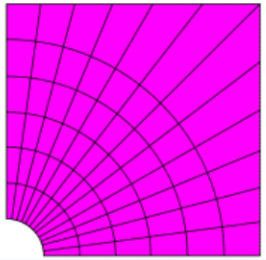
Tutorial 02:
Create MY CONFIGURATION

Numerical Aspects: Horizontal discretization

- **Structured grids**

The grid cells have the same number of sides.



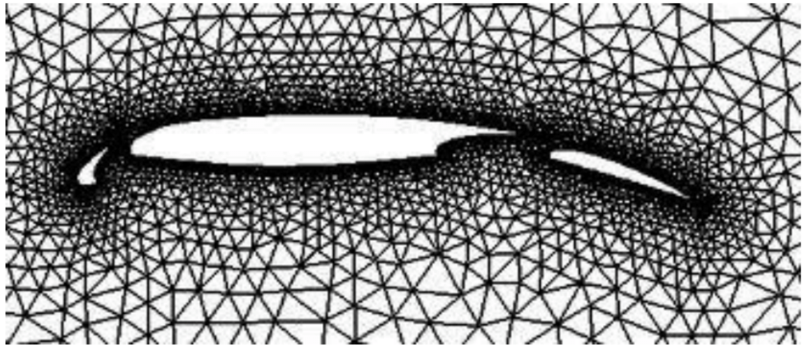
CROCO

- **Unstructured grids**

The domain is tiled using more general geometrical shapes (triangles, ...) pieced together to optimally fit details of the geometry.

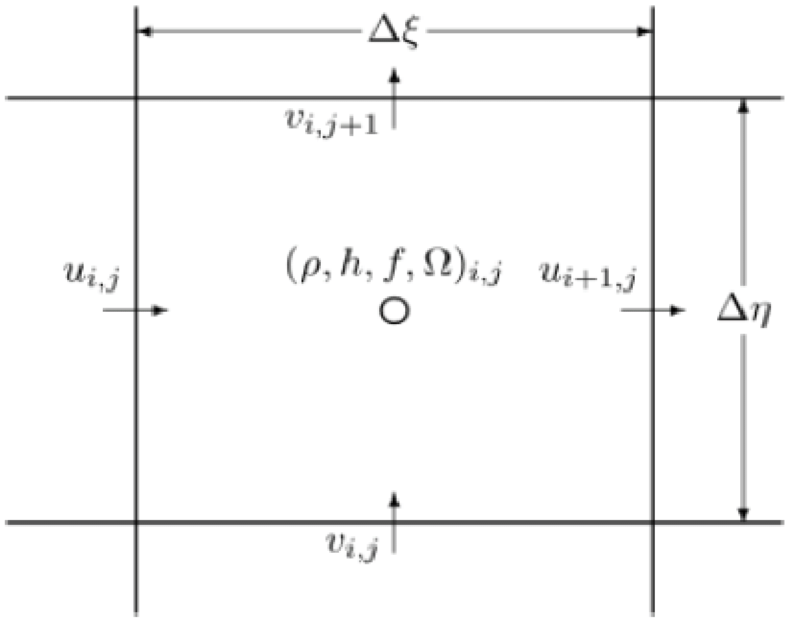
✓Good for tidal modeling, engineering applications.

✓Problems:
geostrophic balance accuracy,
wave scattering by non-uniform grids, conservation and positivity properties, ...



Numerical Aspects: Horizontal discretization

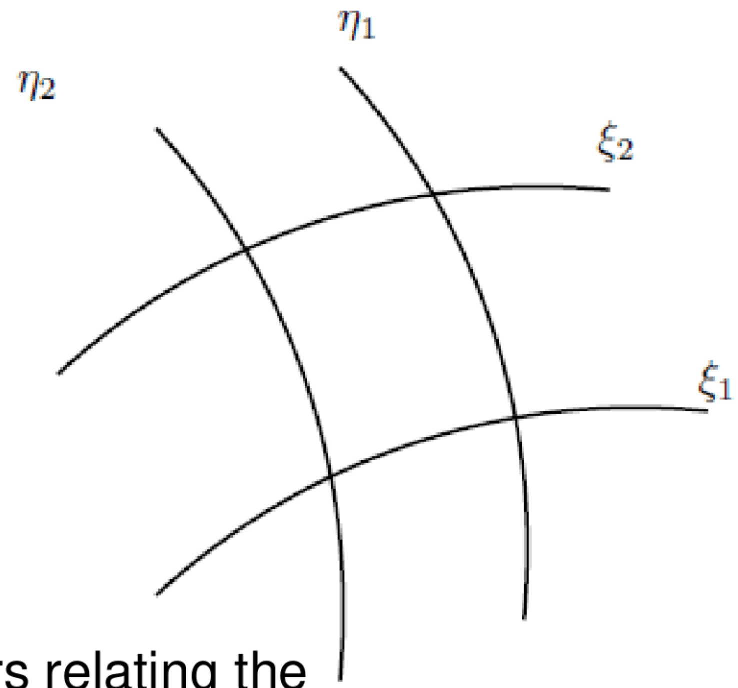
CROCO grid is discretized in **coastline-** and **terrain-following curvilinear coordinates** with **free-surface**, on an **Arakawa-C grid**.



(a) Grille C d'Arakawa

$$(ds)_\xi = \left(\frac{1}{m}\right) d\xi$$

$$(ds)_\eta = \left(\frac{1}{n}\right) d\eta$$



m, n: scale factors relating the differential distances to the physical arc lengths

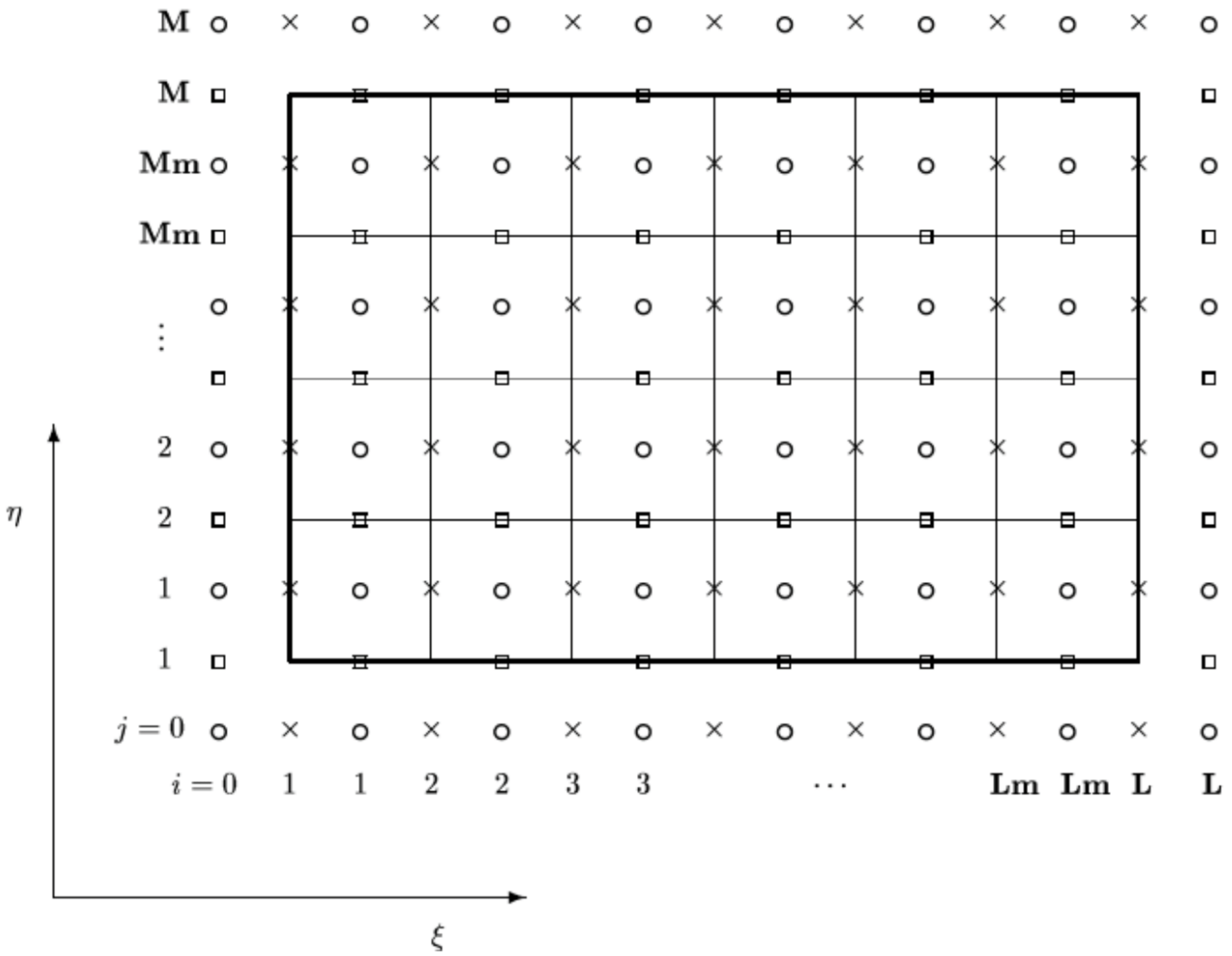
Numerical Aspects: Horizontal discretization

CROCO horizontal grid

$\times$ – u points

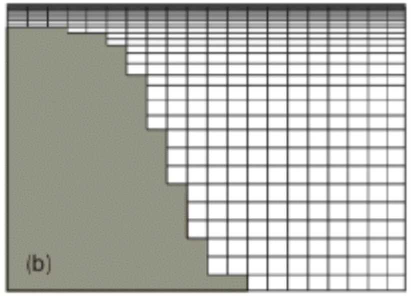
$\square$ – v points

$\circ$ – ρ points



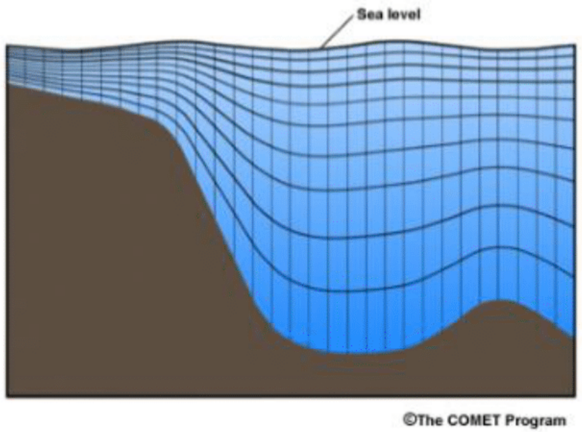
Numerical Aspects: Vertical discretization

- **Z-Coordinates**



- **Sigma (and stretched) Coordinates**

Terrain-following curvilinear coordinates with free-surface



CROCO

free surface

Vertical sigma coord.

Water column thikness

$$z(x, y, \sigma, t) = \eta(1+ \sigma) + hc .\sigma + (h - hc).C(\sigma)$$

Parameters (thickness) controlling the stretching

Stretching function
C=f(θs,θb)

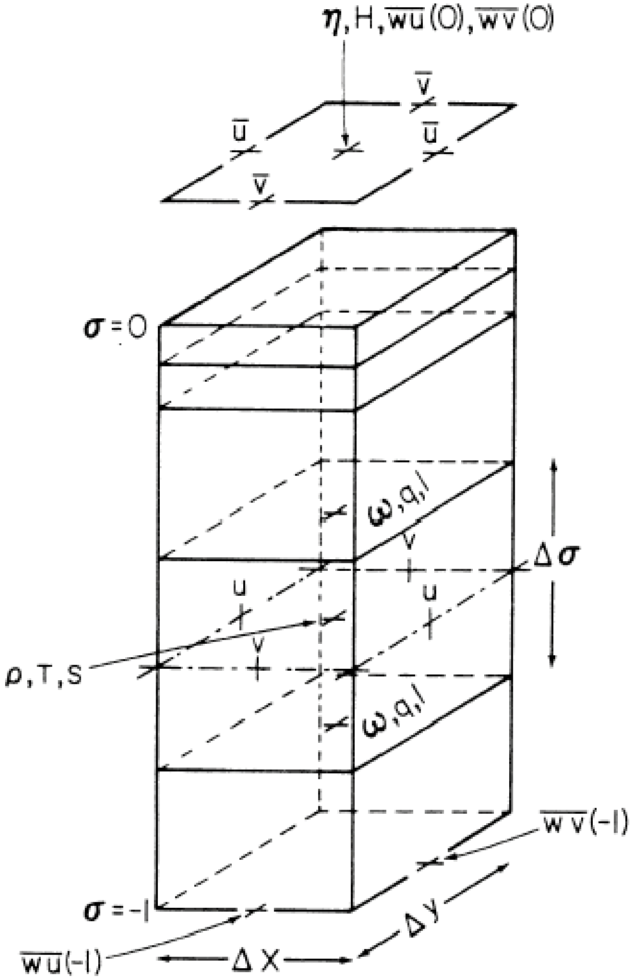


Fig. 1. The locations of the variables on the finite difference grid.

OBJECTIVES

- Get familiar with **CROCO and CROCO_TOOLS directories**.
- Run MATLAB and pre-processing tools for the 1st experiment.
- Modify and run MATLAB scripts.
- Create the horizontal and vertical grids.
- Make your own grid and configuration.

STEP 1: Logging onto the HPC cluster

- From a terminal/konsole:

```
ssh -X login@lengau.chpc.ac.za
```

- Reserve one interactive processor to do this pre-processing step

```
[login@login2 ~]$ qsubi1
[login@cnode0220 ~]$
```

- Go directly into your `lustre` directory:

```
[login@cnode0220 ~]$ cd lustre/CROCO
[login@cnode0220 CROCO]$ ls
croco-v2.0.1 croco_tools-v2.0.0
[login@cnode0220 CROCO]$
```

NODES

SAEON NRF
nicis CHPC
IRD Institut de Recherche pour le Développement FRANCE
cnrs
LEGOS
CROCO Sud Coastal and Regional Ocean COmmunity model
SHOM
LOCEAN
Universidad de Concepción
DGEO Universidad de Concepción

STEP 2: Creating the work directory of your first CROCO

- From croco-v2.0.1 directory, Edit the file **create_config.bash** using the Linux command **vi** or the **nedit** software
  - line 68: replace the string "**Run**" by "Run_BENGUELA_LR"

- Execute the new file **create_config.bash**

```
[login@cnode0220 croco-v2.0.1]$ ./create_config_bash
```

NODES

- Go to the new directory created: Run_BENGUELA_LR.

STEP 3: Recreating the BENGUELA_LR Grid

- Run the basic pre-processing steps to create the grid:

```
[login@cnode0220 Run_BENGUELA_LR]$ matlab -nodesktop
```

```
                          < M A T L A B (R) >
                  Copyright 1984-2020 The MathWorks, Inc.
              R2020a Update 8 (9.8.0.1873465) 64-bit (glnxa64)
                             February 3, 2022

To get started, type doc.
For product information, visit www.mathworks.com.
>> start
>> edit crocotools_param
>> make_grid
```

- Take a look at the new files in the CROCO_FILES directory
- Inspect the vertical grid and play with the parameters using the Matlab function:

```
>> draw_zonal_section(N,theta_s,theta_b,hc,vtransform,lat_index)
```

Vertical grid: Parameters θs,θb

- Formulation Vtransform=1

$$z(x,y,\sigma,t) = z_0(x,y,\sigma) + \zeta(x,y,t)\left[1 + \frac{z_0(x,y,\sigma)}{h(x,y)}\right]$$

$$z_0(x,y,\sigma) = h_c\sigma + [h(x,y) - h_c]Cs(\sigma)$$

$$Cs(\sigma) = (1-\theta_b)\frac{sinh(\theta_s\,\sigma)}{N} + \theta_b\left[\frac{0.5\,tanh\,((\sigma+0.5)\theta_s)}{tanh(0.5\,\theta_s)} - 0.5\right]$$

- Formulation Vtransform=2

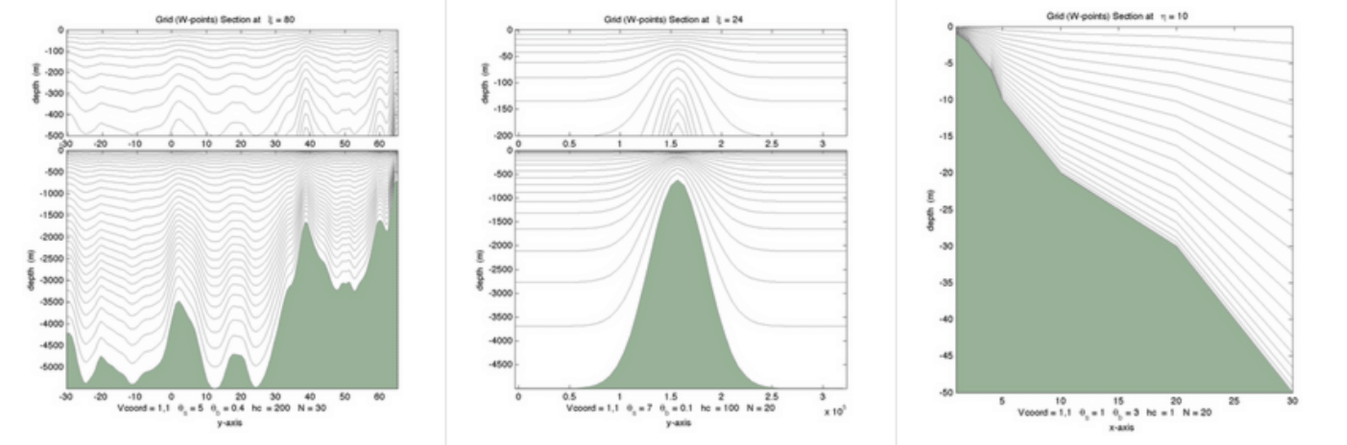
$$z(x,y,\sigma,t) = \zeta(x,y,\sigma) + [\zeta(x,y,t) + h(x,y)]z_0(x,y,\sigma)$$

$$z_0(x,y,\sigma) = \frac{h_c\sigma + h(x,y)Cs(\sigma)}{h_c + h(x,y)}$$

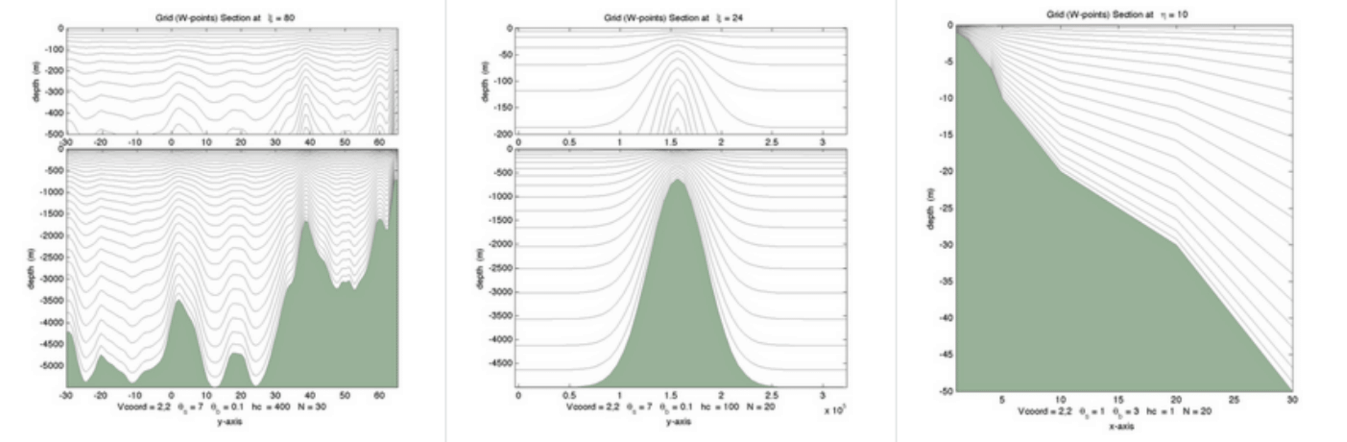
$$sc = \frac{\sigma - N}{N}$$

$$csf = \frac{1. - cosh(\theta_s\,sc)}{cosh(\theta_s) - 1.} \quad \text{if } \theta_b > 0, \quad csf = -sc^2 \quad \text{otherwise}$$

$$Cs(\sigma) = \frac{e^{\theta_b csf} - 1.}{1. - e^{-\theta_b}} \quad \text{if } \theta_s > 0, \quad Cs(\sigma) = csf \quad \text{otherwise}$$



Vtransform=1, $\theta_S = 7.0, \theta_B = 0.1$ | Vtransform=1, $\theta_S = 7.0, \theta_B = 1.0$ | Vtransform=1, $\theta_S = 7.0, \theta_B = 3.0$



Vtransform =1, $\theta_S = 7.0, \theta_B = 1.0$ | Vtransform=2, $\theta_S = 7.0, \theta_B = 1.0$ | Vtransform=2, $\theta_S = 7.0, \theta_B = 3.0$

Coastal and Regional Ocean COmmunity model

STEP 4: Creating your CROCO Grid

- Re-do **Steps 2 & 3** with your own parameters:
  Edit the file `create_config.bash`

- Call your configuration **`Run_Clim`**.
  - ➥ You can play with the grid rotation and the editmask.

Special attention to:

- **grid size (LLm, MMm): do not exceed a 100x100 grid**
- **at least 5°x5°**
- **do not create boundaries of less than 5 points**
- **do not overlap the equatorial zone by less than 2°**

STEP 5: Exiting

- Exit Matlab:

```
exit
```

- Give back the compute node:

```
exit
```

- Logoff the Lengau cluster:

```
exit
```