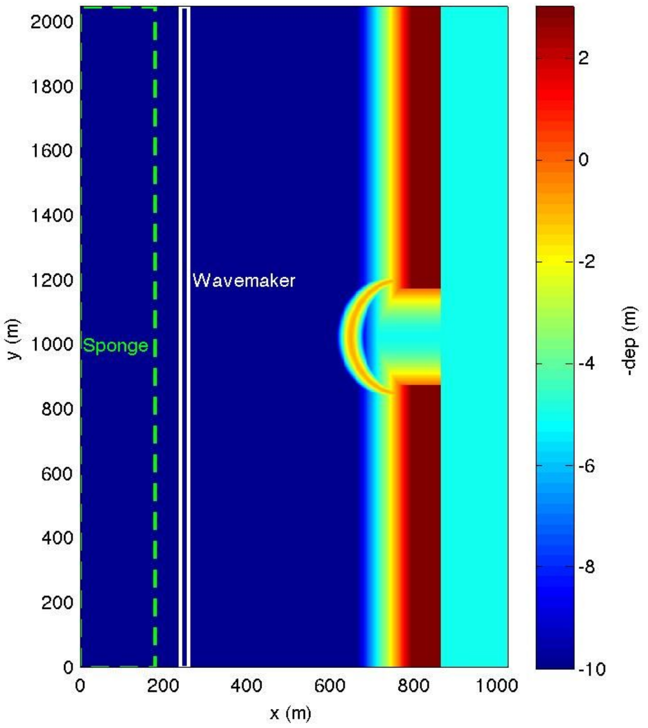
Topics

- Wave simulation on 2D random bathymetry with complex shoreline geometries

  1) Inlet Problem (to include post-processing analysis of Harbor Resonance)
  2) Obstacles and Breakwaters (partially absorbing and reflecting inner boundaries)

Practice

1) Inlet Problem



Model configuration: Grid dimensions: 512X1024. Grid sizes: DX=DY=2m. The bathymetry/topography include flat bottom, barrier beaches, shoal, inlet and shallow basin.

- Go to **`FUNWAVE-TVD/simple_cases/inlet_shoal`**
- Create a work directory

```
> mkdir work
```

- Copy and rename one of input files into the work directory (regular wave as an example)

```
> cd work
> cp ../input_files/input_reg.txt   input.txt
```

- Check and Modify `input.txt`

  The following statements are necessary in input.txt

```
 Parallel (if applicable) … running here on 16 CPU’s (or cores, or ranks)
          PX = 4
          PY = 4

 Depth
          DEPTH_TYPE = DATA
          DEPTH_FILE = ../bathy/dep_shoal_inlet.txt

Output folder
          RESULT_FOLDER = output/


Dimensions
          Mglob = 512
          Nglob = 1024

Time
          TOTAL_TIME = 1200.0
          PLOT_INTV = 30.0
          PLOT_INTV_STATION = 0.5
          SCREEN_INTV = 30.0

Grid sizes
          DX = 2.0
          DY = 2.0

Wavemaker
          WAVEMAKER = WK_REG
          DEP_WK = 10.0
          Xc_WK = 250.0
          Yc_WK = 0.0
          Tperiod = 12.0
          AMP_WK = 1.0
          Theta_WK = 0.0

Sponge layer
          FRICTION_SPONGE = T
          DIRECT_SPONGE = T
          Csp = 0.0
          CDsponge = 1.0
          Sponge_west_width = 180.0
          Sponge_east_width = 0.0
          Sponge_south_width = 0.0
          Sponge_north_width = 0.0

Wetting and drying
          MinDepth=0.01

Breaking scheme
          VISCOSITY_BREAKING = T
```

```
Cbrk1 = 0.65
Cbrk2 = 0.35
```

Wave averaging property

```
T_INTV_mean = 240.0
STEADY_TIME=480.0
```

Output

```
ETA = T
MASK = T
WaveHeight = T
```

- Use the same procedures as in the previous sessions to run the model and post-process results

EXTRA CHALLENGE - PROGRESSION:
Try different options, for example, **`periodic boundary condition for obliquely incident waves`**, **`different wavemaker and wave parameters`**.

Add periodic boundary condition: PERIODIC = T

3) Obstacles and Breakwaters (partially absorbing and reflecting inner boundaries)

For obstacles: **Add an OBSTACLE FILE**

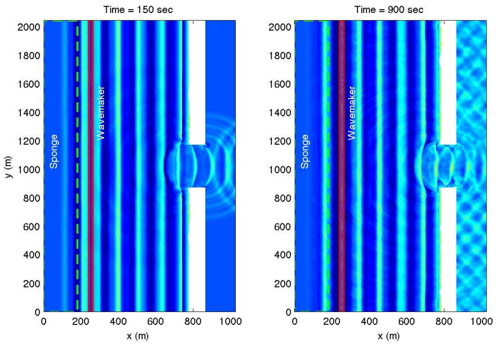
```
OBSTACLE_FILE = ../bathy/obs_shoal_inlet.txt
```

For breakwaters with full reflection: Add a breakwater structure in the bathymetry file

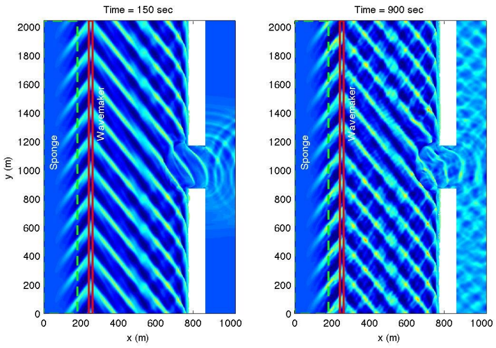
```
DEPTH_FILE = ../bathy/dep_shoal_inlet_brk.txt
```

For Breakwaters with partially absorbing and reflecting inner boundary conditions: In addition to the bathymetry with the breakwater structure, **add BREAKWATER FILE for partial reflection**

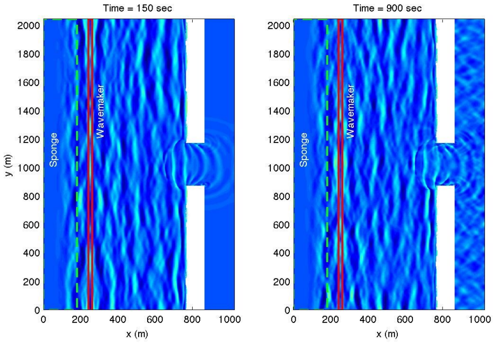
```
BREAKWATER_FILE = brk_shoal_inlet.txt
```



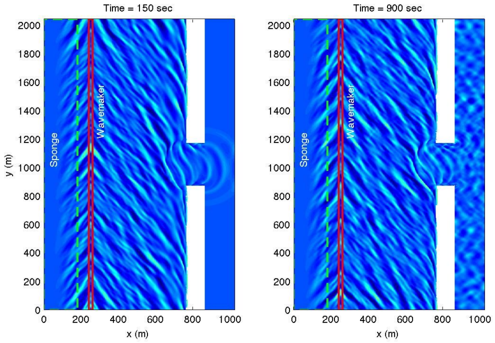
Case: monochromatic, normal incidence



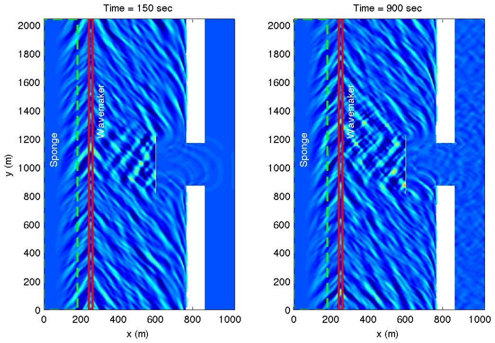
Case: monochromatic, oblique incidence



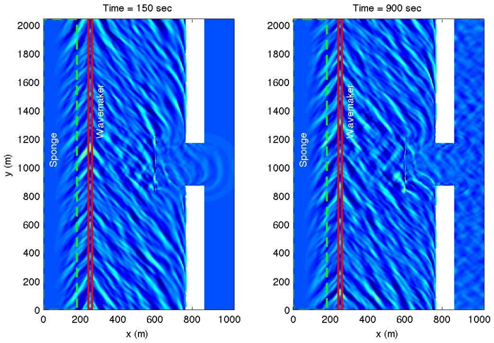
Case: irregular, normal incidence



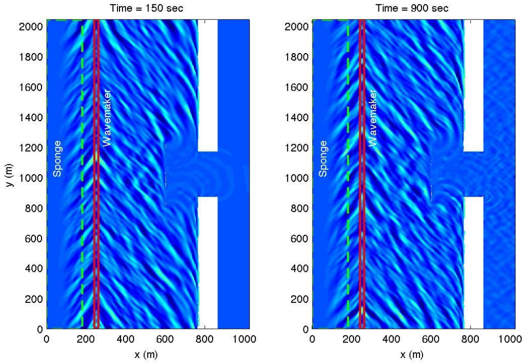
Case: irregular, oblique incidence



Case: breakwater using obstacle option



Case: breakwater by modifying depth file



Case: partially reflecting/absorbing breakwater

EXTRA CHALLENGE - PROGRESSION:

1) Analysis of harbor resonance.
2) Set up your own surface wave case

Training Session # 5 (Thursday, lead: **Fatima Nemati)**

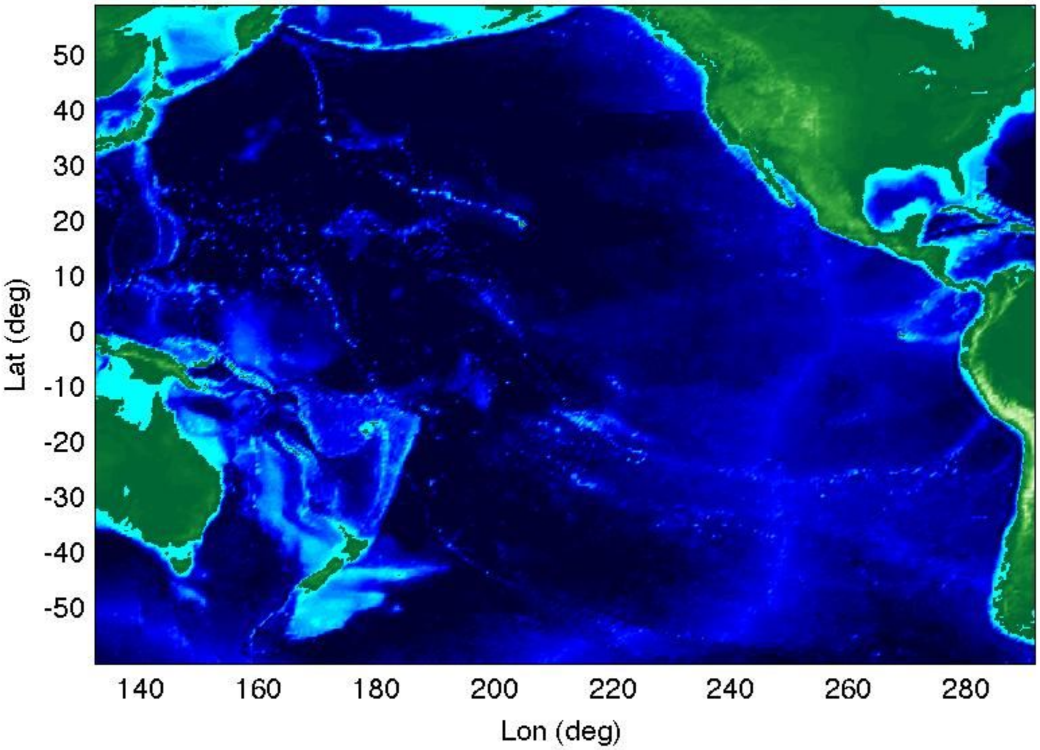
Topics

- Tsunami simulation

1) Tsunami simulation using the `spherical coordinates`
2) Tsunami sources

Practice

1) Tohoku Tsunami



Model configuration:Computational domain covers a region of the Pacific Ocean from 60°S to 60°N in the south-north direction, and from 132°E to 68°W in the west-east direction. The example is a 30min x 30min resolution case. Grid dimensions: 320x240, Grid sizes Dphi=Dtheta=0.5 deg.

- Recompile the source code without the flag **`-DCARTESIAN`** in Makefile
- Go to **`FUNWAVE-TVD/simple_cases/tohoku_tsunami`**
- Copy the new executable into the current working directory

- Check and Modify **`input.txt`**

The following statements are necessary in **`input.txt`**

Parallel (if applicable)

```
PX = 2
PY = 2
```

Specify bathymetry

```
DEPTH_TYPE = DATA
DEPTH_FILE = ../external_files/depth_30min.txt
```

Dimensions

```
Mglob = 320
Nglob = 240
```

Grid

```
Lon_West = 132.0
Lat_South = -60.0
Dphi = 0.5
Dtheta = 0.5
```

Time

```
TOTAL_TIME = 86400.0
PLOT_INTV = 3600.0
PLOT_INTV_STATION = 1.0
SCREEN_INTV = 3600.0
```

Add initial conditions

```
INI_UVZ = T
ETA_FILE = ../external_files/ETA_30min.txt
U_FILE = ../external_files/U_30min.txt
V_FILE = ../external_files/V_30min.txt
```

Add Sponge layers

```
DIRECT_SPONGE = T
FRICTION_SPONGE = T
Sponge_west_width = 100000.0
Sponge_east_width = 100000.0
Sponge_south_width = 100000.0
Sponge_north_width = 100000.0
```

Add friction

```
Cd = 0.001
```

Avoid inundation in the basin scale (specify a large minimum depth)

```
MinDepth= 10.0
```

Stations/Wave Gaguges

```
NumberStations = 78
STATIONS_FILE = stations-pacific.txt
```

Output

```
RESULT_FOLDER = output/
ETA = T
Hmax = T
```

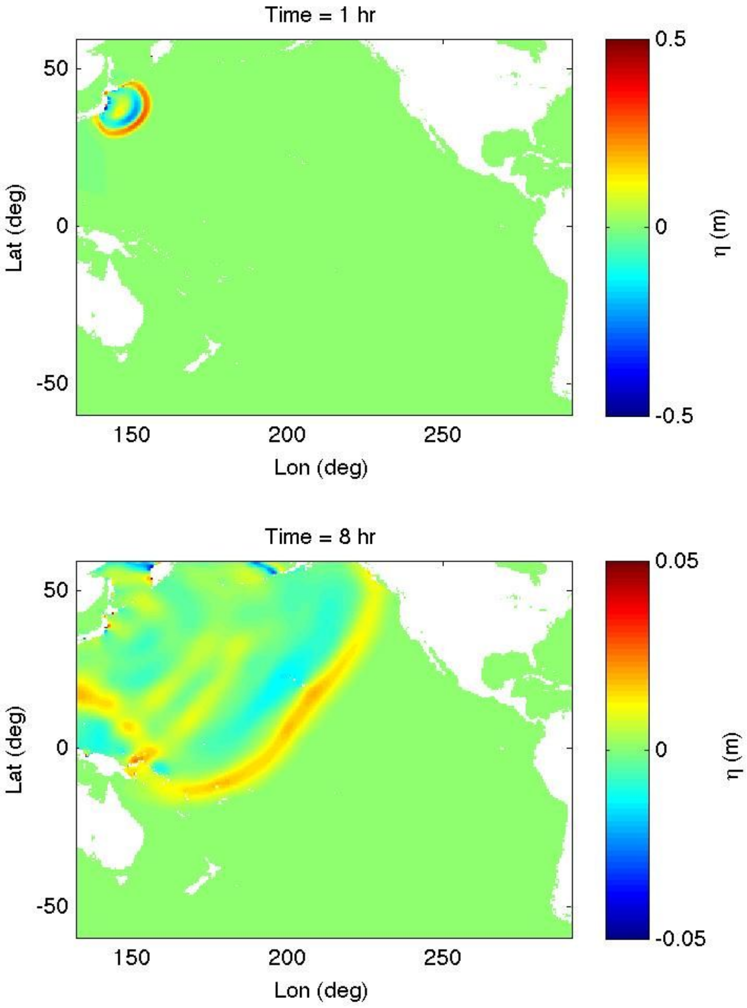


Figure: an example of plot using matlab code ***`plot_surface.m`***

Training Session # 6 (Thursday, lead: **Fengyan Shi)**

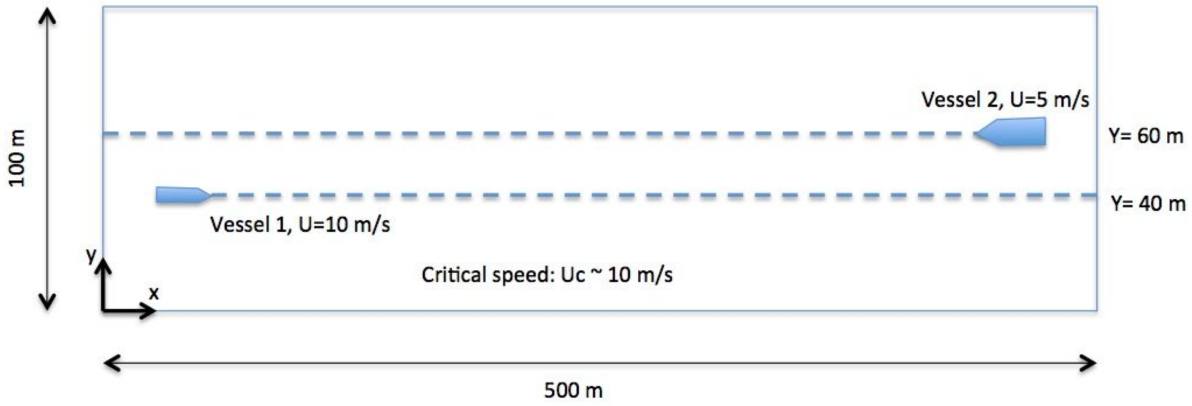
Topics

- Ship-wakes

  1) Setup with multiple vessels (`paths, size, velocities`)
  2) A vessel moving on random bathymetry

Practice

1) Setup with Multiple Vessels



Model configuration: Grid dimensions 500x100. Grid sizes DX=DY=1.0m. Vessel 1, length=10m, width=5m, draft=2.0m, speed=10m/s, starts from x=50m. Vessel 2, length=20m, width=8m, draft=3m, speed=5m/s, starts from 450m toward -x direction.

- Compile the code with **`-DVESSEL`** flag in the Makefile
- Go to **`FUNWAVE-TVD/simple_cases/vessel_flat_bottom`**
- Copy the executable to the current working directory
- Check and Modify **`input.txt`**

  The following statements are necessary in **`input.txt`**

Parallel Info (if use parallel)
```
        PX = 4
        PY = 1
```

Depth
```
        DEPTH_TYPE = FLAT
        DEPTH_FLAT = 10.0
```

Dimensions
```
        Mglob = 500
        Nglob = 100
```

Grid sizes

```
DX = 1.0
DY = 1.0
```

Set time

```
TOTAL_TIME = 50.0
PLOT_INTV = 1.0
PLOT_INTV_STATION = 50000.0
SCREEN_INTV = 1.0
```

Add vessels

```
VESSEL_FOLDER = ./
NumVessel = 2
```

NOTE: You will need two vessel files: **`vessel_00001`** and **`vessel_00002`** in the current folder.

In **`vessel_00001`**, specify:

```
Title: Vessel # 1
Blue_Star_I

Length(m), Width(m), Alpha(0.5), Beta(0.5), P(draft,m)
10.0 5.0, 0.5, 0.5, 2.0

Time, X(m), Y(m) (relative to the origin of the coordinates)
0.0 50.0 40.0
100.0 1050.0 40.0
```

In **`vessel_00002`**, specify:

```
Title: Vessel # 2
Blue_Star_II

Length(m), Width(m), Alpha(0.5), Beta(0.5), P(draft,m)
20.0 8.0, 0.5, 0.5, 3.0

Time, X(m), Y(m) (relative to the origin of the coordinates)
0.0 450.0 60.0
100.0 -50.0 60.0
```

Output

```
RESULT_FOLDER = output/
ETA = T
```

- Use the same procedures as in the previous sessions to run the model and post-process results

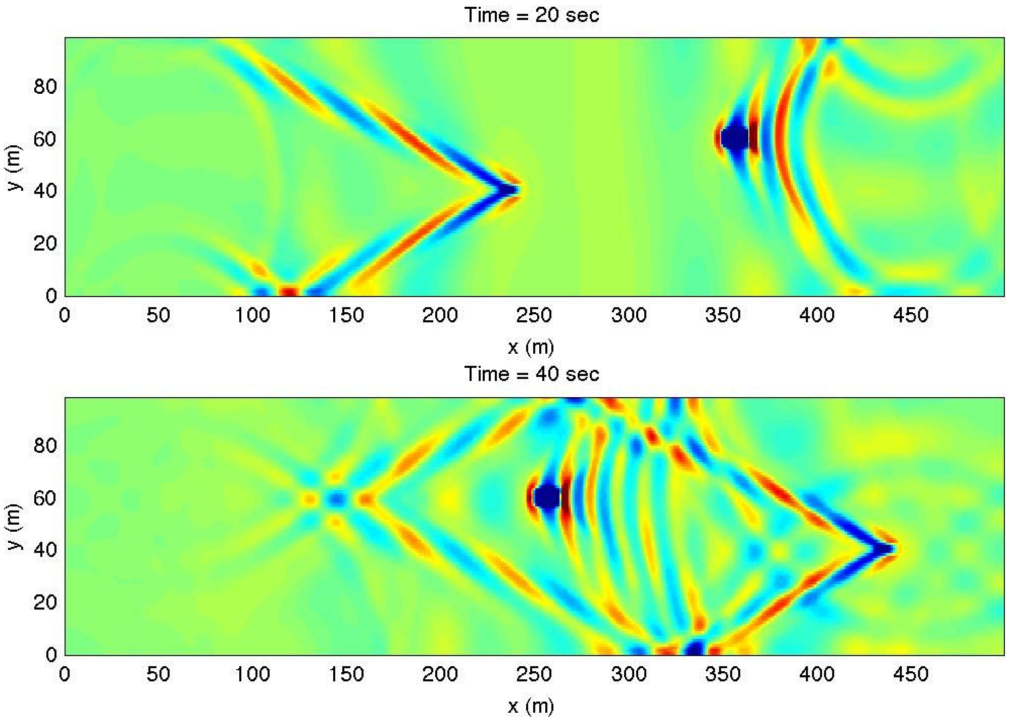
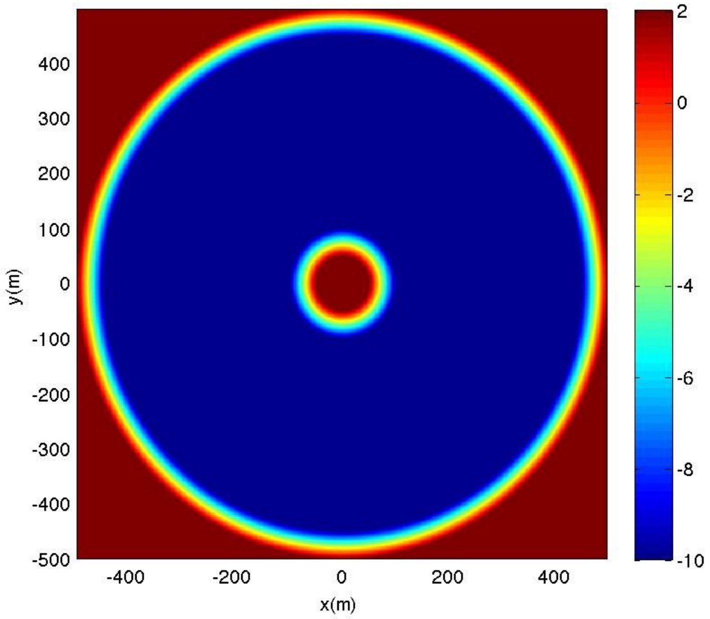


Figure: an example of plot using matlab code ***`plot_wave_vessel.m`***

2) A vessel moving in a random bathymetry



Model configuration: Grid dimensions 500x500. Grid sizes DX=DY=2.0m. Vessel is specified in vessel_00001. Vessel length=20m, width=10.0, draft=1m.

- Go to **`FUNWAVE-TVD/simple_cases/vessel_island_beach`**
- Check and Modify **`input.txt`**

  The following statements are necessary in **`input.txt`**

Parallel info (if use parallel) … 16 CPU's used for computation here

```
PX = 4
PY = 4
```

Depth

```
DEPTH_TYPE = DATA
DEPTH_FILE = depth.txt
```

Set up dimensions

```
Mglob = 500
Nglob = 500
```

Discretization

```
DX = 2.0
DY = 2.0
```

Time

```
TOTAL_TIME = 300.0
PLOT_INTV = 1.0
SCREEN_INTV = 1.0
```

Add a vessel

```
VESSEL_FOLDER = ./
NumVessel = 1
```

In **`vessel_00001`**, specify:

```
Title: Vessel # 1
Blue_Star_I

Length(m), Width(m), Alpha(m), Beta(m), P(unit)
20.0 10.0, 0.5, 0.5, 1.0
0.0000000e+00, 5.6000000e+02, 5.0000000e+02
1.0000000e+00, 5.6374897e+02, 5.0255132e+02

…
```

Output

```
RESULT_FOLDER = output/
ETA = T
```

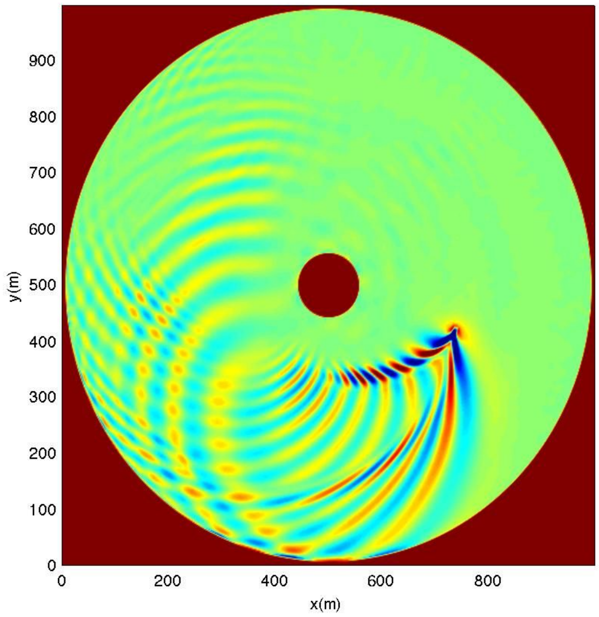
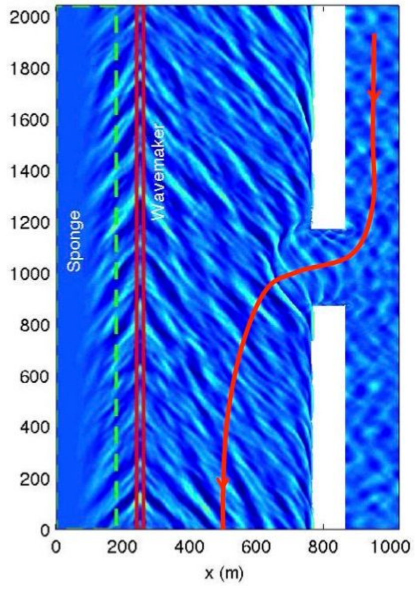


Figure: example of plot using matlab code ***plot_wave_vessel_island.m***

EXTRA CHALLENGE - PROGRESSION:

Design vessel paths in the inlet-shoal domain. Consider the interaction between wind waves and ship-waves



An example of vessel path in the inlet-shoal domain.