Running WWATCH on triangular meshes

Fabrice Ardhuin

*The **friends of** WAVEWATCH III Team*
Marine Modeling and Analysis Branch
NOAA / NWS / NCEP / EMC

ardhuin@ifremer.fr
NCEP.list.waves@NOAA.gov

Covered in this lecture:

- Why triangles?
- Grid structure & running ww3_grid
- Time integration & grid optimization
- Numerical schemes: spatial advection
- Post-processing
- Triangles + squares in 2-way nested runs

What is not covered:

- Hands-on tutorial on grid generation
- for this : search for “tutorial” on Ifremer's wiki (using Polymesh)

https://forge.ifremer.fr/plugins/mediawiki/wiki/ww3/

Next course at UMD or at Ifremer!

Background material

- The basic numerics story is summarized in Aron Roland's Ph.D.
- Thesis (T. U. Darmstadt 2008). Code identical to WWM
- See also Roland (2012, ECMWF workshop proceedings)

- One paper with specific numerics (for coastal reflection)
- Ardhuin & Roland (JGR 2012)
- Some other papers with just application
- Ardhuin & al. (JPO 2009: Stokes drift; JPO 2012: wave-current)
- All these are at

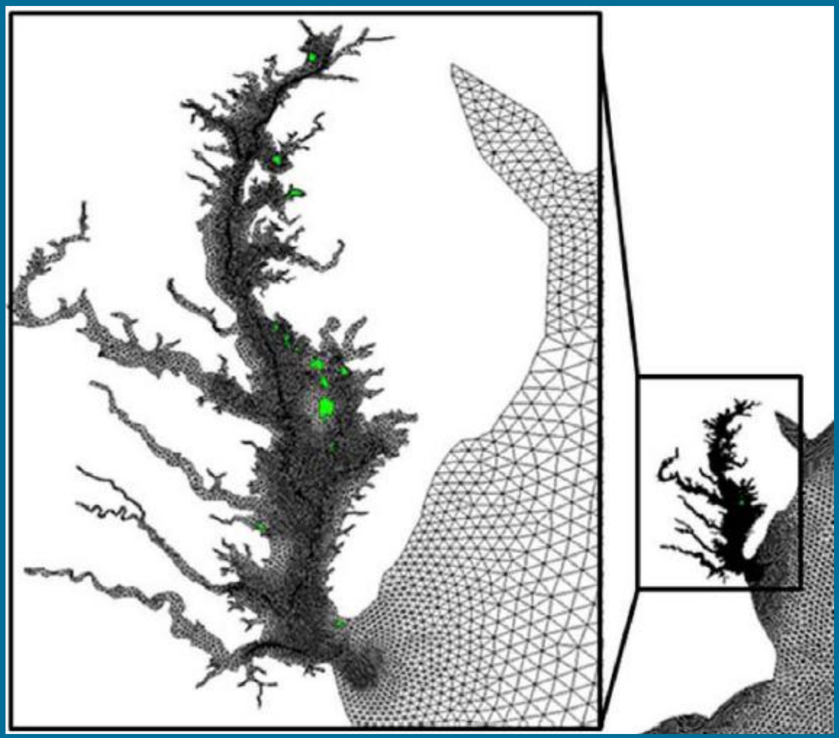
http://wwz.ifremer.fr/iowaga (just google IOWAGA)

- And you can find forecasts and hindcasts there

http://www.previmer.org/en/forecasts/waves

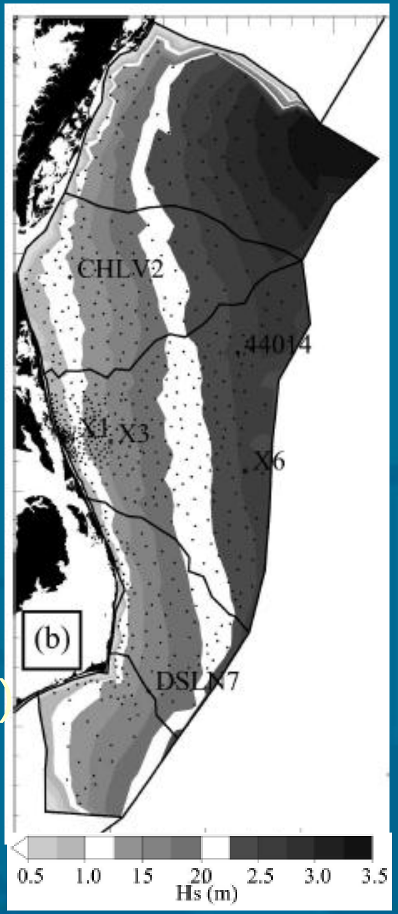
Minimizing number of nodes …

- 1) Because we really need high resolution in some places
- 2) Because we want to use crazy expensive physics



From Roland et al. (JGR 2012)

Extreme case:
only 900 nodes,
to use exact
non-linear interactions
(Ardhuin et al. JPO 2007)



And better shoreline orientation

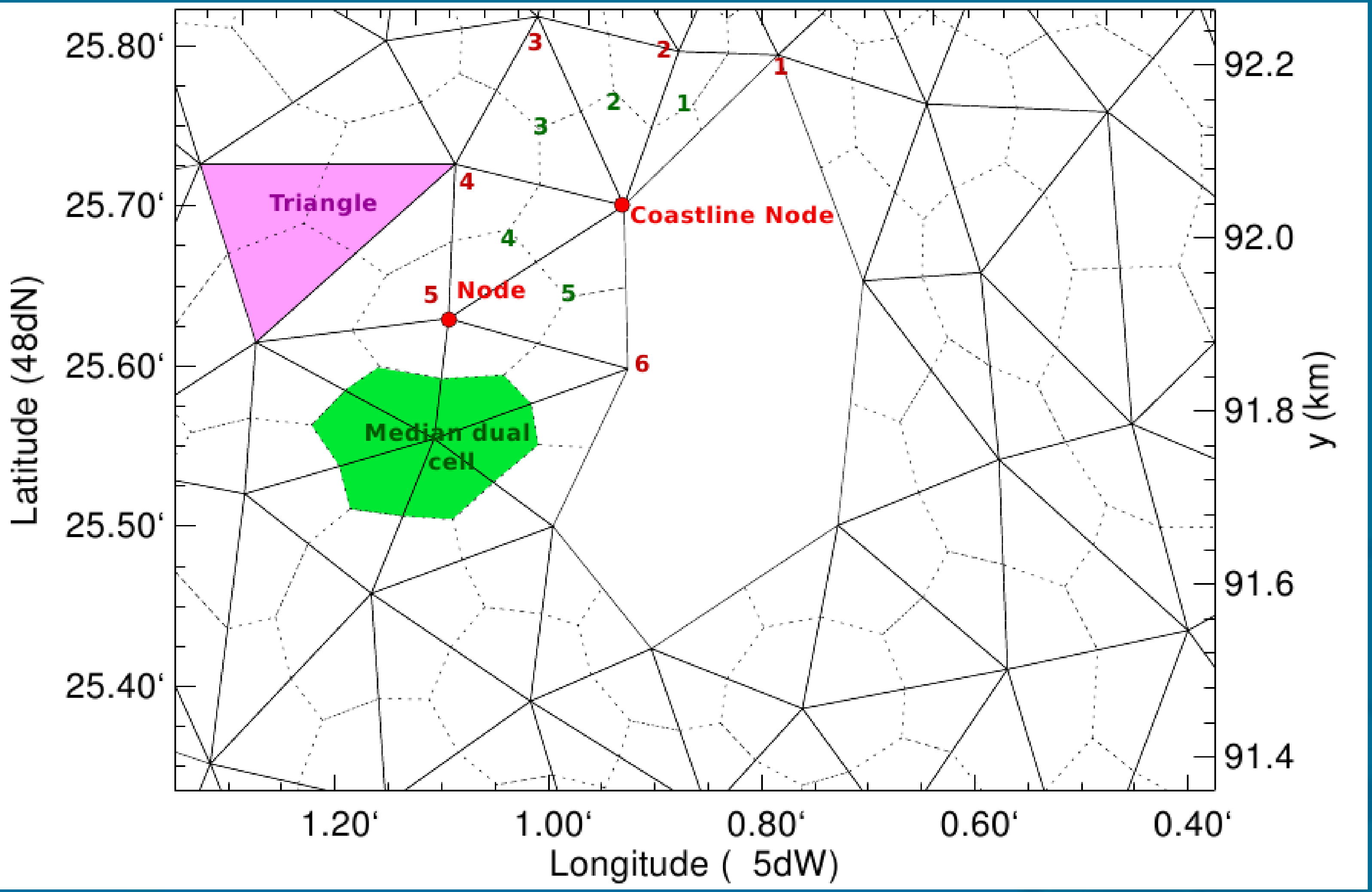
There are other alternatives

- 1) Nesting regular grids
- 2) Curviliear grids
- 3) Quad-trees or “SMC” grids
- 4) Hexagons …

- It is a matter of taste (and practicality and CPU time)…
- but I do not know how to help you with those. So let's go back to the simplest grid element: **a triangle**

The grid structure

Basic triangle stuff



The input file: built on “Gmsh” format

- 1) Nodes
- 2) Elements
  - - Active boudary points (element type 15)
  - - Triangles (element type 2)

- The input file was kept to a minimum, we did not include further info: neighbour lists...
- That extra info is recomputed when running ww3_grid and stored in mod_def.ww3
- … which can take minutes with more than 100K nodes !

- Internal storage in WWATCH is unchanged, we just have **NY = 1**
- **This carries into output files from ww3_outf & ww3_ounf**

Running ww3_grid for triangles

- Only a few things in **ww3_grid.inp** are different from other grids:

- 1) You have to tell ww3_grid that your grid is a triangle mesh:
- $ Define grid -- $
- 'UNST' T F *! NB: T T for a global grid... never tried yet!*

- 2) Define the depth threshold and scale factor + mesh file name
- 4.0 0.30 20 -1. 4 1 '(20f10.2)' 'NAME' 'hawaii_v5.msh'

```
$  4 Limiting bottom depth (m) to discriminate between land and sea
$    points, minimum water depth (m) as allowed in model, unit number
$    of file with bottom depths, scale factor for bottom depths (mult.),
$    IDLA, IDFM, format for formatted read, FROM and filename.
```

- **That's it! Nothing else**
- *NB: no need to change the switches …*

Defining list of input boundary points

OK, but listing the active input points (MAPSTA=2) is very cludgy!
You can do it … but you can also tell ww3_grid to figure it out.

- Otherwise, just add these two **namelist parameters** near the top of **ww3_grid.inp**

&UG UGOBCAUTO = T, UGOBCDEPTH = -20. ! or any other depth

- This means that ww3_grid will turn boundary points into **active** boundary points if the local depth (before water level added) is more than 20 m (yes, z is positive up in ww3_grid.inp)

Time integration

Because it is not so simple to define the proper advection **time step**, in the case of 'UNST' grid, the time step is **not set** to the value defined in ww3_grid.inp but instead it is **dynamically adjusted** (for explicit schemes). This time step will be different for each spectral component and will vary with current speed and water level.
So if your grid has **just one** very flat or very tiny triangle this time step could well be **0.1 s** (instead of 10 or 20 s) and the run grinds to a halt!
So how do I know about it?

Option 1) check the CFL numbers in the model output (not so easy)

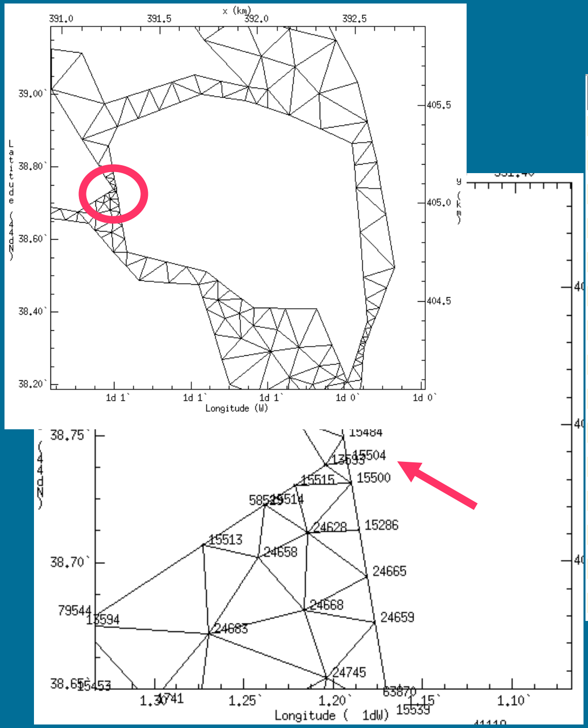
Option 2) – my choice - Run a version of the code compiled with the “T” switch (for test output). The screen output (or, in the future, some file fort.994) will list the “bad guys”, the nodes that cause the time step to be so small.

Time integration and grid optimization

Test output to find the bad guys

This test output comes from w3profsmd.ftn

```
!/T             DO IP = 1, NX
!/T               DTMAXEXP = SI(IP)/MAX(DBLE(10.E-10),KKSUM(IP))
!/T               IF (DTMAXEXP.LT.DTMAXGL*1.3.AND.IK.EQ.1) WRITE(6,'(A,3I8,2F8.3,3F10.4)') 'DTMAX:', IK, ITH, IP, &
!/T                          REAL(DTMAXEXP), REAL(DTMAXGL), XYB(IP,1), XYB(IP,2), XYB(IP,3)
!/T             END DO ! IP
```



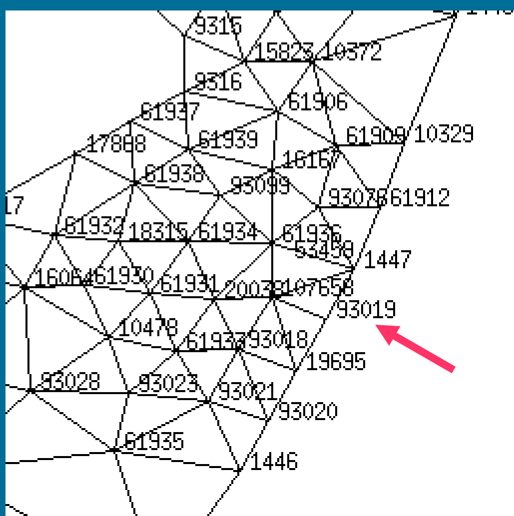
DTMAX:	1	1	15504	0.629	0.629	-1.0199	44.6457	0.0000
DTMAX:	1	2	15504	0.622	0.622	-1.0199	44.6457	0.0000
DTMAX:	1	3	15504	0.661	0.661	-1.0199	44.6457	0.0000
DTMAX:	1	4	15504	0.759	0.759	-1.0199	44.6457	0.0000
DTMAX:	1	5	15457	1.131	0.929	-2.3814	47.5000	-2.2883
DTMAX:	1	5	15504	0.929	0.929	-1.0199	44.6457	0.0000
DTMAX:	1	6	15504	0.990	0.990	-1.0199	44.6457	0.0000
DTMAX:	1	7	15449	1.354	1.142	-5.1328	48.4594	0.0000
DTMAX:	1	7	15469	1.408	1.142	0.1134	49.4831	-1.9901
DTMAX:	1	7	15504	1.142	1.142	-1.0199	44.6457	0.0000
DTMAX:	1	7	15508	1.438	1.142	-3.0863	48.8736	0.0000
DTMAX:	1	7	24622	1.444	1.142	-3.0862	48.8738	0.0000
DTMAX:	1	8	15449	1.105	1.105	-5.1328	48.4594	0.0000
DTMAX:	1	8	15469	1.415	1.105	0.1134	49.4831	-1.9901
DTMAX:	1	8	15508	1.308	1.105	-3.0863	48.8736	0.0000
DTMAX:	1	8	24622	1.304	1.105	-3.0862	48.8738	0.0000

Time integration and grid optimization

Test output to find the bad guys

This test output comes from w3profsmd.ftn

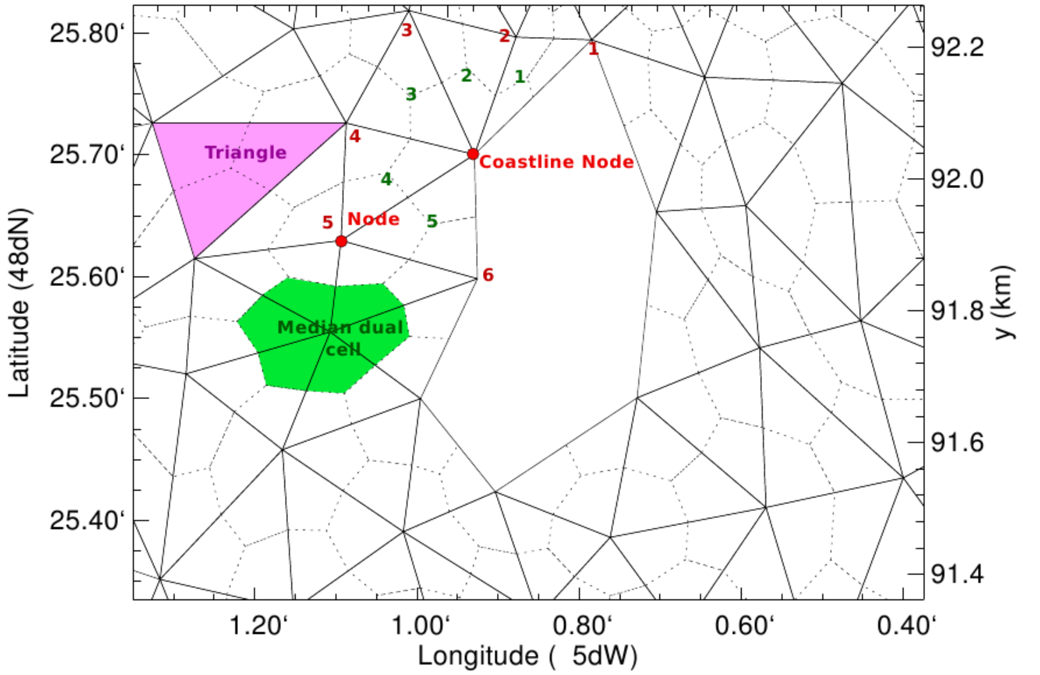
```
!/T            DO IP = 1, NX
!/T              DTMAXEXP = SI(IP)/MAX(DBLE(10.E-10),KKSUM(IP))
!/T              IF (DTMAXEXP.LT.DTMAXGL*1.3.AND.IK.EQ.1) WRITE(6,'(A,3I8,2F8.3,3F10.4)') 'DTMAX:', IK, ITH, IP, &
!/T                         REAL(DTMAXEXP), REAL(DTMAXGL), XYB(IP,1), XYB(IP,2), XYB(IP,3)
!/T            END DO ! IP
```




```
 WAVEWATCH III calculating for 2011/01/28 00:01:00 UTC at 11:41:01
DTMAX:        1       1   19695   8.311   7.168   -1.2241   44.5794   -0.0441
DTMAX:        1       1   22116   7.790   7.168   -1.2454   44.5558   -2.5164
DTMAX:        1       1   42848   9.238   7.168   -1.7780   48.7403   25.3937
DTMAX:        1       1   62295   9.114   7.168   -1.8370   48.7218   21.5274
DTMAX:        1       1   65564   8.305   7.168   -2.5881   47.0962   60.7536
DTMAX:        1       1   65696   8.437   7.168   -2.6156   47.0895   45.6327
DTMAX:        1       1   67532   8.648   7.168   -2.5328   47.1141   56.2737
DTMAX:        1       1   72761   8.670   7.168   -1.8543   48.7613   28.4829
DTMAX:        1       1   72762   9.115   7.168   -1.8465   48.7591   28.2976
DTMAX:        1       1   93019   7.168   7.168   -1.2230   44.5808   -0.3705
DTMAX:        1       1   93083   9.155   7.168   -1.2444   44.5650   11.5519
DTMAX:        1       1   93321   9.108   7.168   -2.5852   47.0968   61.3314
DTMAX:        1       1   93324   9.243   7.168   -2.5423   47.1108   47.0504
DTMAX:        1       1   93449   9.171   7.168   -1.8817   48.7248   22.4942
DTMAX:        1       1   93546   9.003   7.168   -2.5860   47.1011   44.4219
DTMAX:        1       1   93563   8.524   7.168   -2.6029   47.0885   53.5324
DTMAX:        1       1   93677   8.874   7.168   -2.5352   47.1151   58.8437
DTMAX:        1       1   93681   8.967   7.168   -1.7322   48.7123   18.7894
```

I know what you think … this ought to be easier. This is why Aron Roland is trying to get the ww3_shel interacting with Polymesh via a GUI

General principles : Contour Residual Distribution



Advection equation :

$$\frac{\partial N}{\partial t} + \nabla_X \left(\boldsymbol{c}_X N \right) = 0$$

Discrete form :

$$\int_T \frac{\partial N}{\partial t} dA = -\int_T \boldsymbol{c}_X \nabla N dA = \Phi_T$$

Update of spectrum = sum on dual cell

$$N_i^{n+1} = N_i^n + \frac{\Delta t}{S_i} \sum_{T, i \in D_i} \Phi_{i,T}$$

4 schemes implemented :

1) « Narrow sencil » scheme (N) : EXPFSN (Csik et al. 2002, Roland 2008), CPU cost of full model : 12

2) « Positive Streamline Invariant » EXPFSPSI (Abgrall 2001)
CPU cost of full model :15

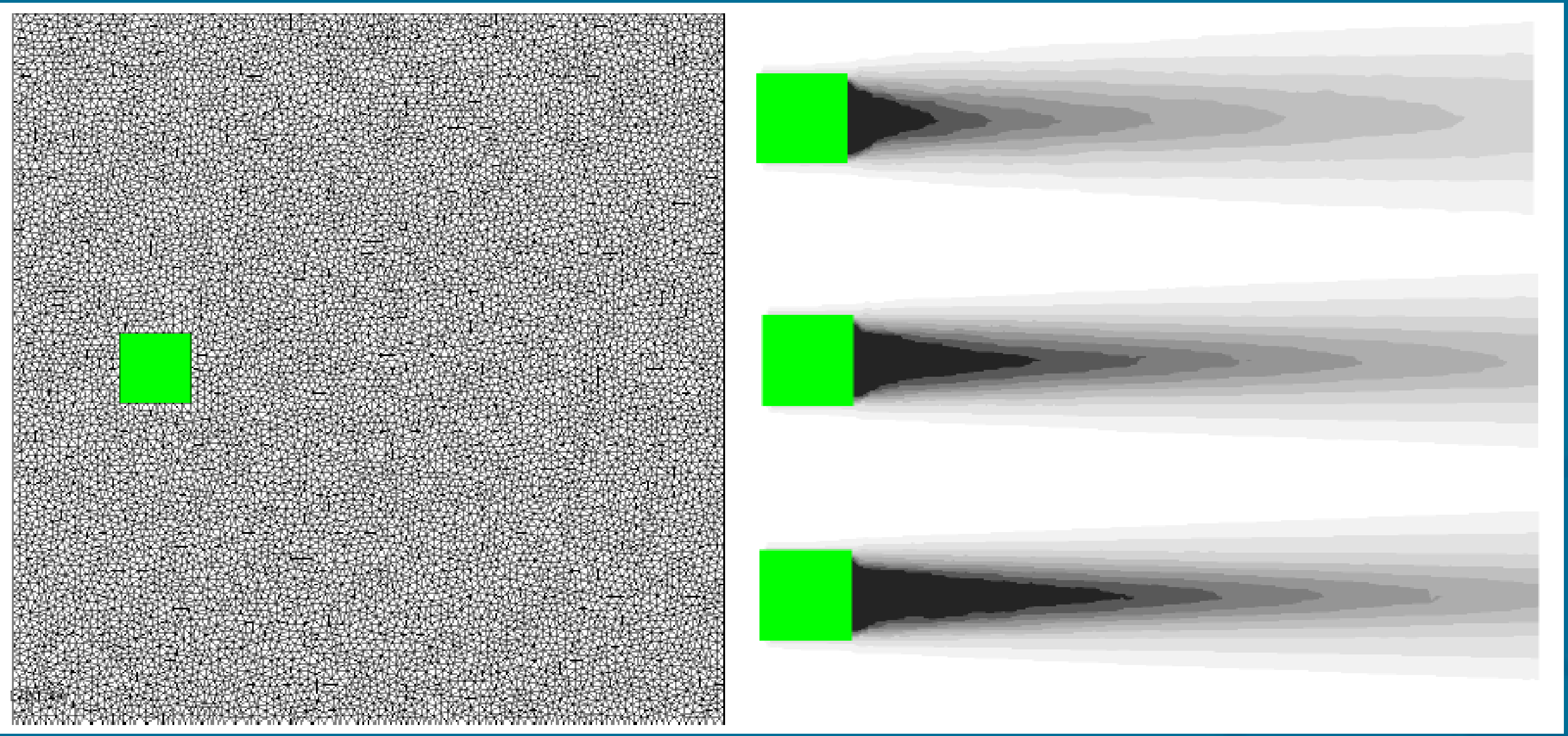
3) « Flux Corrected Transport » : EXPFSFCT (Csik et al. 2002, Roland 2008), CPU cost of full model : 29
(Lax-Wendroff kind of scheme)

4) Implicit N scheme : IMPFSIMP

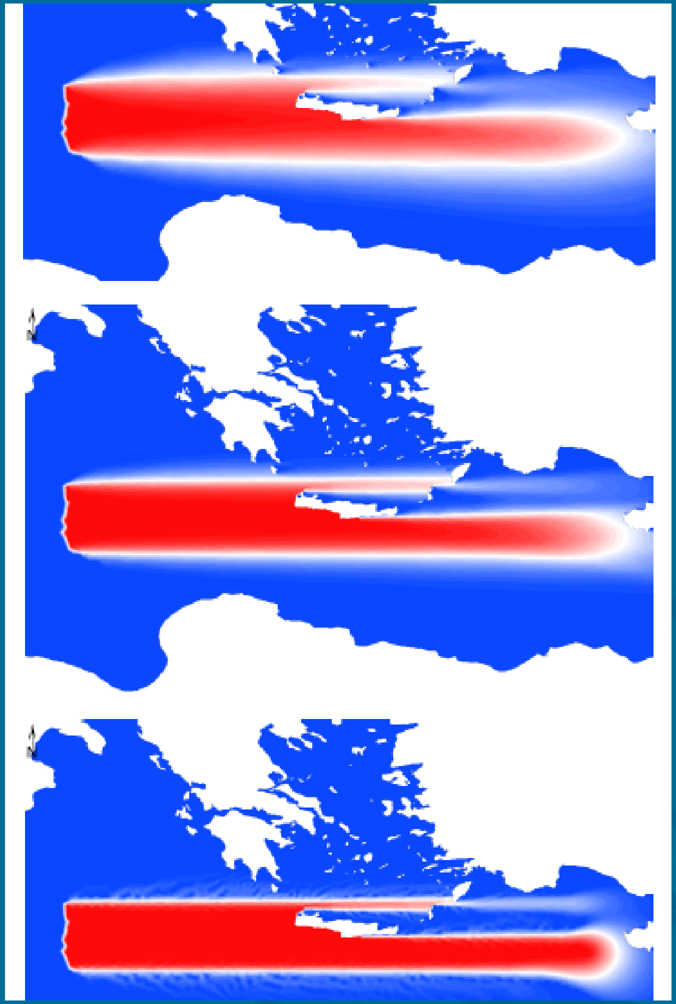
Higher order = more expensive and less numerical diffusion

Numerical schemes

Higher order = more expensive and less numerical diffusion



Higher order = more expensive and less numerical diffusion



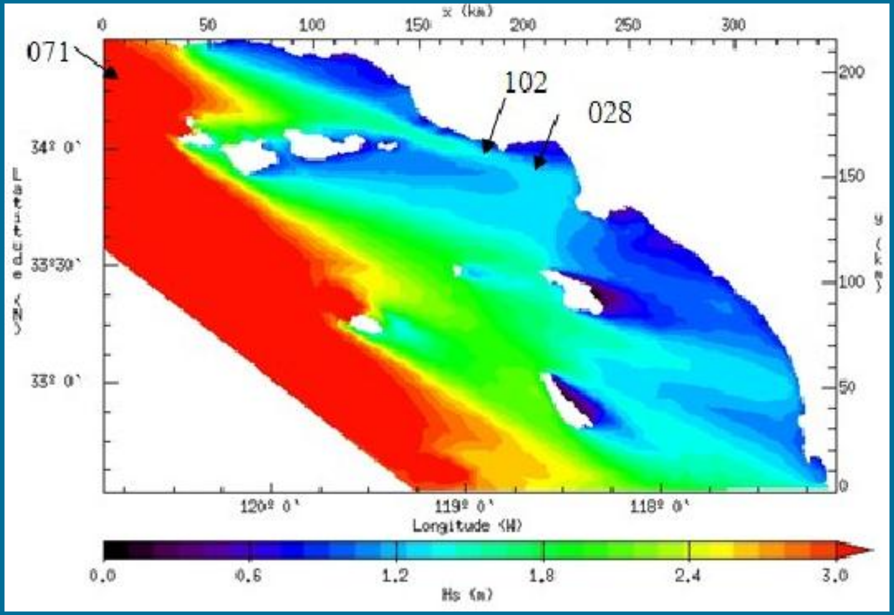
N

PSI

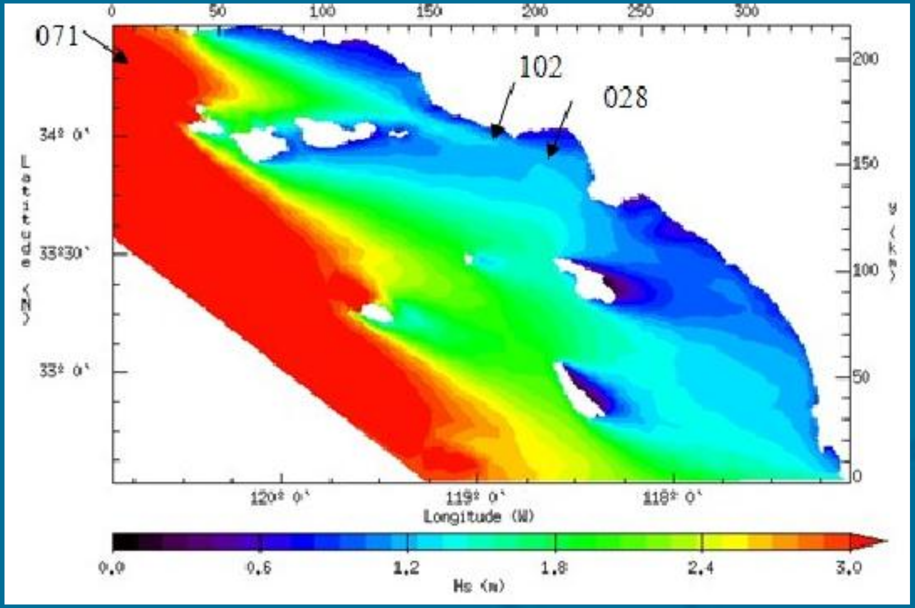
FCT

Higher order = more expensive and less numerical diffusion

PSI



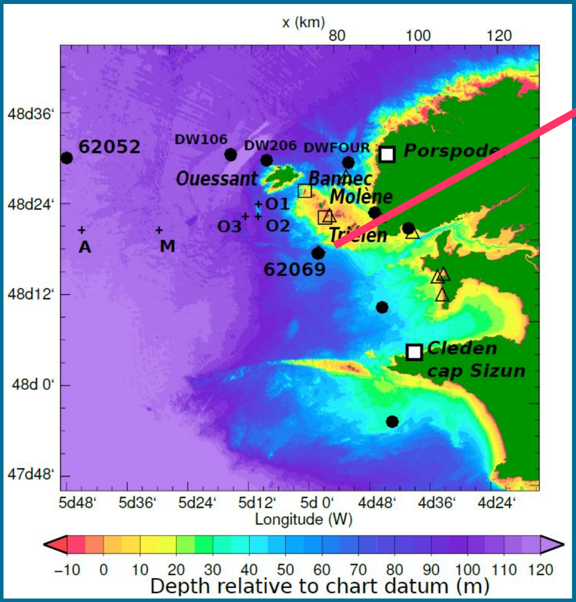
N

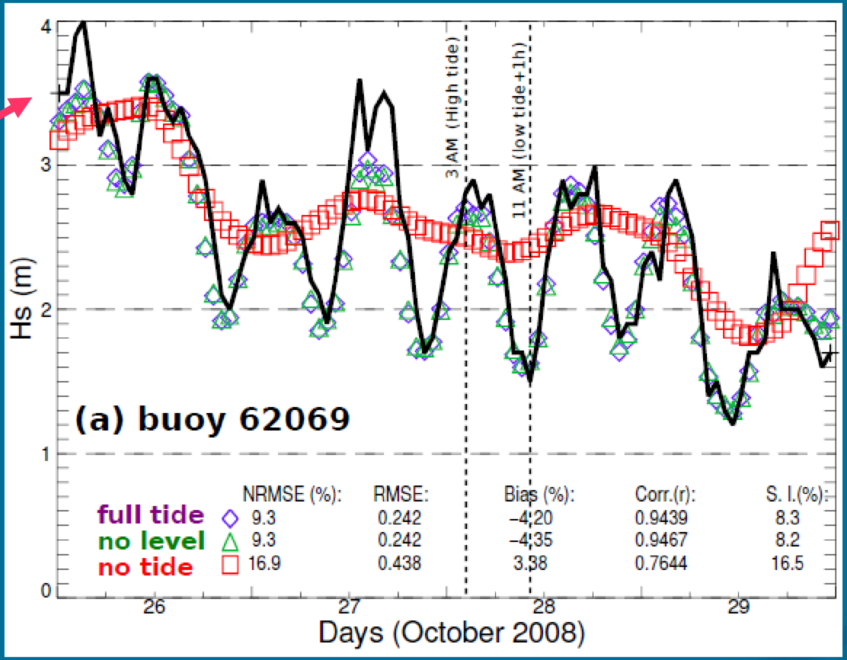


Example applications

with currents & water levels

Ardhuin & al. (2012) :





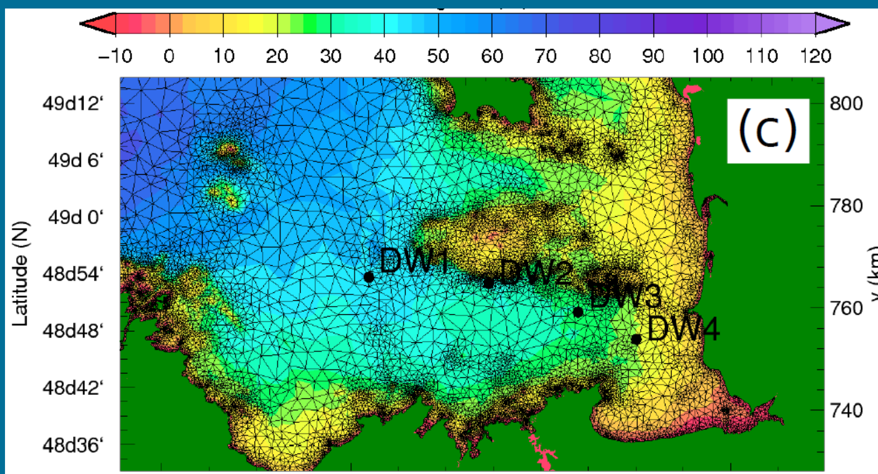
Important code change : « refraction filter » on total refraction (now PR3 only)

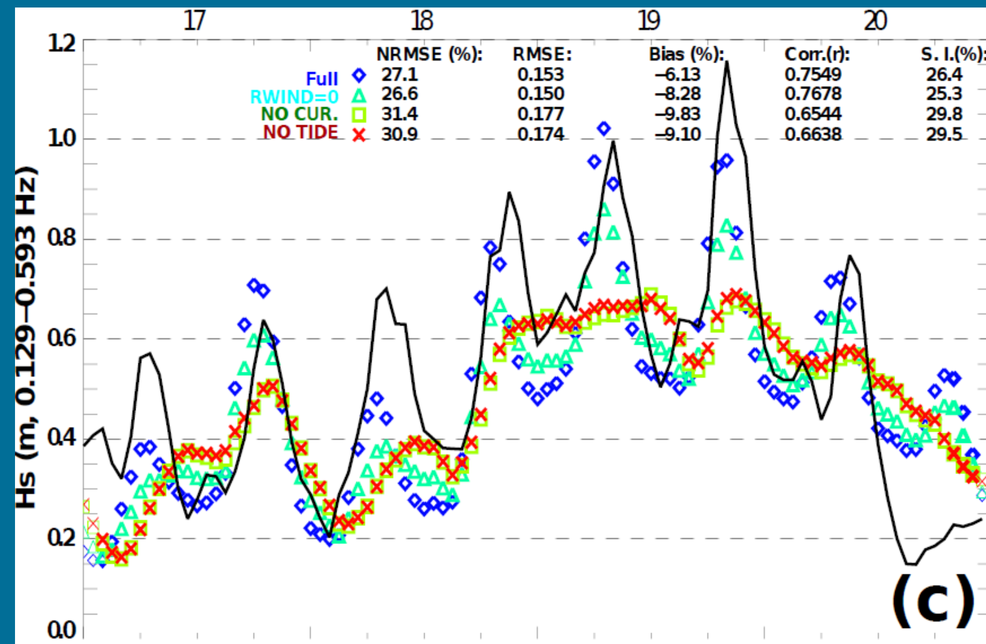
In the pipeline :

- use of tidal constituents (saves disk space!!) : OK in tide branch

with currents & water levels

Ardhuin & al. (2012) :

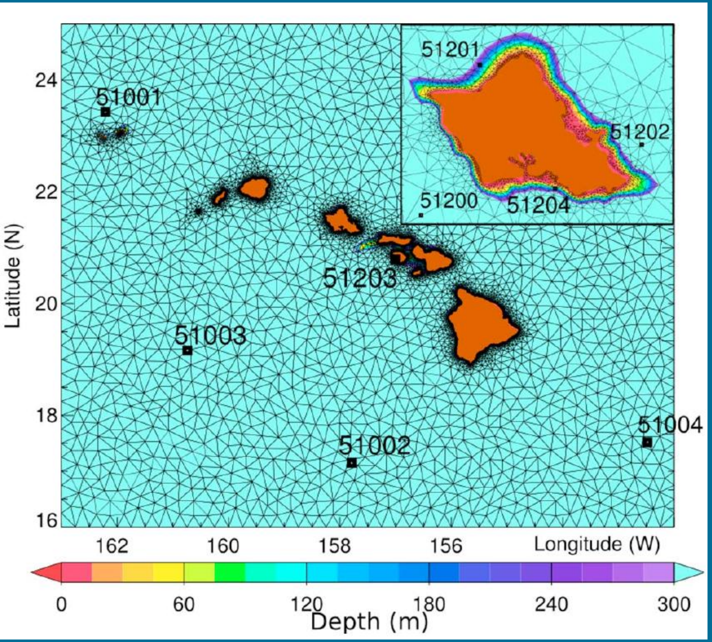


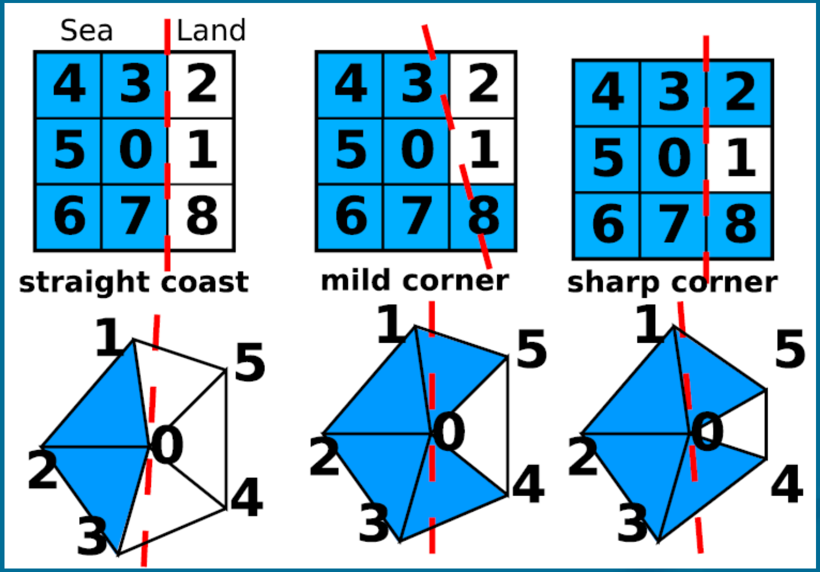


Important code change : « refraction filter » on total refraction
(put in March 2011, v. 4.04, now PR3 only)
In the pipeline :
- use of tidal constituents (saves disk space!!) : OK in tide branch

Coastal reflection : Hawaii grid used in tutorial

Ardhuin & Roland (JGR 2012) :

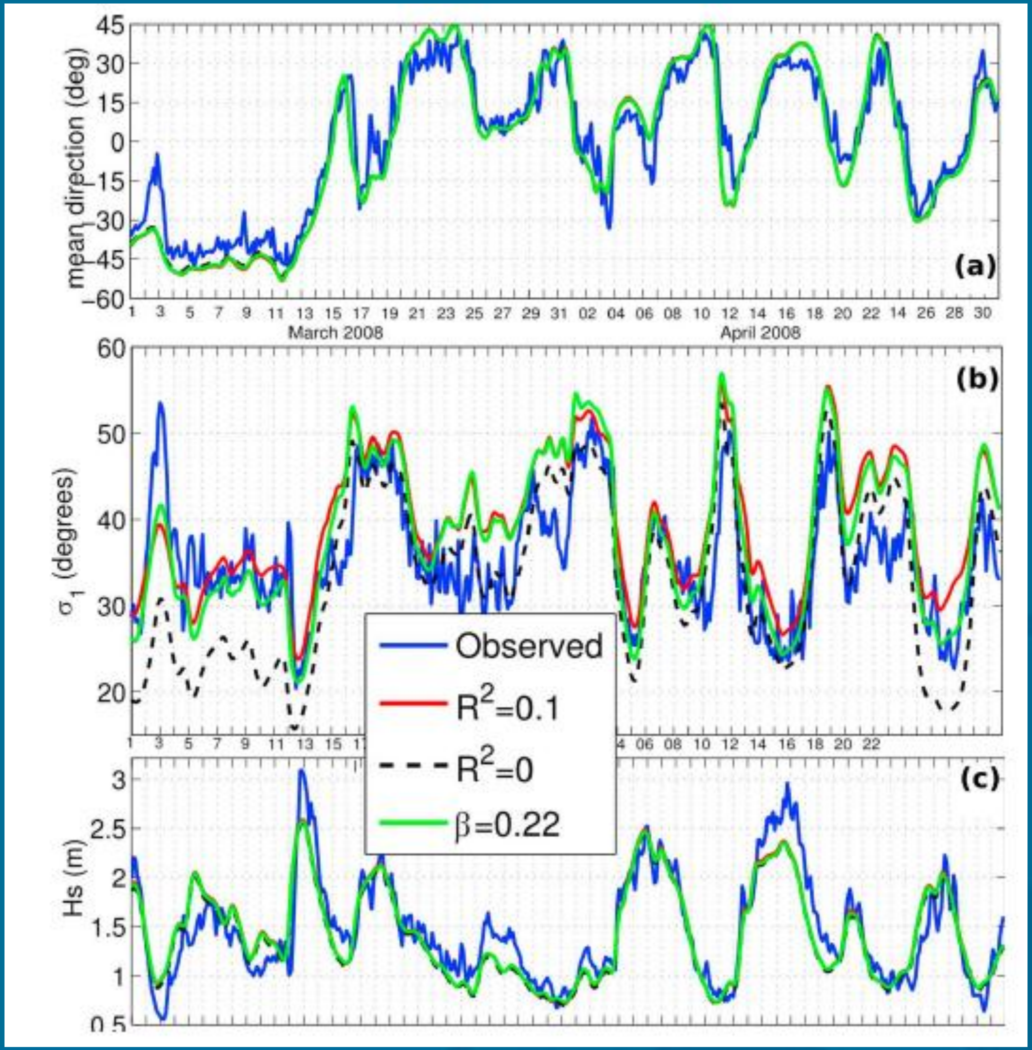




Shoreline orientation is easier to define : reflection AT shoreline nodes
Different with regular grid : shoreline BETWEEN nodes

Coastal reflection : Hawaii grid used in tutorial

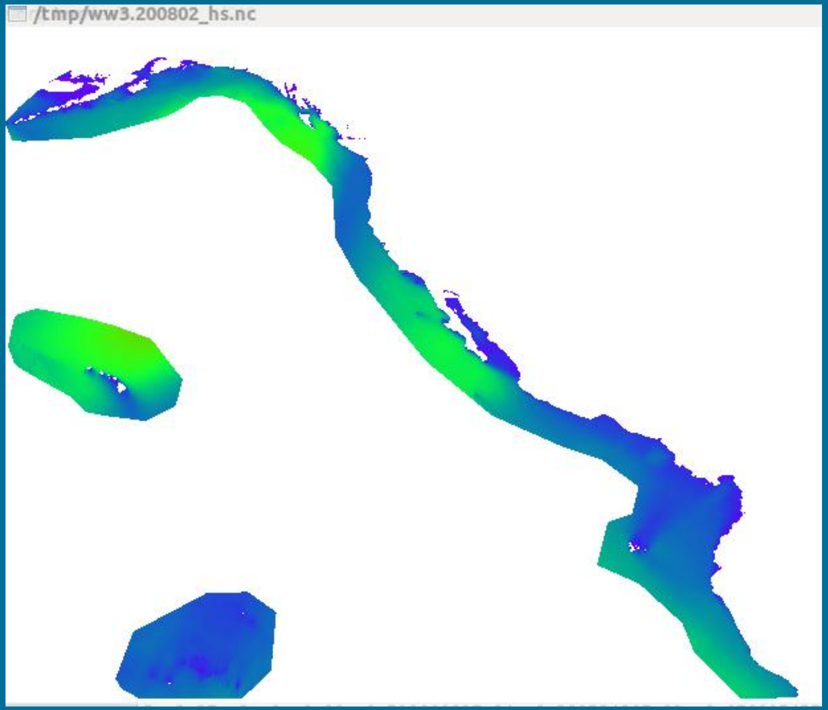
validation at Waimea buoy 51201

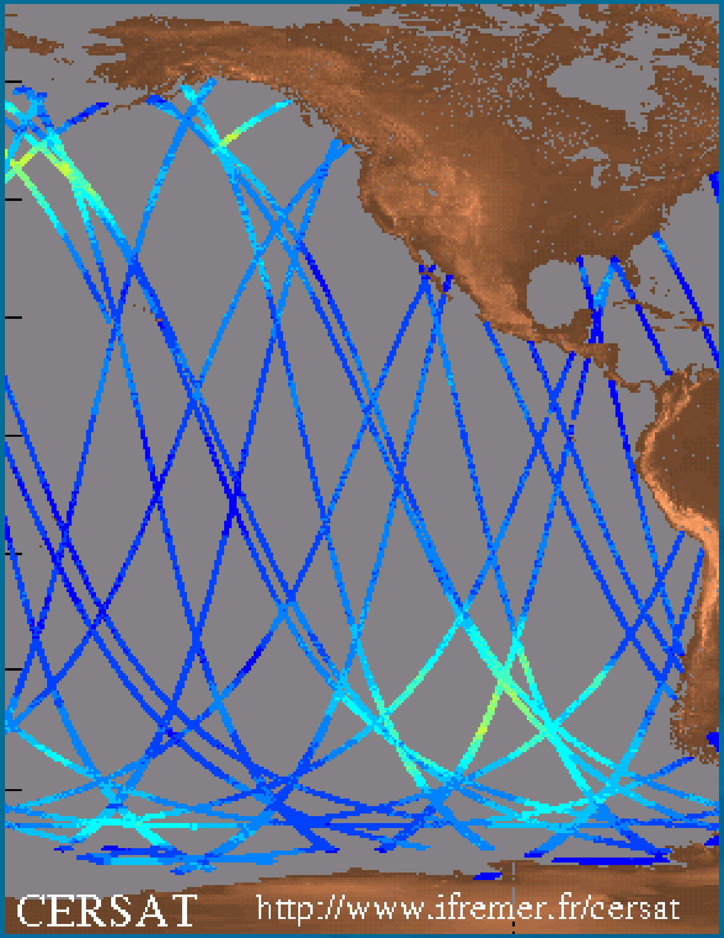


Hawaii grid used in tutorial

Boundary conditions from Ifremer's hindcast

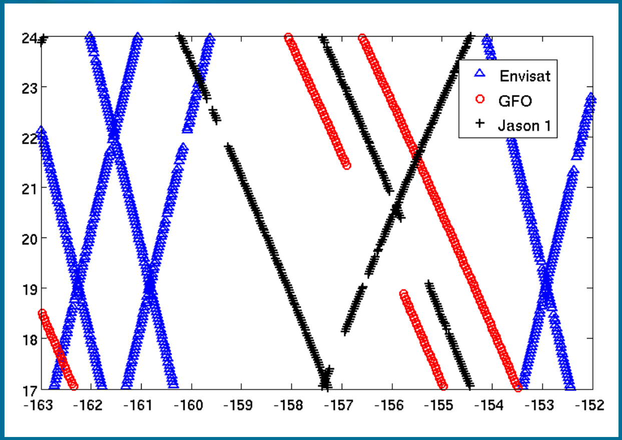
http://tinyurl.com/iowagaftp/HINDCAST/GLOBAL/2008_ECMWF/

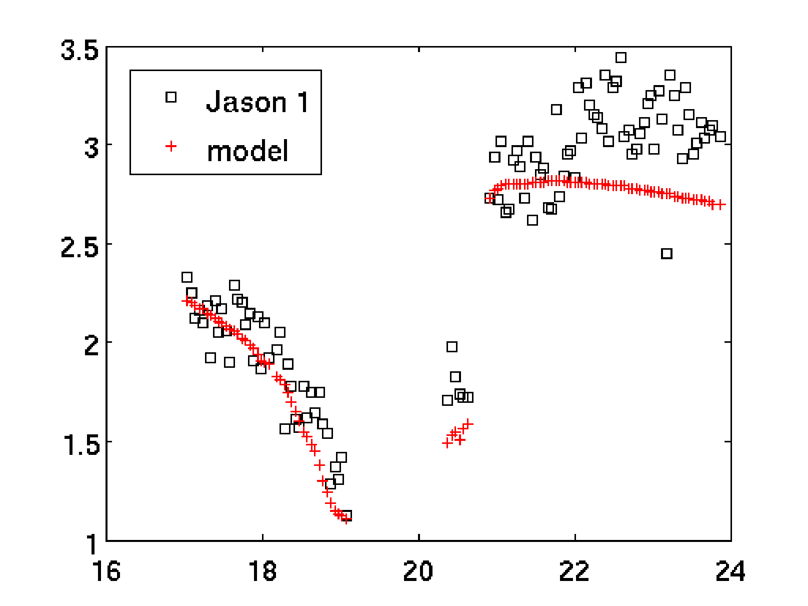


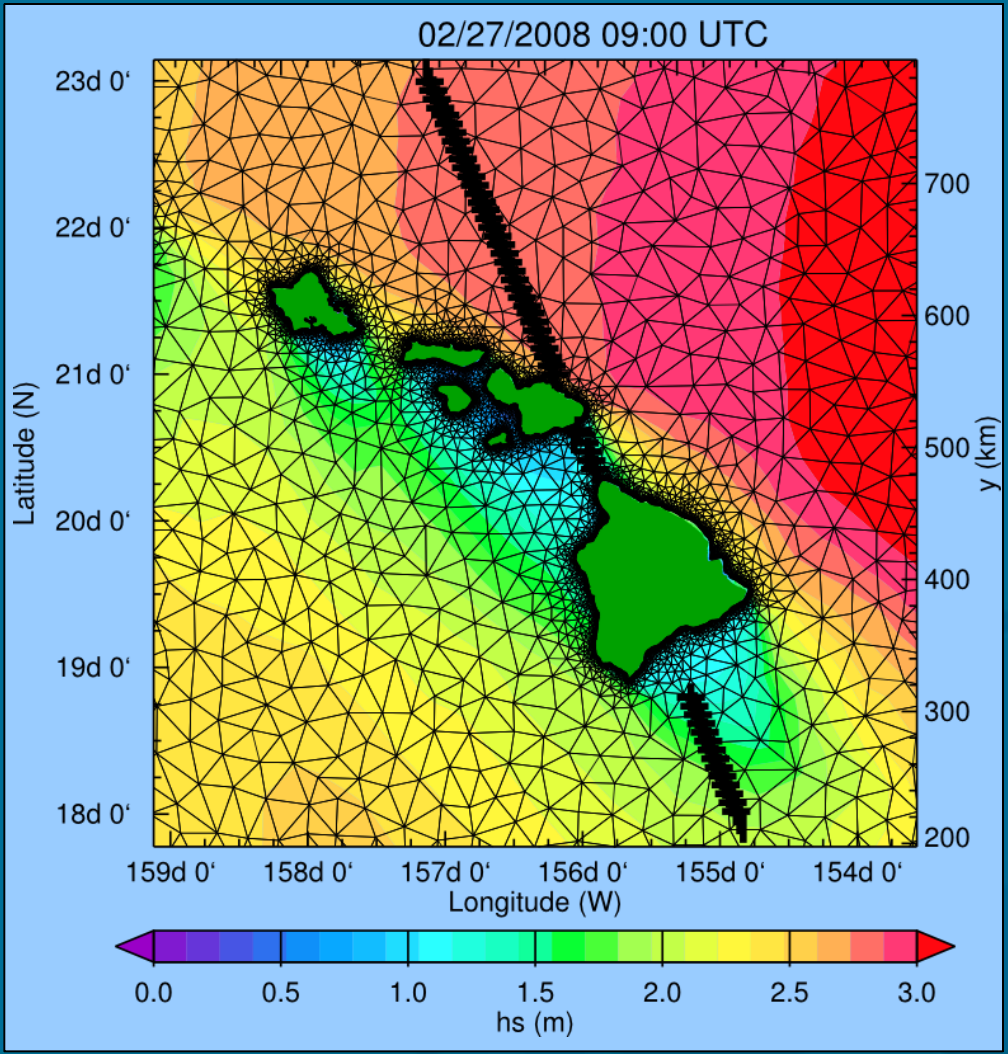


Example applications

Hawaii grid used in tutorial







- Advection schemes in WWATCH taken straight from WWM II

→ several validation and comparison study

→ 2 year of routine forecasts

→ 20 year hindcast for test area, now expanding to full France

- 4 different schemes to chose from. Personally the N scheme works great. If you are a daredevil, maybe the IMP is for you