

Introduction to Coastal and Regional Ocean Modeling With CROCO

Theory and Hands-on Practice

Speaker

Dr. K M Azam Chowdhury
Director
International Center for Ocean Governance
University of Dhaka

Presentation Outline

Module 1

Motivations for Ocean Modeling

Module 2

Types of Ocean Model

Module 3

Fundamentals of Ocean Model

Module 4

Introduction to CROCO Model

Module 5

CROCO Model Configuration and Input

Module 6

Simulation, Post-processing and Validation

Overview

A structured learning pathway from ocean modeling fundamentals to practical CROCO applications

Theory



Ocean Modelling

Role and motivation of numerical ocean models



Types of Ocean Models

Model classifications and major modeling systems



$$\nabla \cdot u = 0$$

Numerical Fundamentals

Governing equations and numerical methods



Introduction to CROCO

Model architecture, features, and applications

Hands-on



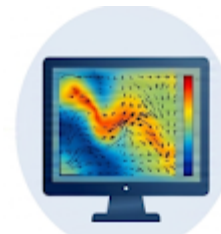
Configuration

Domain setup, grid generation, and input preparation



Simulation

Model execution and computational workflow



Analysis

Post-processing and visualization of model outputs



Validation

Performance evaluation using observational datasets

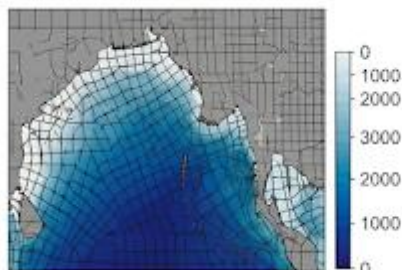
Computational Workflow

1. Study Domain



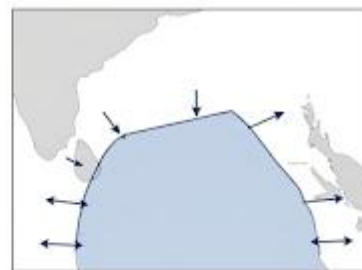
Define the study region and modeling objectives.

2. Grid & Bathymetry



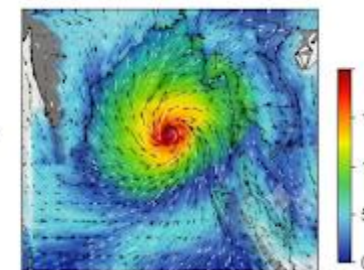
Generate the computational grid and prepare bathymetry.

3. Initial & Boundary Conditions



Specify initial conditions and open boundary forcing.

4. Atmospheric Forcing



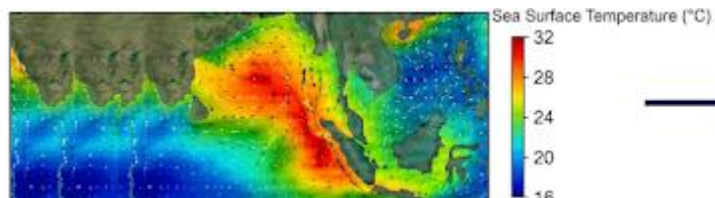
Apply surface forcing (wind, heat, and freshwater fluxes).

5. Run CROCO Simulation



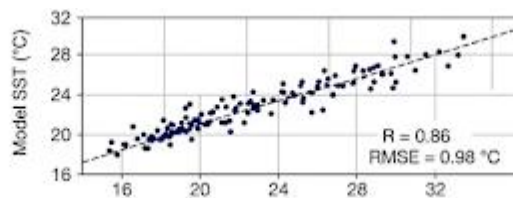
Execute the numerical simulation.

6. Post-processing & Visualization



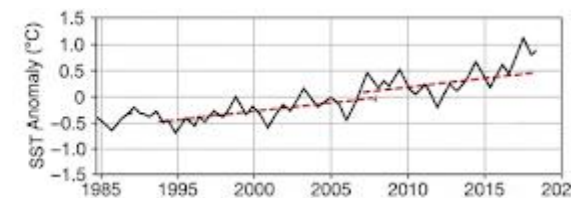
Analyze and visualize model outputs.

7. Validation



Compare simulations with observational datasets.

8. Scientific Interpretation



Interpret results to address scientific objectives.

Expected Learning Outcomes

From theoretical foundations to practical ocean model implementation



Scientific Understanding

- Fundamental principles of numerical ocean modeling and regional ocean dynamics
- Ocean model classifications, numerical frameworks, and scientific applications



Model Configuration

- Configuration of the CROCO modeling system for regional and coastal applications
- Preparation of computational grids, bathymetry, atmospheric forcing, and boundary conditions



Simulation & Analysis

- Execution of numerical simulations and analysis of model outputs
- Post-processing, visualization, and interpretation of oceanographic variables



Model Evaluation

- Validation of model simulations using in situ and satellite observations
- Assessment of model performance using statistical metrics and scientific interpretation

Module 01

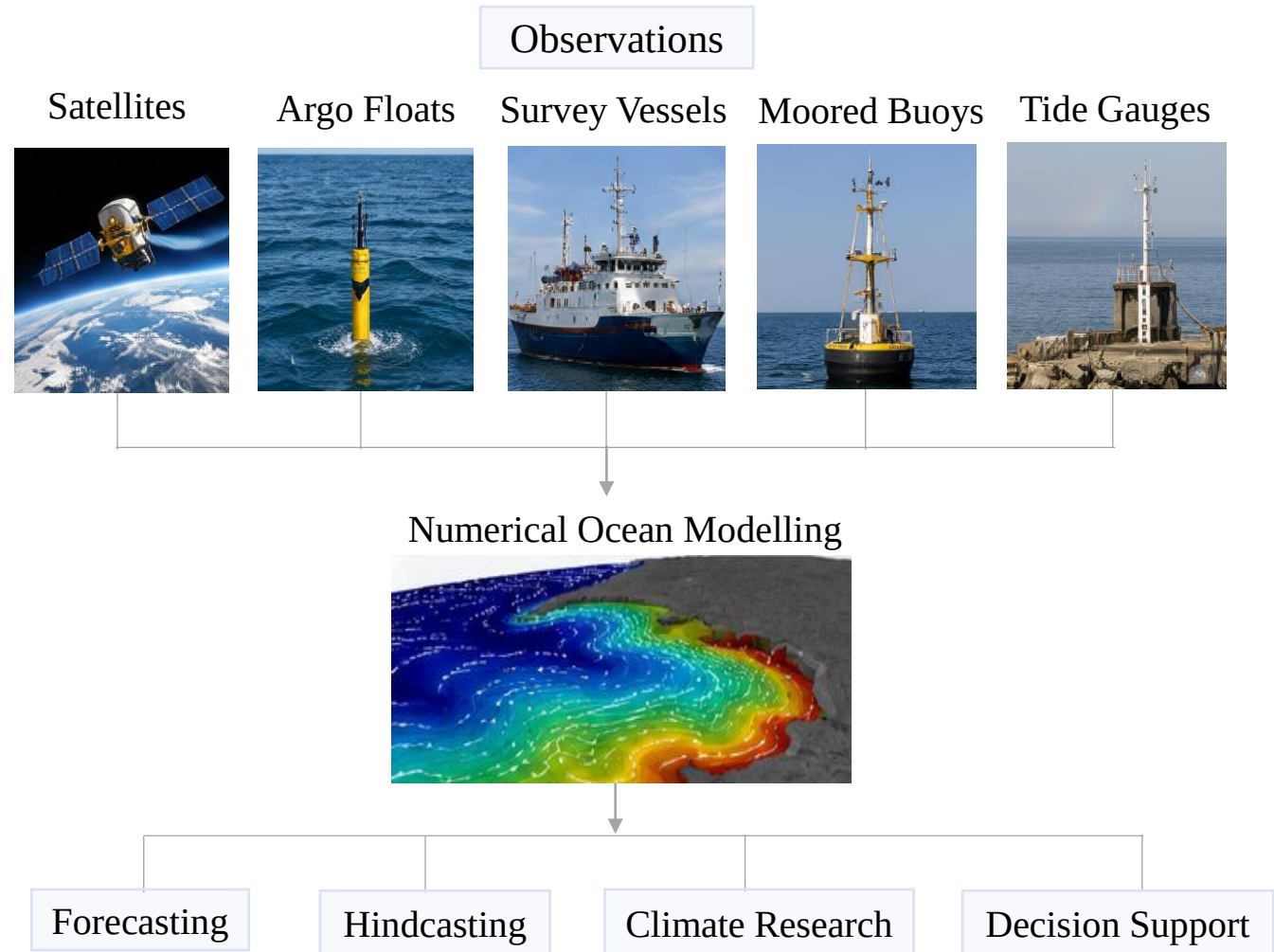
Motivations for Ocean Modeling

Motivation for Ocean Modeling

Understanding, reconstructing, and predicting the ocean through numerical modeling

Drivers for Ocean Modeling

- **Vast Ocean:**
Global-scale processes exceed observational coverage.
- **Highly Dynamic:**
Variability spans multiple spatial and temporal scales.
- **Incomplete Observations:**
Measurements are incomplete in space and time.
- **Complex Interactions:**
Coupled physical and biogeochemical interactions.
- **Predictive Capability:**
Reconstruct past and forecast future conditions.

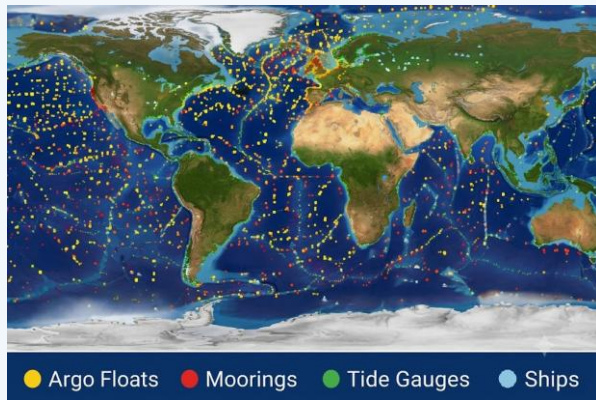


Limitations of Ocean Observations

Challenges in obtaining a complete and continuous representation of the ocean

Spatial Limitations

- Sparse sampling across the global ocean
- Limited observations in the deep ocean and polar regions
- Incomplete coverage of coastal and marginal seas



Temporal Limitations

- Intermittent observations in time
- Short observational records
- Irregular sampling frequencies



Intermittent
Sampling



Short
Campaign



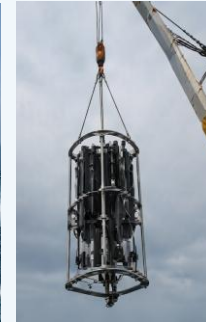
Discontinuous
Records

Operational Limitations

- High deployment and maintenance costs
- Harsh marine environments
- Instrument failures and data gaps



Harsh
Environment

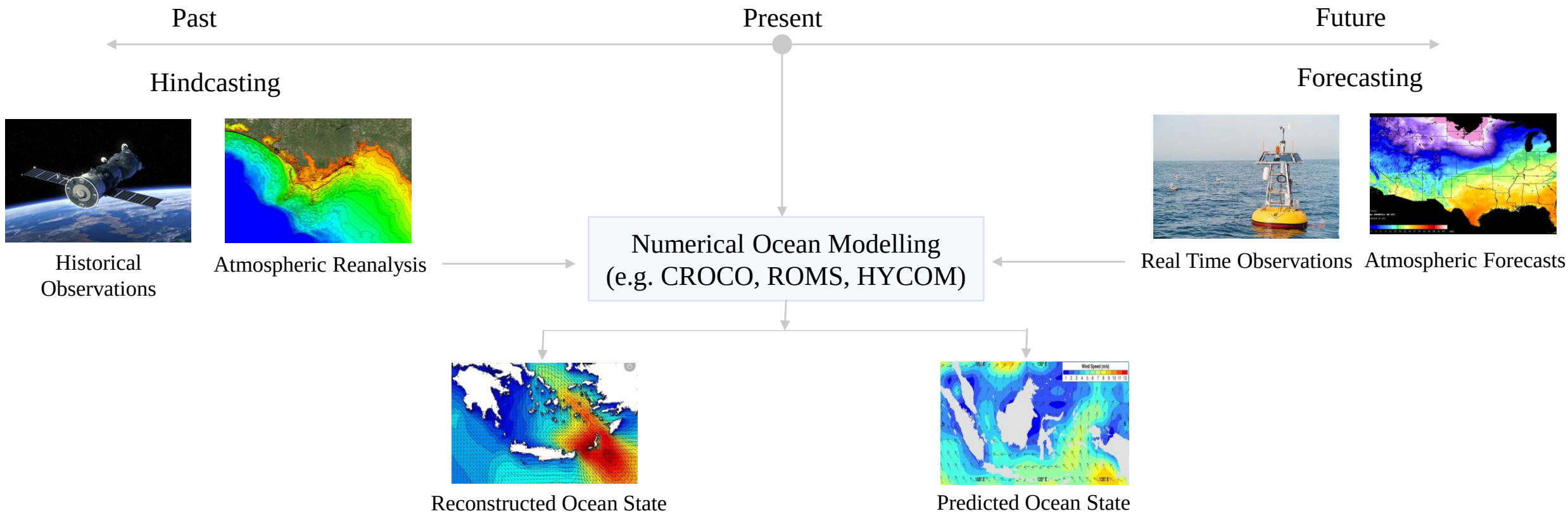


High Cost



Instrument
Failures

Hindcasting and Forecasting

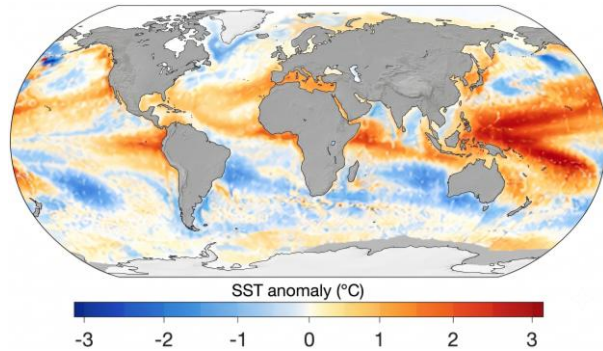


Aspect	Hindcasting	Forecasting
Objective	Reconstruct historical ocean conditions	Predict future ocean conditions
Primary Forcing	Atmospheric reanalysis	Forecast atmospheric forcing
Time Horizon	Years to decades	Hours to seasonal
Representative Applications	Climate variability, Extreme event analysis	Operational forecasting, Marine hazards

Research Applications of Ocean Models

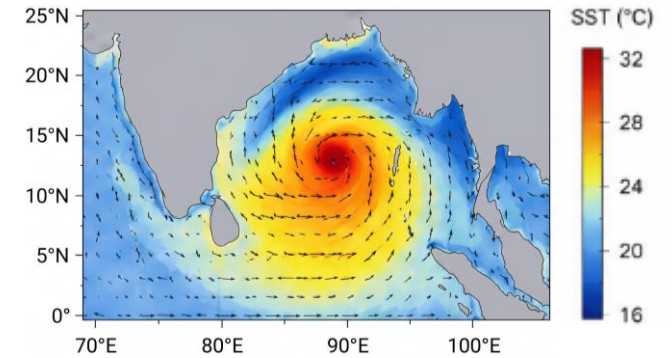
Climate Variability and Change

- Ocean warming and long-term trends
- Air–sea interaction
- Climate variability (e.g., ENSO, IOD)



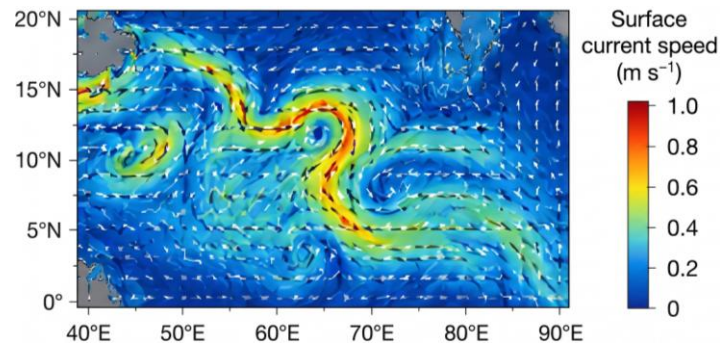
Extreme Ocean Events

- Marine heatwaves
- Storm surge dynamics
- Coastal flooding
- Tropical cyclone–ocean interactions



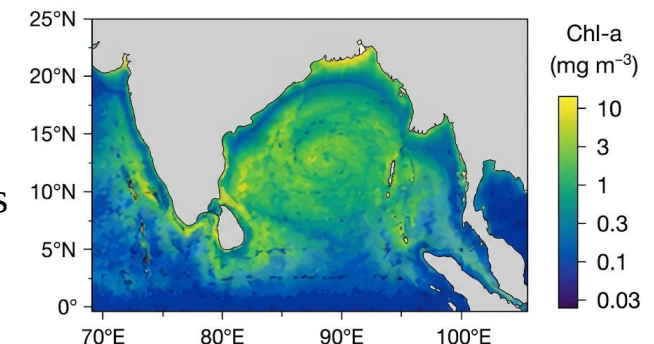
Physical Oceanography

- Ocean circulation
- Coastal upwelling
- Mesoscale and sub-mesoscale processes
- Water mass transport



Biogeochemical and Ecosystem Studies

- Nutrient transport
- Primary productivity
- Oxygen minimum zones
- Carbon cycling



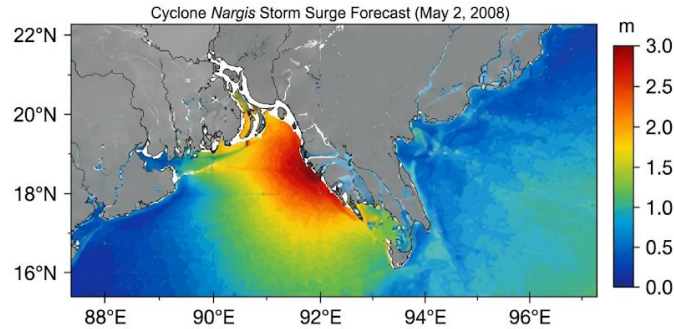
Representative Outputs

- Sea Surface Temperature
- Salinity
- Ocean Currents
- Sea Surface Height
- Mixed Layer Depth
- Biogeochemical Tracers

Operational Applications of Ocean Models

Coastal Hazard Forecasting

- Storm surge prediction
- Coastal inundation
- High-wave and coastal flooding forecasts



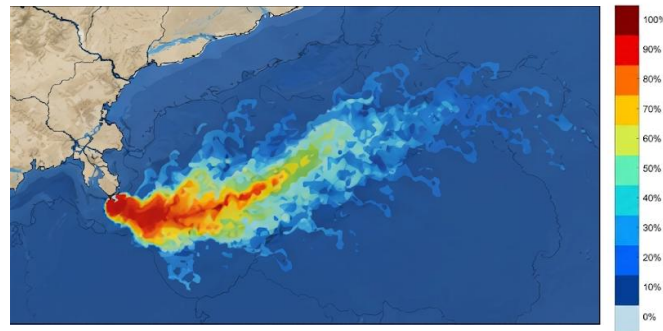
Marine Navigation and Safety

- Ship route optimization
- Search and rescue (SAR)
- Maritime situational awareness



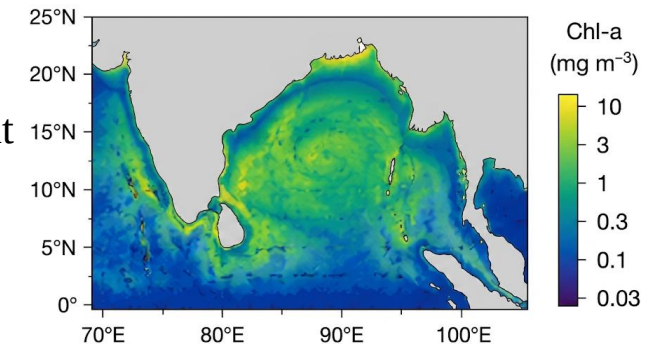
Environmental Emergency Response

- Oil spill trajectory forecasting
- Marine pollution dispersion
- Harmful algal bloom monitoring



Marine Resource Management

- Fisheries forecasting
- Aquaculture management
- Marine spatial planning



Representative Operational Outputs

- Sea Surface Temperature
- Ocean Currents
- Storm Surge Forecasts
- Wave Conditions
- Sea Surface Height
- Drift Trajectories

Scenario Experiments

Scenario Definition

Define an alternative forcing scenario.
(SSP, Wind, River discharge, SLR)



Model Configuration

Configure the numerical model with the modified forcing conditions.



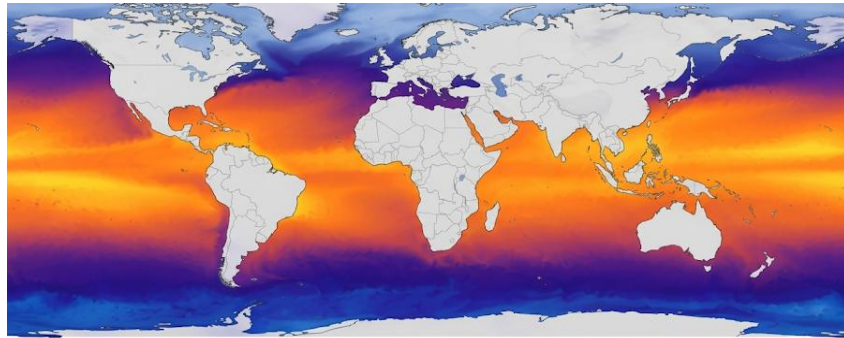
Numerical Simulation

Run the model to simulate the ocean response under the prescribed scenario.

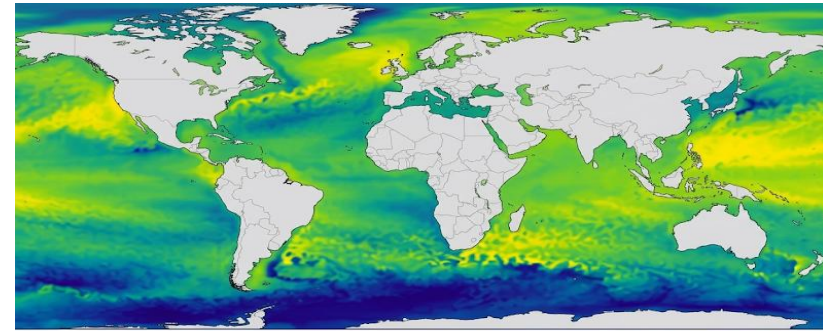


Impact Assessment

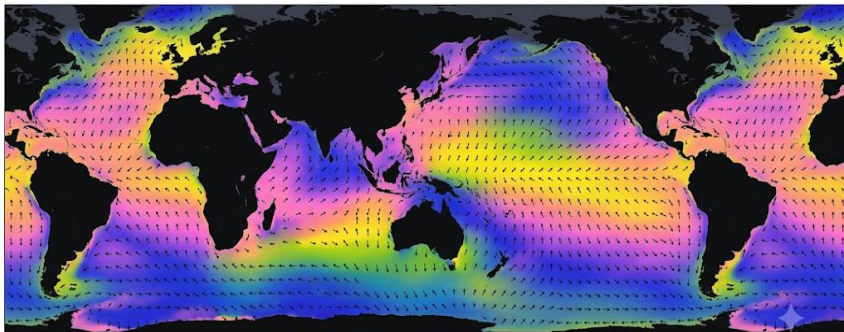
Analyze changes relative to a reference simulation.



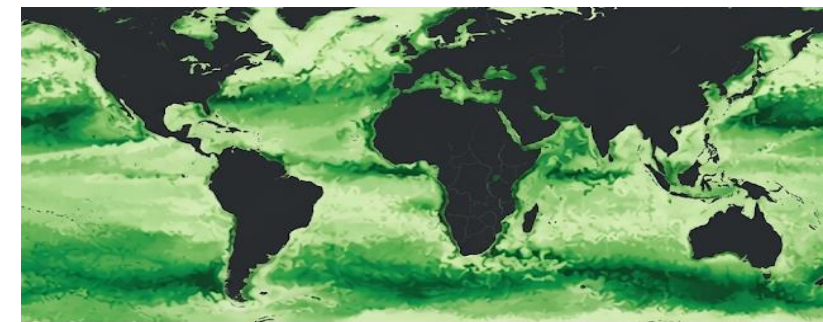
Sea Water Potential Temperature (°C) – CMEMS GLORYS12V1



Sea Surface Height Above Geoid (m) – CMEMS DUACS



Sea Surface Wave Direction (°) – CMEMS MFWAM



Mass Concentration of Chlorophyll-a in Sea Water (mg m^{-3}) – CMEMS GOBAF

Sensitivity Experiments

Reference Simulation

Run the model using the baseline configuration.

Modify A Parameter

Perturb a single parameter while keeping all others constant.

Numerical Simulation

Run the modified experiment.

Compare with Reference

Evaluate differences relative to the baseline simulation.

Quantify Model Response

Assess the sensitivity of model outputs to the applied perturbation.

Common Sensitivity Parameters

Forcing Parameters

- Wind stress
- Surface heat flux
- Freshwater flux
- River discharge

Physical Parameters

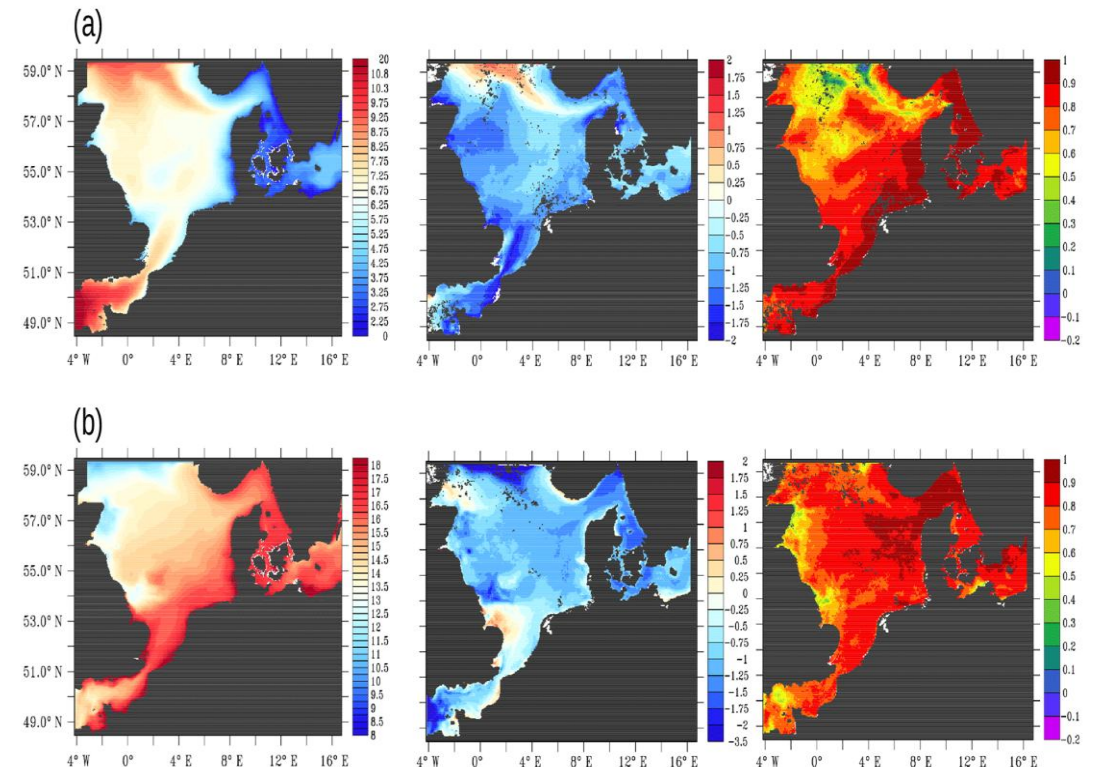
- Vertical mixing coefficients
- Bottom friction
- Surface roughness

Model Configuration

- Horizontal resolution
- Vertical levels
- Time step
- Numerical schemes

Boundary & Initial Conditions

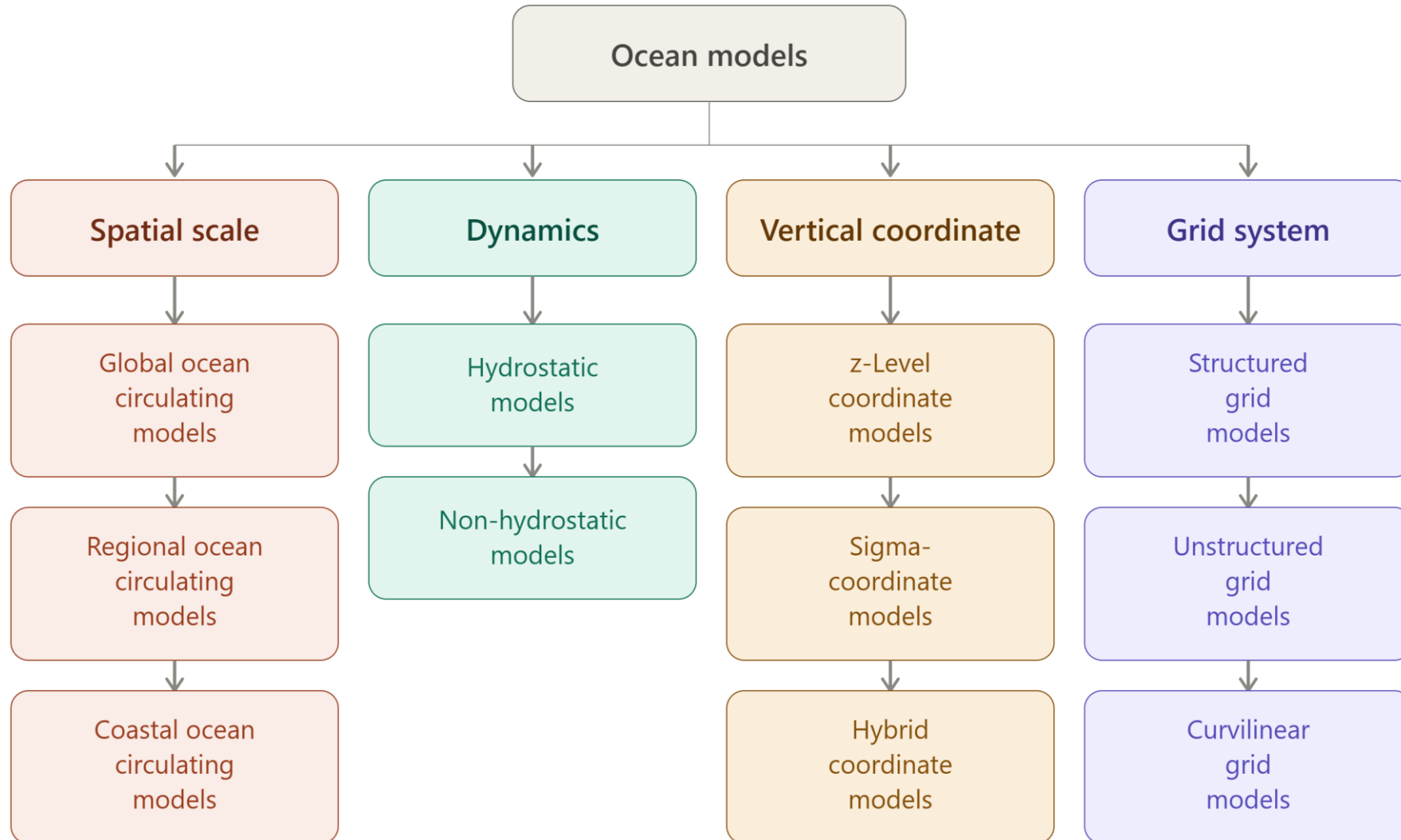
- Open boundary conditions
- Initial temperature and salinity
- Tidal forcing



Module 02

Types of Ocean Model

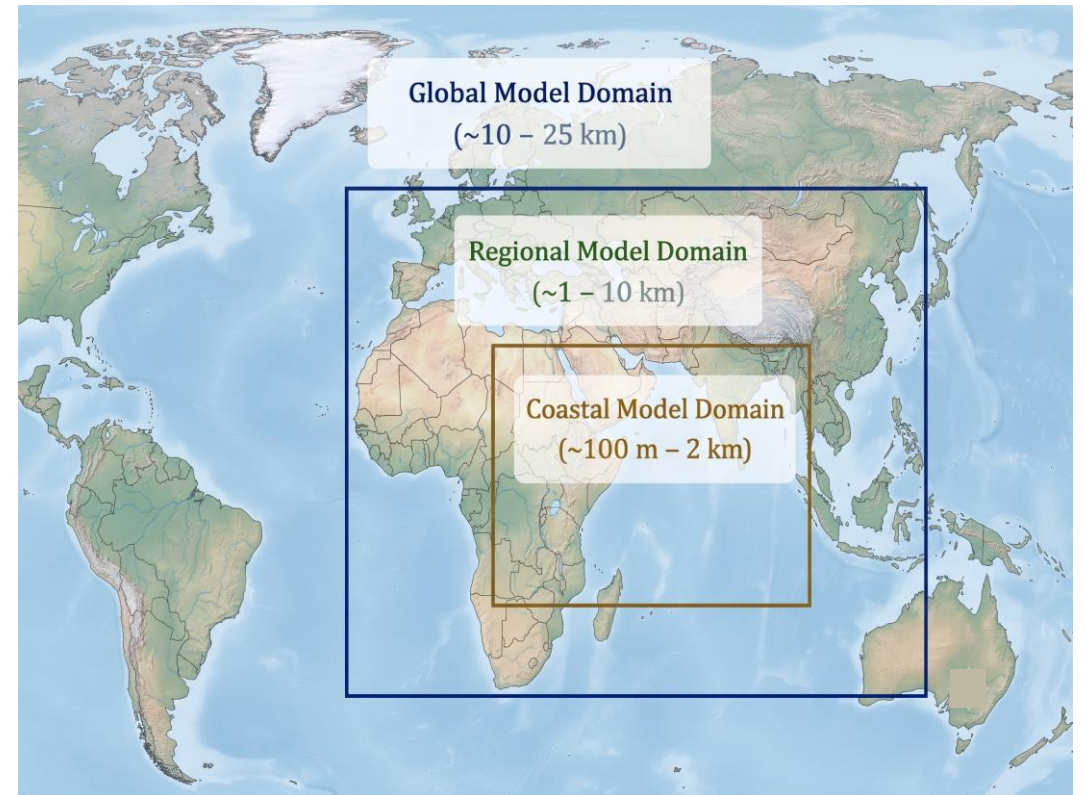
Classification of Ocean Models



Classification by Spatial Scale

Spatial scale defines the geographical extent of an ocean model, from global circulation to high-resolution coastal environments.

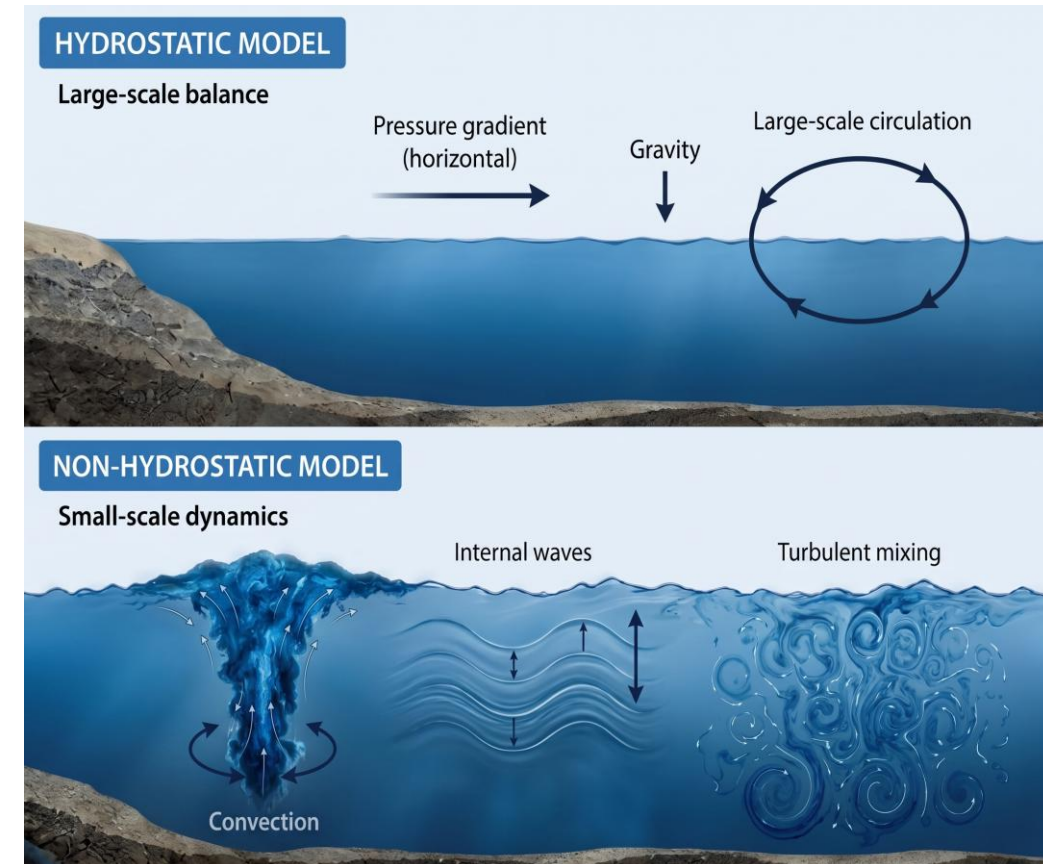
Category	Coverage	Typical Resolution	Representative Models
Global Ocean Circulating Models (OGCMs)	Entire globe	5–25 km	HYCOM, NEMO
Regional Ocean Circulating Models	Ocean basins or regional seas	1–10 km	ROMS, CROCO
Coastal Ocean Circulating Models	Estuaries, shelves, bays	100 m–2 km	CROCO, FVCOM, SCHISM



Classification by Governing Dynamics

Ocean models differ in the physical assumptions used to simulate ocean dynamics.

Hydrostatic Models	Non-Hydrostatic Models
Assumes vertical acceleration is negligible	Resolves vertical acceleration explicitly
Suitable for large-scale ocean circulation	Suitable for small-scale turbulent processes
Computationally efficient	Computationally expensive
Used for climate and operational forecasting	Used for convection, internal waves, and overflow studies
Examples: HYCOM, NEMO, ROMS, CROCO (default), FVCOM	Examples: CROCO (optional), MITgcm, Delft3D

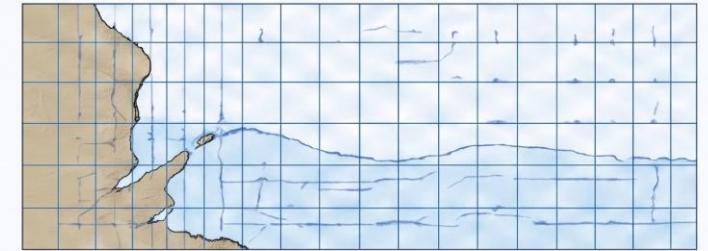


Classification by Computational Grid System

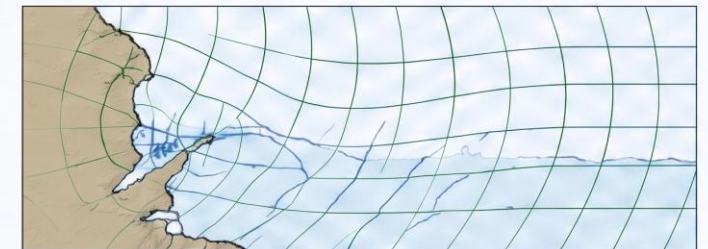
Ocean models are classified by the computational grid used to discretize the model domain

Grid Type	Characteristics	Advantages	Representative Models
Structured Grid Models	Regular rectangular grid	Simple, computationally efficient	HYCOM, NEMO
Curvilinear Grid Models	Grid follows coastlines while maintaining connectivity	Better representation of complex coastlines	ROMS, CROCO, POM
Unstructured Grid Models	Variable-size triangular or polygonal mesh	Flexible resolution for irregular coastlines and estuaries	FVCOM, SCHISM

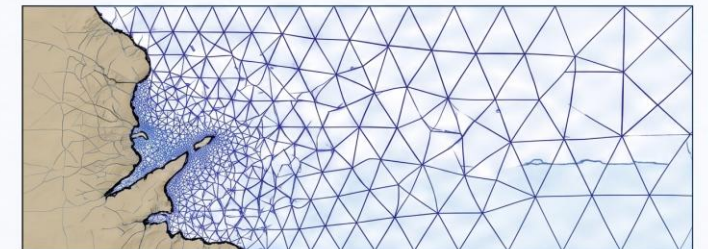
(a) Structured (Rectilinear) Grid



(b) Curvilinear Structured Grid



(c) Unstructured Grid



Numerical Ocean Modeling: Major Modeling Systems

HYCOM (Hybrid Coordinate Ocean Model)

Developer

- U.S. Naval Research Laboratory (NRL)

Grid

- Structured Arakawa C-grid

Vertical Coordinate

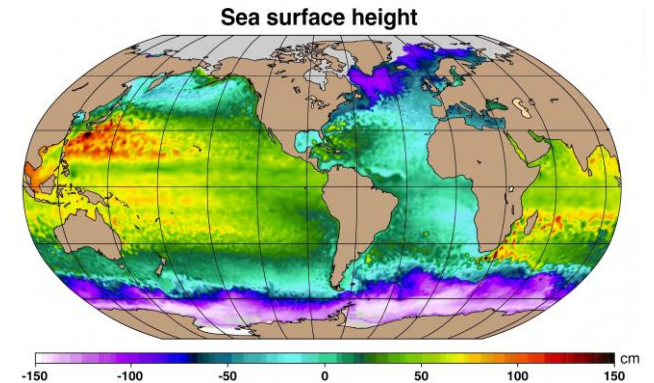
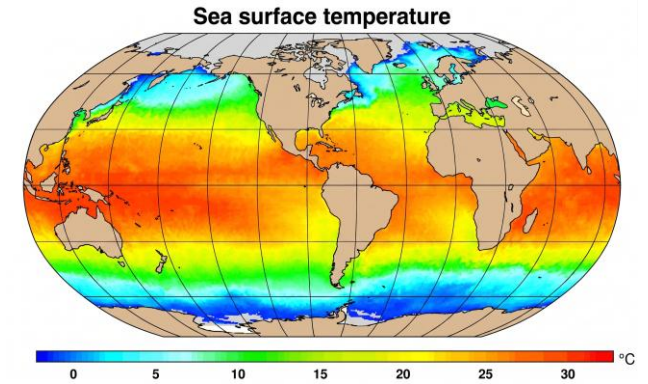
- Hybrid (Iso-pycnal + Sigma + z-level)

Applications

- Global ocean forecasting & climate research
- Boundary conditions for regional ocean models

Key Strengths

- Hybrid vertical coordinate system (deep ocean to coastal regions)
- High-resolution operational forecasting (captures major currents & mesoscale eddies)



1/12° (~7 km) global HYCOM forecast of SST and SSH on January 1, demonstrating high-resolution ocean prediction

NEMO (Nucleus for European Modelling of the Ocean)

Developer

- European NEMO Consortium

Grid

- Structured Arakawa C-grid

Vertical Coordinate

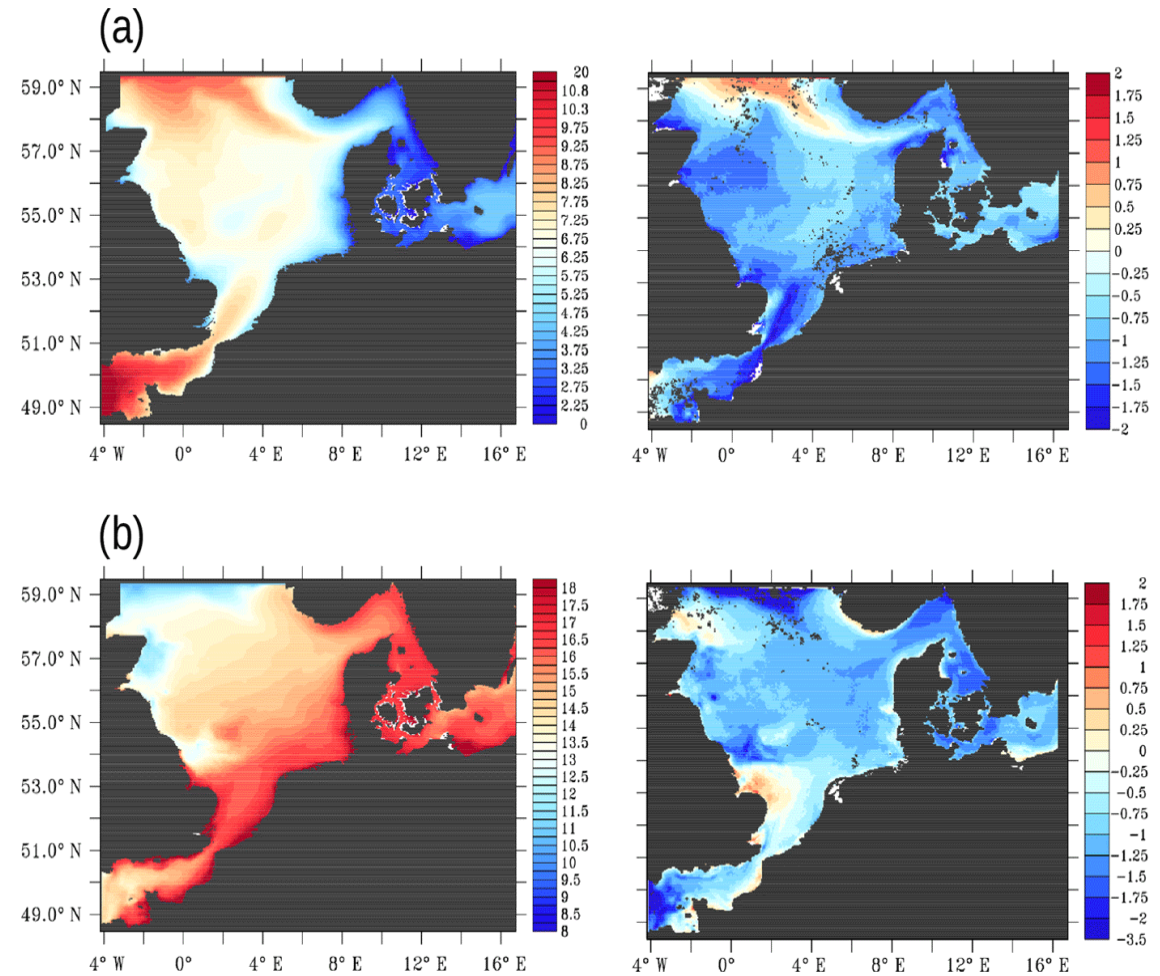
- z-level (with optional partial steps & adaptive coordinates)

Applications

- Climate modeling & Earth system simulations
- Operational ocean forecasting

Key Strengths

- Excellent for long-term climate simulations
- Strong coupling with atmosphere, sea ice & biogeochemical models

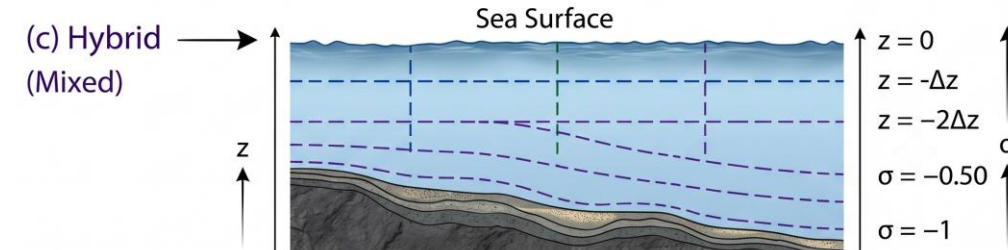
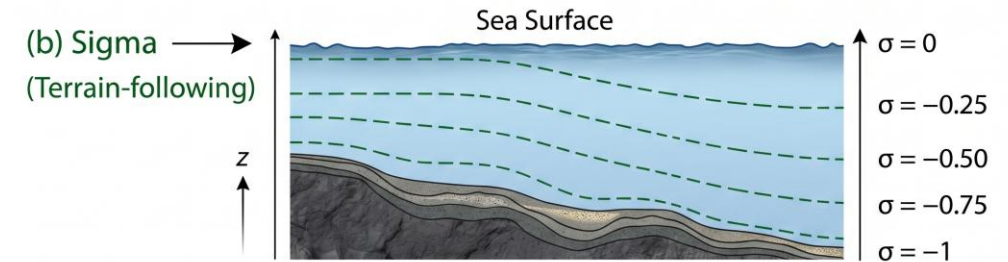
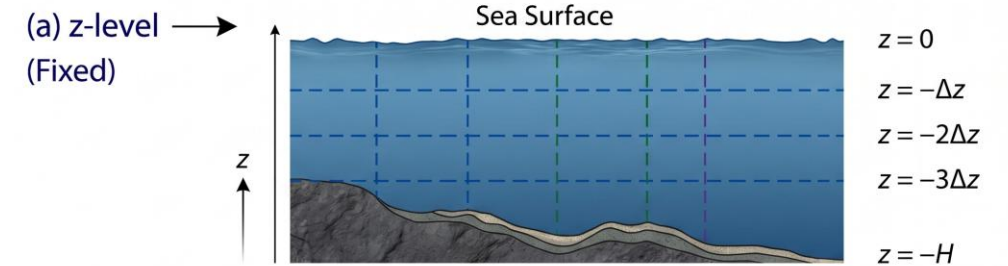


NEMO-Nordic simulated North Sea SST for winter (a) and summer (b) means, 1990–2005, with right-hand panels showing bias against observations.

Classification by Vertical Coordinate System

Ocean models are classified by the vertical coordinate system used to represent the water column and ocean bathymetry.

Coordinate System	Description	Advantages	Representative Models
z-level Coordinate Models	Fixed horizontal layers	Simple implementation; suitable for deep-ocean circulation	NEMO
Terrain Following (Sigma-coordinate) Models	Layers follow the seabed	Better representation of coastal and shelf processes	POM, ROMS, CROCO, FVCOM
Hybrid Coordinate Models	Combination of z-level, sigma, and iso-pycnal coordinates	High accuracy across both deep and shallow waters	HYCOM



POM (Princeton Ocean Model)

Developer

- Princeton University

Grid

- Structured Curvilinear C-grid

Vertical Coordinate

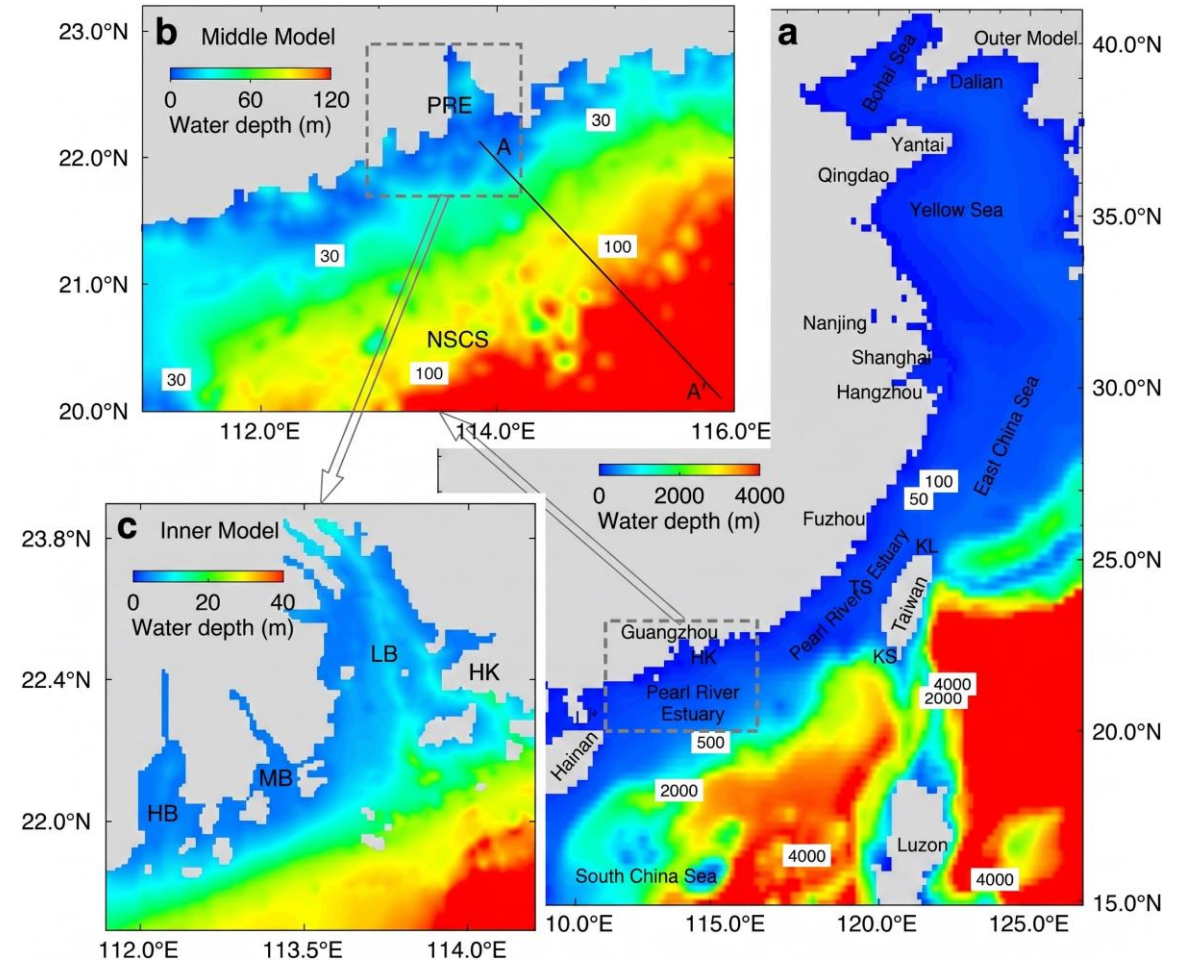
- Sigma (Terrain-following)

Applications

- Coastal & shelf sea circulation
- Estuarine and regional ocean modeling

Key Strengths

- Efficient terrain-following sigma coordinate
- Foundation for many modern coastal ocean models



Triply-nested POM domains from the South China Sea to the Pearl River Estuary, showing multi-scale coastal modeling from open ocean to estuarine scales.

ROMS (Regional Ocean Modeling System)

Developer

- Rutgers University (UCLA)

Grid

- Structured Curvilinear Arakawa C-grid

Vertical Coordinate

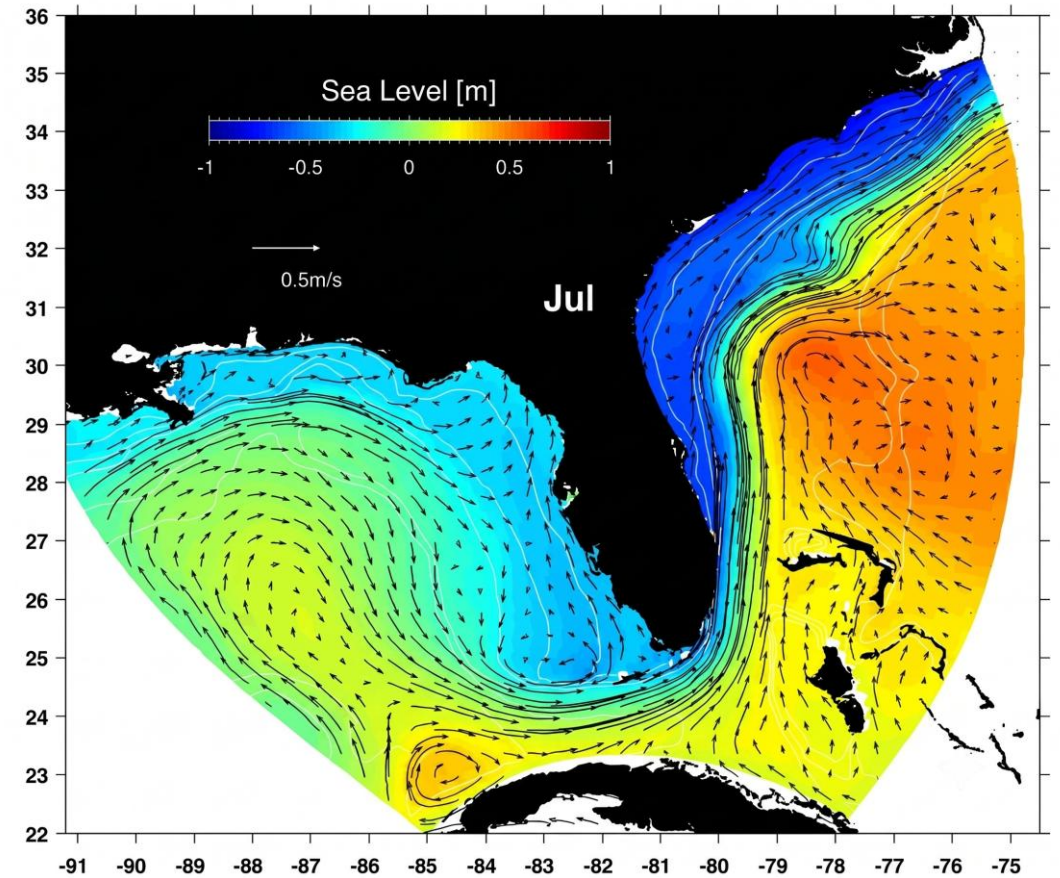
- Sigma (Terrain-following)

Applications

- Coastal circulation & upwelling systems
- Regional ocean forecasting

Key Strengths

- High-resolution coastal simulations
- Foundation of the CROCO model



ROMS-simulated July sea level (colors) and surface currents (vectors) over the West Florida Shelf (SEACOOS domain), showing high-resolution coastal circulation.

CROCO (Coastal and Regional Ocean Community Model)

Developer

- CROCO Consortium
- IRD, INRIA, CNRS, IFREMER & partner institutions

Grid

- Structured Curvilinear Arakawa C-grid

Vertical Coordinate

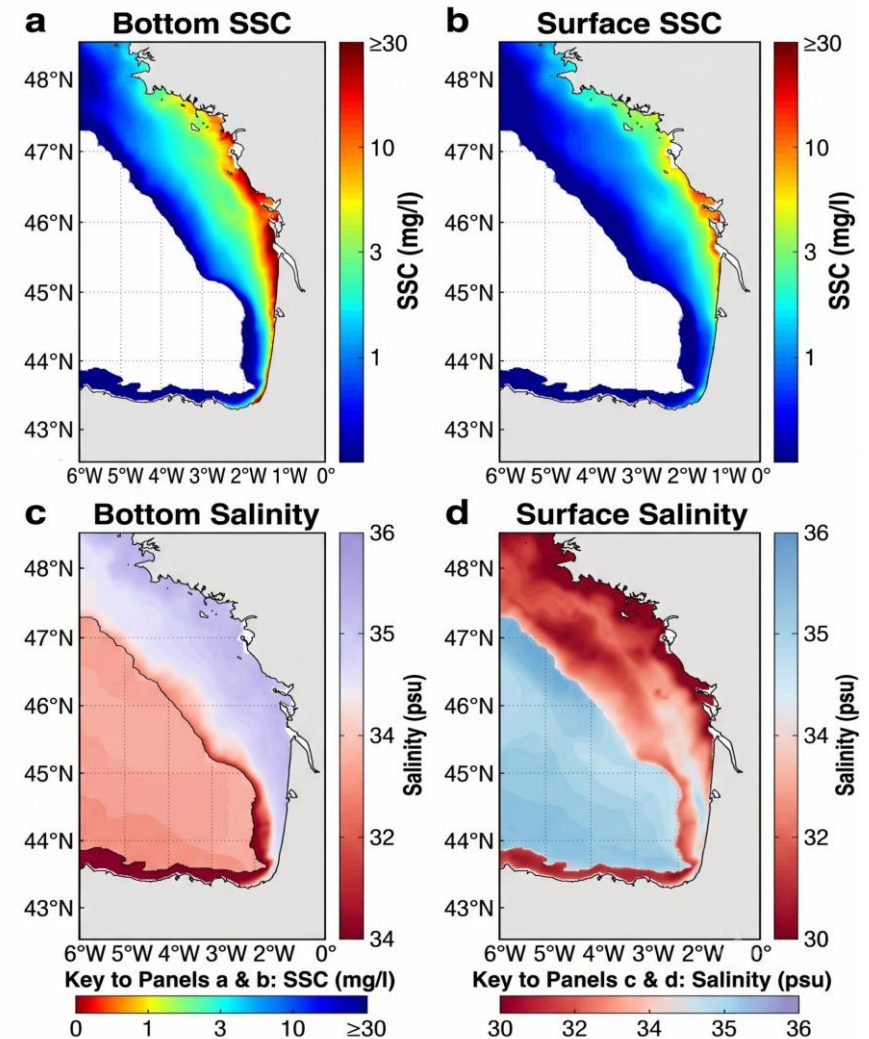
- Terrain-following Sigma-coordinate

Applications

- Coastal & regional ocean modeling
- Coupled ocean–wave–sediment–ecosystem simulations

Key Strengths

- AGRIF nesting & parallel computing
- Flexible coupling (WRF, WW3, PISCES, sediment)



CROCO-DEAL-S simulated bottom and surface suspended sediment concentration (a, b) and salinity (c, d) for the Bay of Biscay, 2000–2019 hindcast, showing peak sediment near the coastline.

FVCOM (Finite Volume Community Ocean Model)

Developer

- University of Massachusetts Dartmouth (SMAST)

Grid

- Unstructured Triangular Grid

Vertical Coordinate

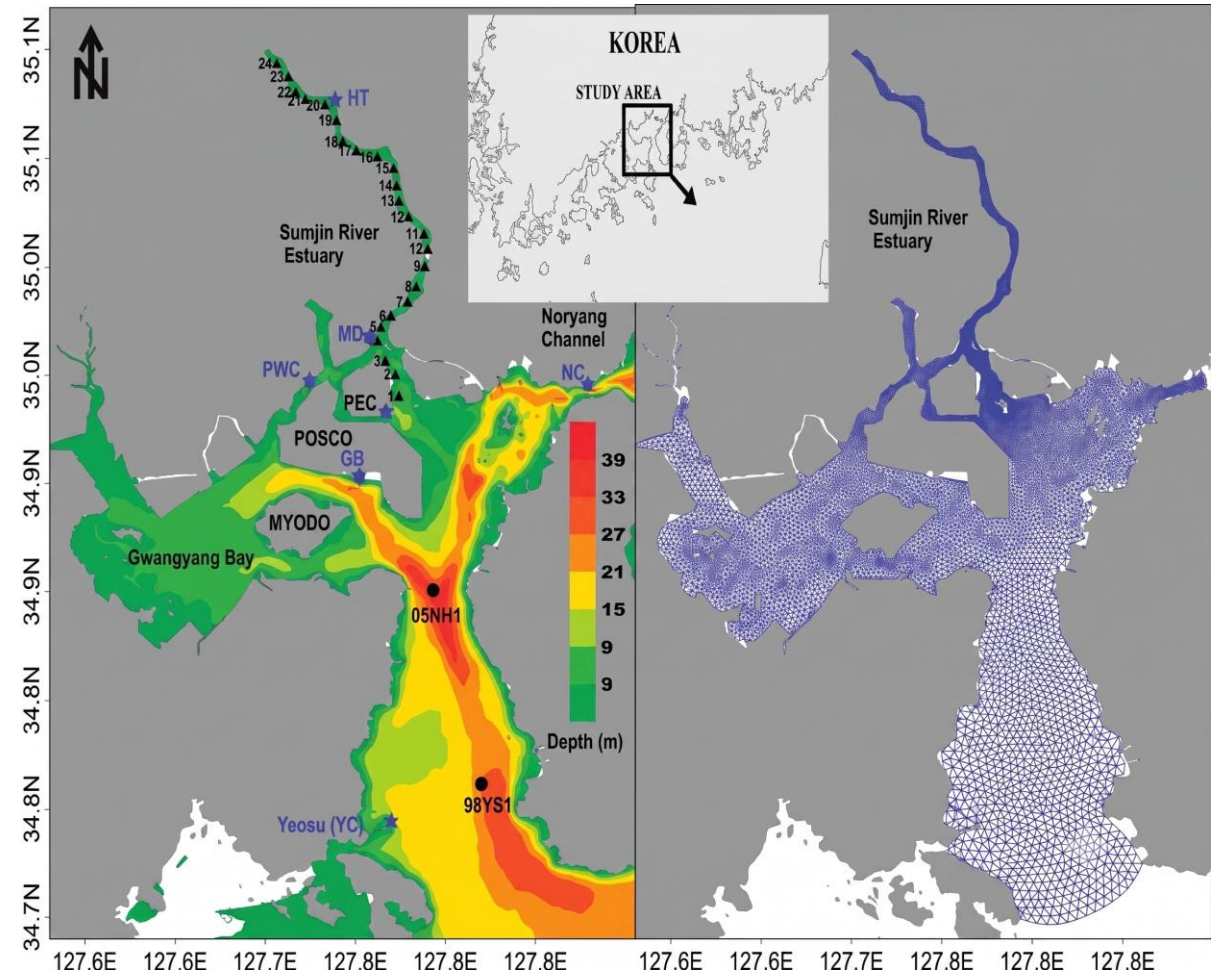
- Terrain-following Sigma-coordinate

Applications

- Estuarine & coastal circulation
- Tides, storm surge & flooding

Key Strengths

- Flexible mesh for complex coastlines
- High accuracy in shallow coastal waters



FVCOM bathymetry (left) and unstructured triangular mesh (right) for the Sumjin River Estuary and Gwangyang Bay, South Korea, showing mesh refinement in narrow channels and near the coastline.

SCHISM (Semi-implicit Cross-scale Hydro-science Integrated System Model)

Developer

- Virginia Institute of Marine Science (VIMS), William & Mary

Grid

- Unstructured Triangular Grid

Vertical Coordinate

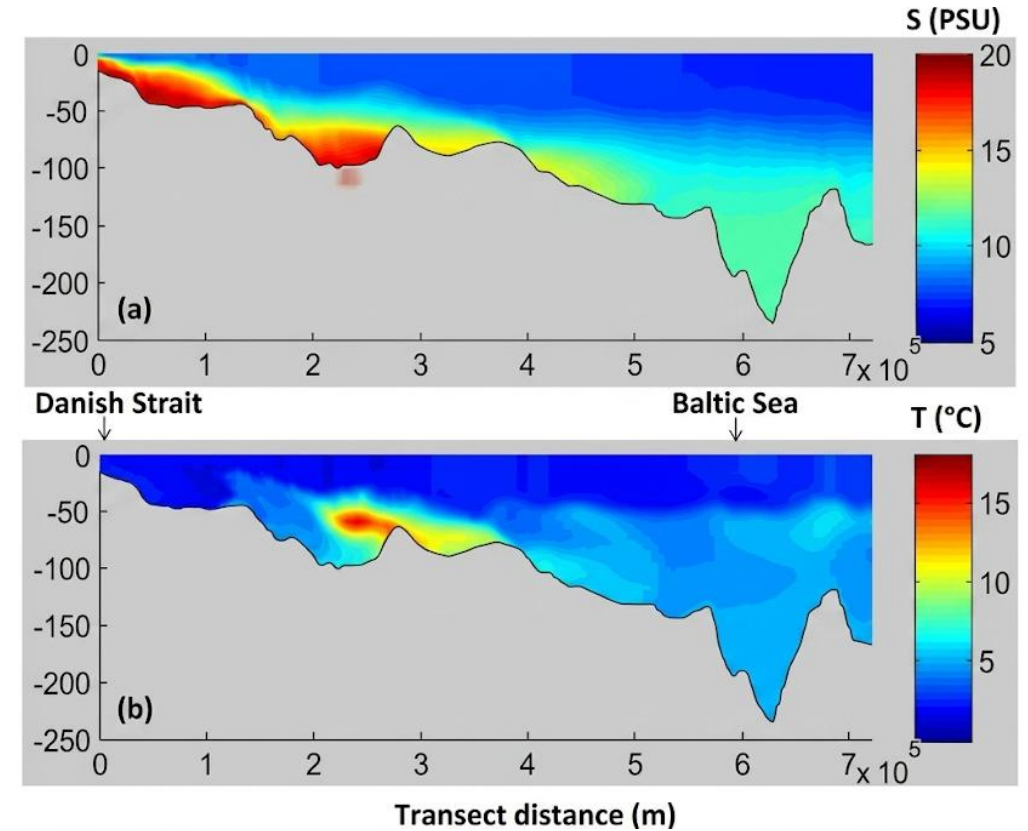
- Hybrid SZ Coordinate (Sigma + Z-level)

Applications

- Estuarine & coastal circulation
- Storm surge, flooding & tsunami simulation

Key Strengths

- Flexible unstructured mesh for complex coastlines
- Excellent wetting–drying capability and cross-scale modeling



SCHISM-simulated salinity (a) and temperature (b) along a transect from the Danish Strait to the Baltic Sea, capturing vertical stratification and dense-water overflow over complex bathymetry.

Comparison of Major Ocean Models						
Model	Typical Domain	Vertical Coordinate	Grid System	Dynamics	Nesting	Primary Strength
HYCOM	Global	Hybrid	Structured Arakawa C-grid	Hydrostatic	One-way	Multi-scale global forecasting
NEMO	Global	z-level	Structured Arakawa C-grid	Hydrostatic	AGRIF	Climate and Earth system modeling
POM	Regional / Coastal	Sigma	Curvilinear Arakawa C-grid	Hydrostatic	One-way	Shelf and estuarine circulation
ROMS	Regional / Coastal	Sigma	Curvilinear Arakawa C-grid	Hydrostatic	Two-way	High-resolution coastal dynamics
CROCO	Regional / Coastal	Sigma	Curvilinear Arakawa C-grid	Hydrostatic	AGRIF (Two-way)	Coupled coastal process modeling
FVCOM	Coastal	Sigma	Unstructured Triangular	Hydrostatic	Flexible	Complex coastlines and estuaries
SCHISM	Coastal / Estuarine	Hybrid Sigma-z	Unstructured Hybrid Mesh	Hydrostatic	Flexible	Flooding, inundation, wetlands

Module 03

Fundamentals of Ocean Model

The Primitive Equations: Foundations of Ocean Models

A simplified set of equations derived from Navier–Stokes using the **Boussinesq** and **hydrostatic approximations** that govern ocean motion, mass conservation, and thermodynamic state.

- ◇ **Boussinesq Approximation:** Density variation ignored except in buoyancy term
- ◇ **Hydrostatic Approximation:** Vertical acceleration negligible

1. Momentum Equations (Conservation of motion)
2. Continuity Equation (Conservation of mass)
3. Hydrostatic Equation (Vertical Pressure balance)
4. Equation of State (links density to T, S, P)
5. Heat & Salt (Tracer) Equations (Conservation of energy/salinity)

Momentum equations

$$\frac{\partial u}{\partial t} + u \frac{\partial u}{\partial x} + v \frac{\partial u}{\partial y} + w \frac{\partial u}{\partial z} = -\frac{1}{\rho} \frac{\partial p}{\partial x} + f_v + \frac{\partial}{\partial x} \left[A_x \frac{\partial u}{\partial x} \right] + \frac{\partial}{\partial y} \left[A_y \frac{\partial u}{\partial y} \right] + \frac{\partial}{\partial z} \left[A_z \frac{\partial u}{\partial z} \right]$$

$$\frac{\partial v}{\partial t} + u \frac{\partial v}{\partial x} + v \frac{\partial v}{\partial y} + w \frac{\partial v}{\partial z} = -\frac{1}{\rho} \frac{\partial p}{\partial y} - f_u + \frac{\partial}{\partial x} \left[A_x \frac{\partial v}{\partial x} \right] + \frac{\partial}{\partial y} \left[A_y \frac{\partial v}{\partial y} \right] + \frac{\partial}{\partial z} \left[A_z \frac{\partial v}{\partial z} \right]$$

Hydrostatic equation

$$\frac{\partial p}{\partial z} = -\rho g$$

Equation of State

$$\rho = \rho(T, S, P)$$

Continuity equation

$$\frac{\partial u}{\partial x} + \frac{\partial v}{\partial y} + \frac{\partial w}{\partial z} = 0$$

Heat and salt equations

$$\frac{\partial T}{\partial t} + u \frac{\partial T}{\partial x} + v \frac{\partial T}{\partial y} + w \frac{\partial T}{\partial z} = \frac{\partial}{\partial x} \left[K_x \frac{\partial T}{\partial x} \right] + \frac{\partial}{\partial y} \left[K_y \frac{\partial T}{\partial y} \right] + \frac{\partial}{\partial z} \left[K_z \frac{\partial T}{\partial z} \right] + Q_T$$

$$\frac{\partial S}{\partial t} + u \frac{\partial S}{\partial x} + v \frac{\partial S}{\partial y} + w \frac{\partial S}{\partial z} = \frac{\partial}{\partial x} \left[K_x \frac{\partial S}{\partial x} \right] + \frac{\partial}{\partial y} \left[K_y \frac{\partial S}{\partial y} \right] + \frac{\partial}{\partial z} \left[K_z \frac{\partial S}{\partial z} \right] + Q_S$$

Momentum Equations: Conservation of Motion

$$\begin{array}{c}
 \text{Local} \\ \text{Acceleration} \\ \underbrace{\frac{\partial u}{\partial t}} + \underbrace{u \frac{\partial u}{\partial x} + v \frac{\partial u}{\partial y} + w \frac{\partial u}{\partial z}}_{\text{Advection}} = \underbrace{-\frac{1}{\rho} \frac{\partial p}{\partial x}}_{\text{Pressure Gradient Force}} + \underbrace{fv}_{\text{Coriolis Force}} + \underbrace{\frac{\partial}{\partial x} \left[A_x \frac{\partial u}{\partial x} \right] + \frac{\partial}{\partial y} \left[A_y \frac{\partial u}{\partial y} \right]}_{\text{Horizontal Turbulent Mixing (Friction)}} + \underbrace{\frac{\partial}{\partial z} \left[A_z \frac{\partial u}{\partial z} \right]}_{\text{Vertical Turbulent Mixing (Friction)}}
 \end{array}$$

$$\frac{\partial v}{\partial t} + u \frac{\partial v}{\partial x} + v \frac{\partial v}{\partial y} + w \frac{\partial v}{\partial z} = -\frac{1}{\rho} \frac{\partial p}{\partial y} - fu + \frac{\partial}{\partial x} \left[A_x \frac{\partial u}{\partial x} \right] + \frac{\partial}{\partial y} \left[A_y \frac{\partial u}{\partial y} \right] + \frac{\partial}{\partial z} \left[A_z \frac{\partial u}{\partial z} \right]$$

u, v = horizontal velocity; p = pressure; ρ = density; f = Coriolis parameter; A = eddy viscosity/diffusivity coefficients

- Governs the evolution of horizontal velocity components (u, v) under pressure, rotation, and mixing.
- Vertical velocity (w) has no independent momentum equation. It is diagnosed from the continuity equation.

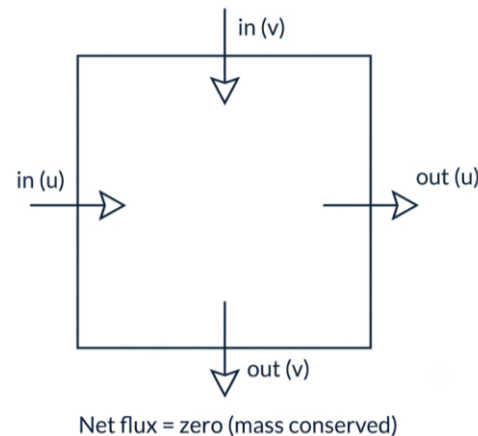
Continuity Equation: Conservation of Mass

Ensures mass is conserved by requiring the flow to be non-divergent at every point.

$$\frac{\partial u}{\partial x} + \frac{\partial v}{\partial y} + \frac{\partial w}{\partial z} = 0$$

u, v = horizontal velocity; w = vertical velocity; x, y = horizontal coordinates; z = vertical coordinate

- Under the Boussinesq approximation, seawater is treated as incompressible (Density changes are ignored except in the buoyancy term).
- This equation states that inflow must equal outflow for any fluid parcel. Water cannot accumulate or vanish at a point.
- It is not a prognostic equation; it does not evolve in time. Instead, it is used to diagnose vertical velocity (w) once u and v are known.



Hydrostatic Balance: Vertical Pressure Equilibrium

Simplifies the vertical momentum equation by balancing pressure against gravity alone.

$$\frac{\partial p}{\partial z} = -\rho g$$

p = pressure; z = depth (vertical coordinate); ρ = density; g = gravitational acceleration

- Vertical acceleration is negligible because ocean flows are far wider than they are deep (horizontal scale $\gg$ vertical scale).
- Pressure at any depth simply equals the weight of the water column above it.

Equation of State: Linking Density to Temperature and Salinity

Defines how seawater density depends on temperature, salinity, and pressure — the link that drives ocean circulation.

$$\rho = \rho(T, S, P)$$

- Density is a non-linear function of T, S, and P, computed using standardized empirical relationships rather than a simple formula.
- Density differences drive ocean circulation; denser water sinks, lighter water rises, generating buoyancy-driven currents.

Heat and Salt Equations: Transport of Temperature and Salinity

Describes how temperature and salinity are carried and mixed by ocean currents

$$\frac{\partial T}{\partial t} + u \frac{\partial T}{\partial x} + v \frac{\partial T}{\partial y} + w \frac{\partial T}{\partial z} = \frac{\partial}{\partial x} \left[K_x \frac{\partial T}{\partial x} \right] + \frac{\partial}{\partial y} \left[K_y \frac{\partial T}{\partial y} \right] + \frac{\partial}{\partial z} \left[K_z \frac{\partial T}{\partial z} \right] + Q_T$$

$$\frac{\partial S}{\partial t} + u \frac{\partial S}{\partial x} + v \frac{\partial S}{\partial y} + w \frac{\partial S}{\partial z} = \frac{\partial}{\partial x} \left[K_x \frac{\partial S}{\partial x} \right] + \frac{\partial}{\partial y} \left[K_y \frac{\partial S}{\partial y} \right] + \frac{\partial}{\partial z} \left[K_z \frac{\partial S}{\partial z} \right] + Q_s$$

T = temperature; S = salinity; K = mixing/diffusivity coefficient; Q = source/sink term (heat or freshwater flux)

- **Advection** (left side, after $\partial T / \partial t$): temperature is carried along with the current; the same u, v, w velocity fields from the momentum equations.
- **Diffusion and sources** (right side): mixing smooths out sharp gradients via coefficients K_x , K_y , K_z , while the source term Q_T represents external heat input such as solar radiation or surface heat flux.

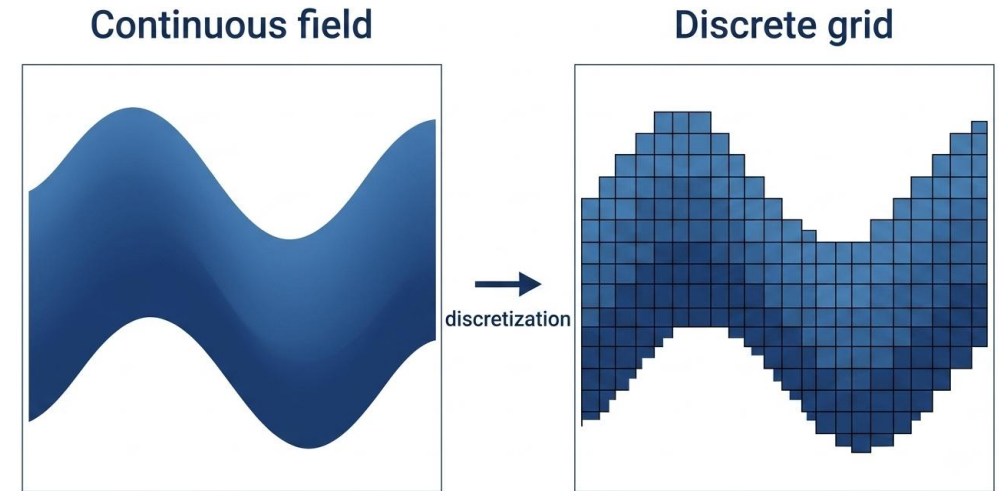
Numerical Discretization: From Continuous to Discrete

Why Ocean Models Need a Grid?

Bridging the physics equation to something a computer can actually solve

- ▶ The primitive equations describe the ocean as continuous in space and time → infinite points, infinite precision.
- ▶ Computers can only store and compute a finite number of values. The ocean must be broken into a grid.

- **Spatial discretization:** the ocean domain is divided into a finite grid of cells; each cell holds one value of velocity, temperature, or salinity, instead of a smooth continuous field.
- **Temporal discretization:** time is advanced in small discrete steps (Δt), not continuously; the model computes the ocean state one time step at a time.



This process converting continuous equations into a form solvable at discrete points is called **numerical discretization**

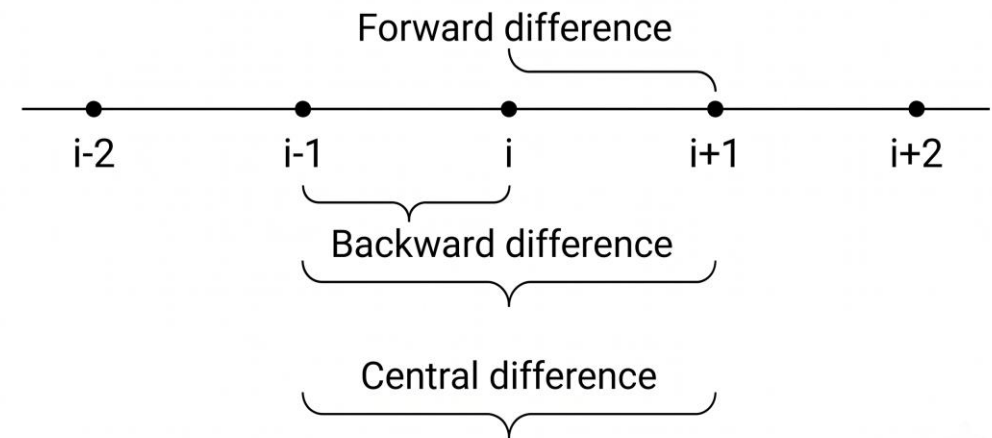
Finite Difference Approximation

- Replacing exact calculus with grid-point arithmetic a computer can perform.
- The true derivative is defined as a limit ($\Delta x \rightarrow 0$), **but a computer can never reach zero**. So it uses a small, finite Δx instead.

$$\text{Forward: } \frac{\partial u}{\partial x} \approx \frac{u_{i+1} - u_i}{\Delta x}$$

$$\text{Backward: } \frac{\partial u}{\partial x} \approx \frac{u_i - u_{i-1}}{\Delta x}$$

$$\text{Central: } \frac{\partial u}{\partial x} \approx \frac{u_{i+1} - u_{i-1}}{2\Delta x}$$



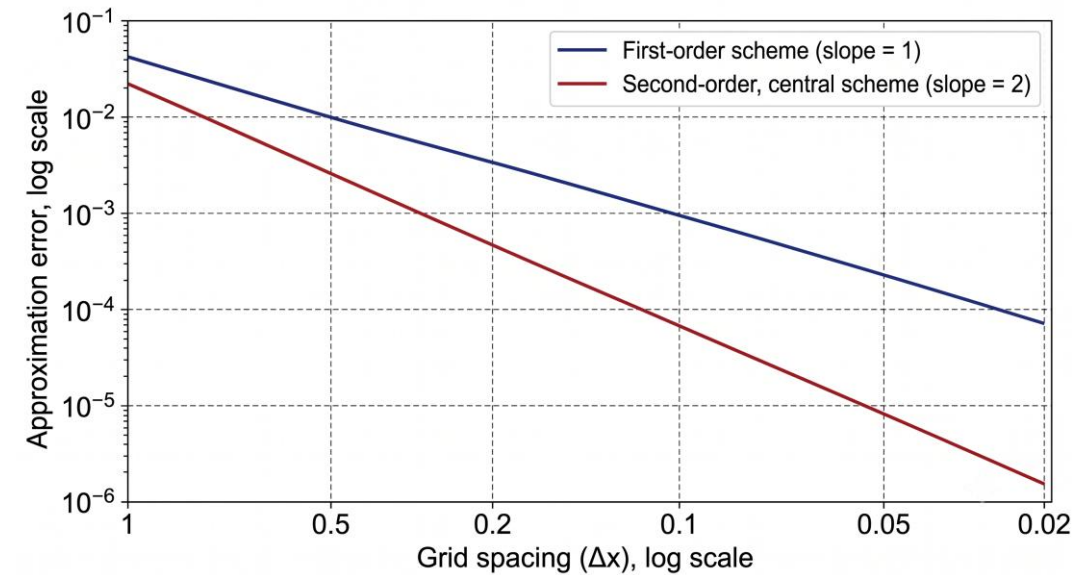
- Each grid point stores one value (u_i), spaced Δx apart. The derivative is approximated using the difference between neighboring points.
- **Forward** looks ahead (i, i+1); **Backward** looks behind (i-1, i); **Central** uses both neighbors symmetrically (i-1, i+1).
- These schemes turn a continuous calculus operation into simple arithmetic subtraction and division that a computer can execute at every grid point.

Accuracy of Finite Difference Schemes

Not all approximations are equally accurate.

The choice of scheme affects how close the model comes to the true solution.

- **Forward and backward** difference schemes are **first-order accurate**, meaning their error decreases in direct proportion to the grid spacing (Δx).
- **Central difference** is **second-order accurate**, meaning its error decreases with the square of the grid spacing (Δx^2), making it far more precise for the same grid resolution.
- **Because of this advantage, most ocean models rely primarily on central difference schemes for spatial derivatives.**



A finer grid reduces error but increases computational cost — a trade-off that reappears throughout model design, including the CFL condition and time stepping.

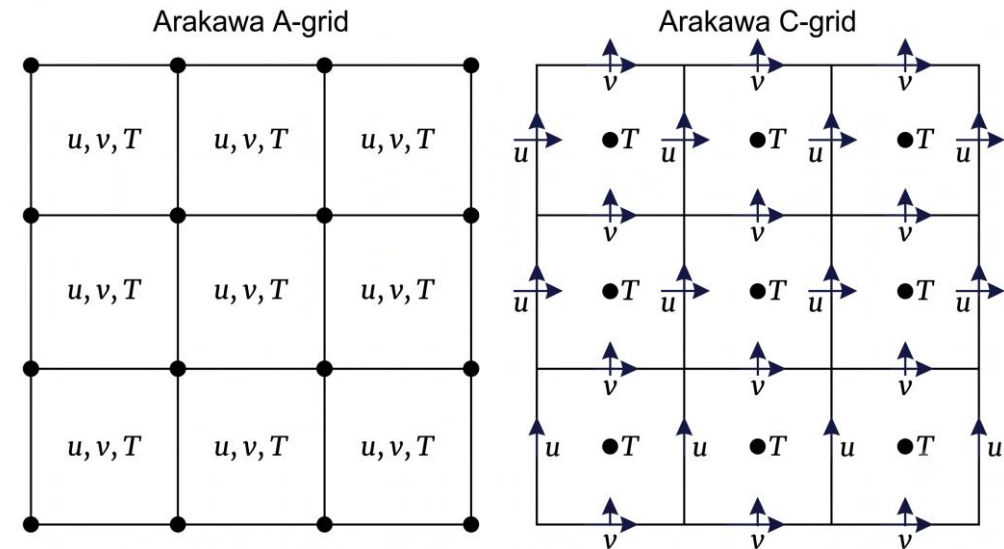
Grid Staggering

Why Variable Placement Matters?

The location where each variable is stored on the grid directly affects the accuracy and stability of the model

The simplest design stores every variable, velocity, temperature, and pressure, at the same grid point. This is called **an Arakawa A-grid**.

- On an A-grid, pressure gradients and velocity divergence require averaging neighboring points, since every variable shares the same location.
- This averaging can introduce spurious **checkerboard noise** that degrades solution accuracy.
- Staggered grids avoid this by placing different variables at different positions within each cell, allowing gradients to be computed directly between adjacent points.

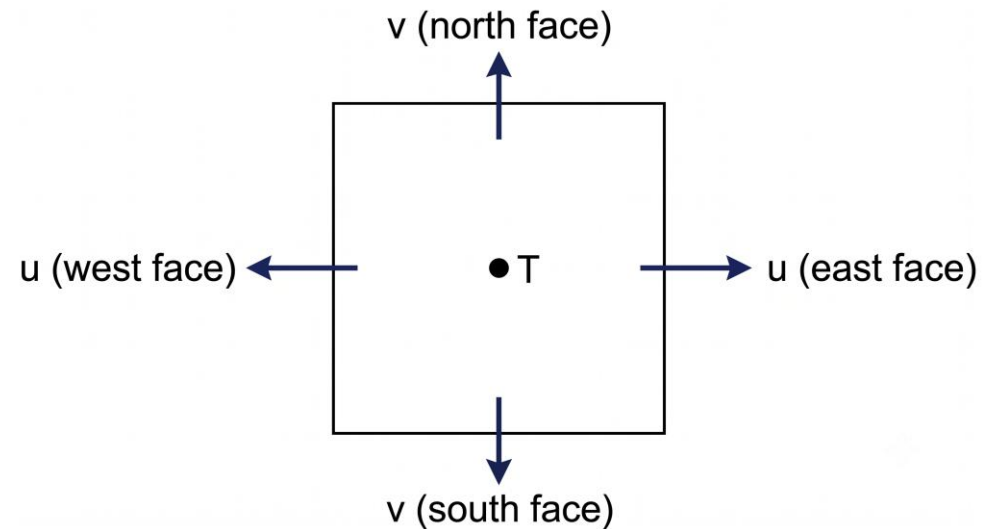


The most widely used staggered layout in ocean modeling is the Arakawa C-grid, used by CROCO, ROMS, and most modern structured-grid models.

Arakawa C-grid: Variable Placement Within a Cell

- A close-up view of a single grid cell shows exactly where each variable is stored.
- Within each C-grid cell, scalar and velocity variables occupy distinct, physically meaningful positions.

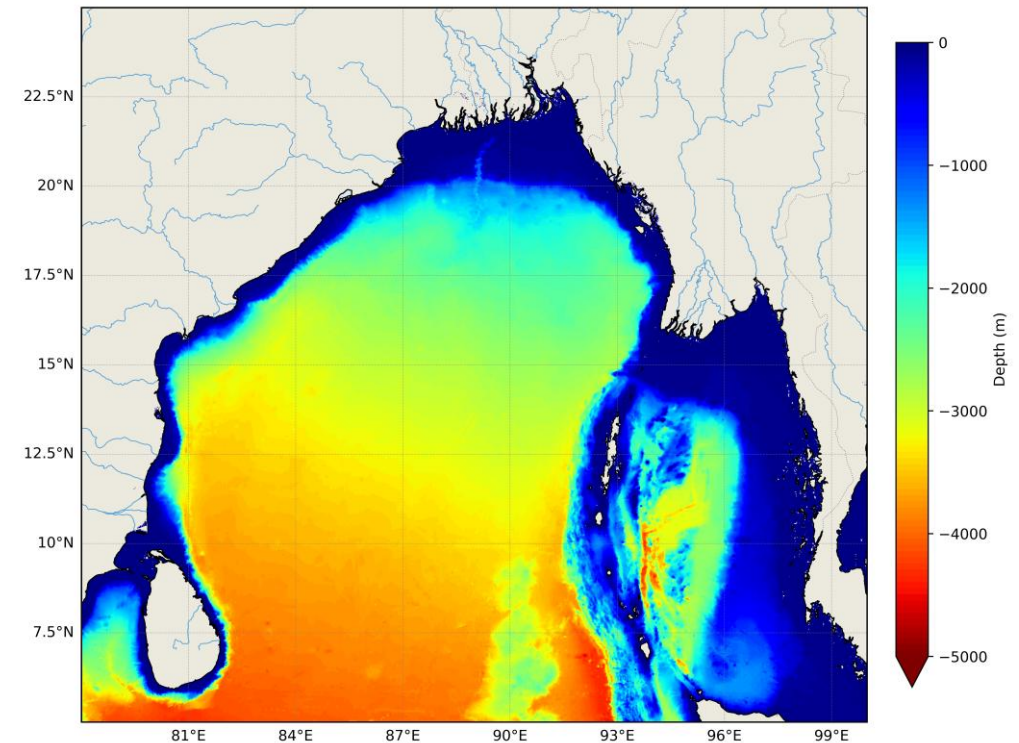
1. Scalar quantities, temperature, salinity, density, and pressure, are stored at the **cell center**.
2. The u velocity component is stored at the **east and west edges** of the cell, aligned with the direction it represents.
3. The v velocity component is stored at the **north and south edges** of the cell, following the same principle.



Bathymetry: Mapping the Ocean Floor

Bathymetry is the underwater equivalent of topography, a map of ocean depth rather than land elevation

- Bathymetry defines the ocean depth at every model grid point and sets the lower boundary of the computational domain.
- Global datasets such as GEBCO or ETOPO provide raw depth data, which is then interpolated onto the model's specific grid resolution.
- Coastal and shelf regions typically require finer bathymetry data than the open ocean, since depth changes far more rapidly near land.



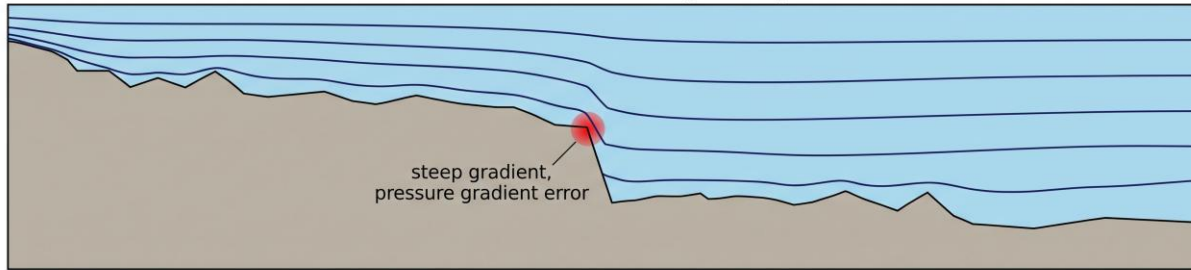
Bathymetry Smoothing: Taming Steep Slopes

Why Smoothing Necessary?

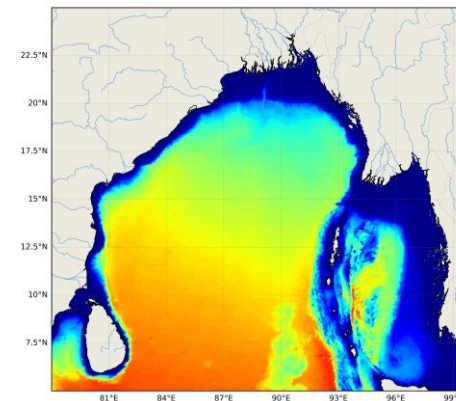
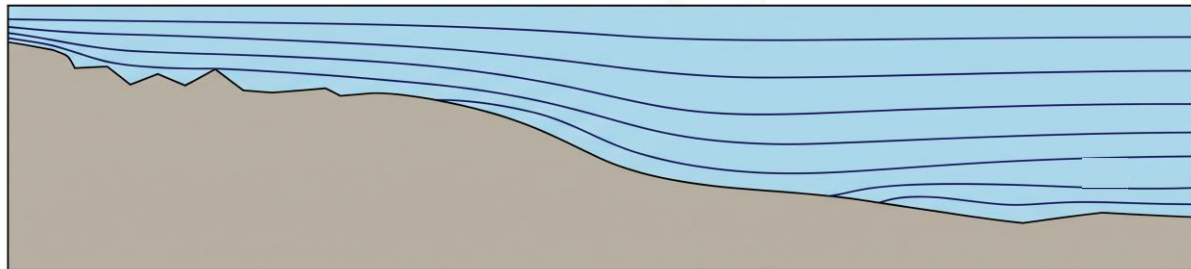
In terrain-following models, how steeply the seafloor changes directly affects numerical accuracy

- Sigma coordinates follow the seafloor shape, so vertical grid levels bend sharply wherever bathymetry changes sharply.
- Steep slopes cause pressure gradient errors, a well-known numerical artifact in terrain-following models.
- Smoothing reduces this error by limiting the steepness between adjacent grid points, measured using the r-factor, typically kept around ~ 0.25 .

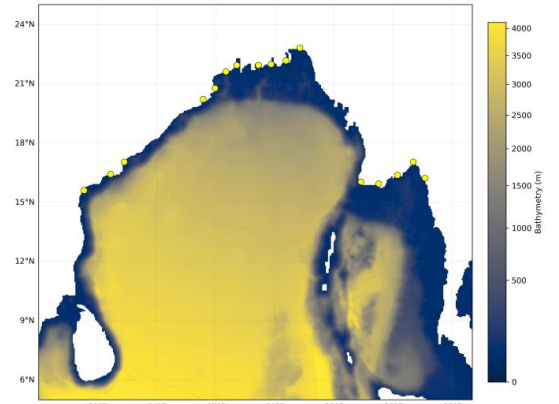
Unsmoothed bathymetry



Smoothed bathymetry



Unsmoothed Bathymetry



Smoothed Bathymetry

Time Stepping: Advancing the Model Forward in Time

Once the ocean state is known at one time step, the model must compute its state at the next.

There are two fundamentally different ways to do this.

$$\text{Explicit: } u^{n+1} = u^n + \Delta t \cdot F(u^n)$$

$$\text{Implicit: } u^{n+1} = u^n + \Delta t \cdot F(u^{n+1})$$

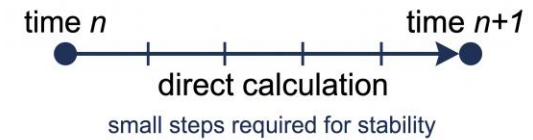
u^n = state at current time step

$u^{(n+1)}$ = state at next time step

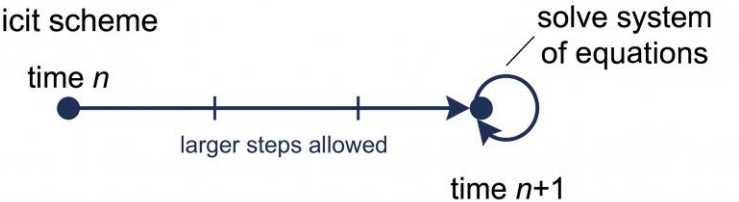
Δt = time step size

F = the governing equation's right-hand side (forcing/tendency terms)

Explicit scheme



Implicit scheme



- Explicit schemes compute the future state directly, making each step fast but requiring small time steps for stability.
- Implicit schemes solve a system involving future unknowns, allowing larger time steps at higher cost per step.

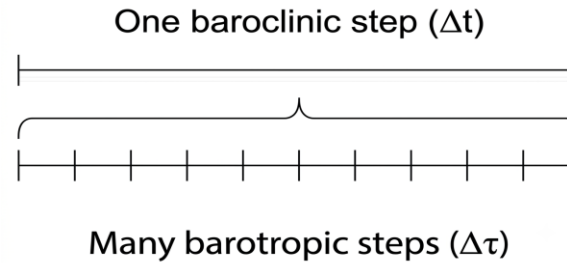
How the model calculates the next moment in time determines both its speed and its stability.

Mode Splitting

Fast surface waves and slow ocean currents need very different time steps, so CROCO integrates them separately.

$\Delta\tau \ll \Delta t$, typically $\Delta t / \Delta\tau \approx 30$ to 60

Δt = baroclinic (slow mode) time step
 $\Delta\tau$ = barotropic (fast mode) time step



- **Barotropic mode:** fast surface gravity waves (~ 200 m/s), governed by depth-averaged momentum and continuity, requiring a **very small time step for stability**.
- **Baroclinic mode:** slower, density-driven currents and stratified dynamics ($\sim 1-3$ m/s), the actual physics of scientific interest, which **can use a much larger time step**.
- Mode splitting integrates the barotropic mode many times within each single baroclinic step, then reconciles the two, avoiding the cost of running the entire model at the fast mode's tiny time step.

CFL Condition: A Speed Limit for Time Stepping

- ▶ A mathematical rule that links the time step directly to grid spacing and gravity wave speed.
- ▶ **For an explicit scheme to remain stable, a gravity wave must not cross more than one grid cell within a single time step.**

$$1D: \Delta t < \frac{\Delta x}{c}, \quad c = \sqrt{gH}$$

$$2D: \Delta t \leq \frac{1}{C_t} \left(\frac{1}{\Delta x^2} + \frac{1}{\Delta y^2} \right)^{-1/2}, \quad C_t = 2\sqrt{gH} + \bar{U}_{max}$$

Δt = time step

$\Delta x, \Delta y$ = grid spacing

g = gravitational acceleration

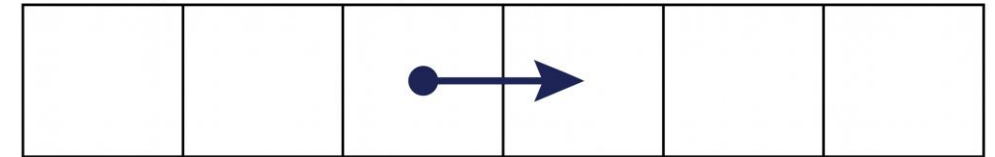
H = water depth

U_{max} = maximum current speed

C_t = total wave/flow speed used in the 2D criterion

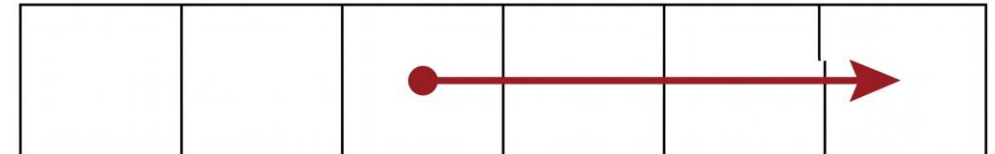
- Wave speed c represents how fast a surface gravity wave travels through depth H , the fastest signal the model needs to track.
- The time step must stay smaller than the time this wave takes to cross one grid cell, or the solution becomes unstable.

Stable (CFL satisfied)



signal moves less than one grid cell per time step

Unstable (CFL violated)



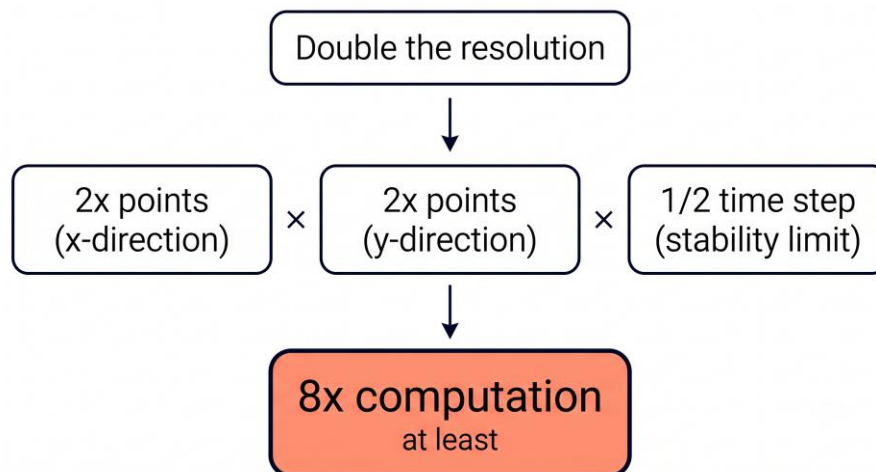
signal jumps more than one grid cell per time step

The CFL condition is not just theoretical — it directly determines how long a simulation takes to run.

Based on the 1D CFL condition, the table below shows the maximum allowable time step for different grid sizes and ocean depths.

Ocean depth (H)	$\Delta x = 1 \text{ km}$	$\Delta x = 10 \text{ km}$	$\Delta x = 100 \text{ km}$
10 m	~100 sec	~17 min	~3 hr (unrealistic H)
100 m	~30 sec	~5 min	~50 min
5000 m	~4.5 sec (unrealistic Δx)	~45 sec	~7.5 min

- A typical **coastal ocean model** ($\Delta x = 1 \text{ km}$, $H_{\text{max}} = 100 \text{ m}$, 1-year simulation) requires roughly **1 million time steps**
- A typical **climate ocean model** ($\Delta x = 100 \text{ km}$, $H_{\text{max}} = 5000 \text{ m}$, 100-year simulation) requires roughly **7 million time steps**



- ✓ Halving the time step doubles the number of steps needed.
- ✓ Combined with 2x grid points in each direction, doubling resolution results in at least an 8x increase in total computation.

Surface Forcing

Momentum Flux at the Sea Surface

Wind transfers momentum across the air-sea interface, driving surface currents, upwelling, and turbulent mixing.

$$\vec{\tau} = \rho_{air} C_d |\vec{U}_{10}| \vec{U}_{10}$$

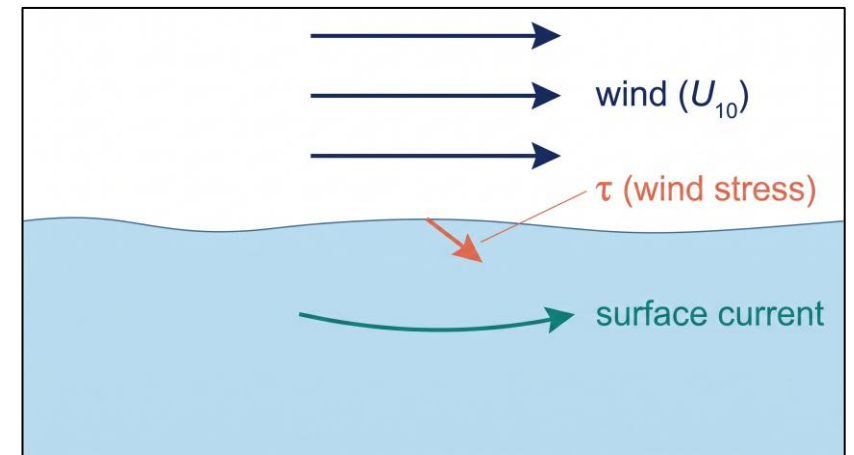
τ = wind stress

ρ_{air} = air density

C_d = drag coefficient

U_{10} = wind speed at 10 m

- Computed using the bulk aerodynamic formula, where C_d is a drag coefficient depending on wind speed and sea state.
- Stress scales with the square of wind speed, doubling wind speed roughly quadruples the stress.



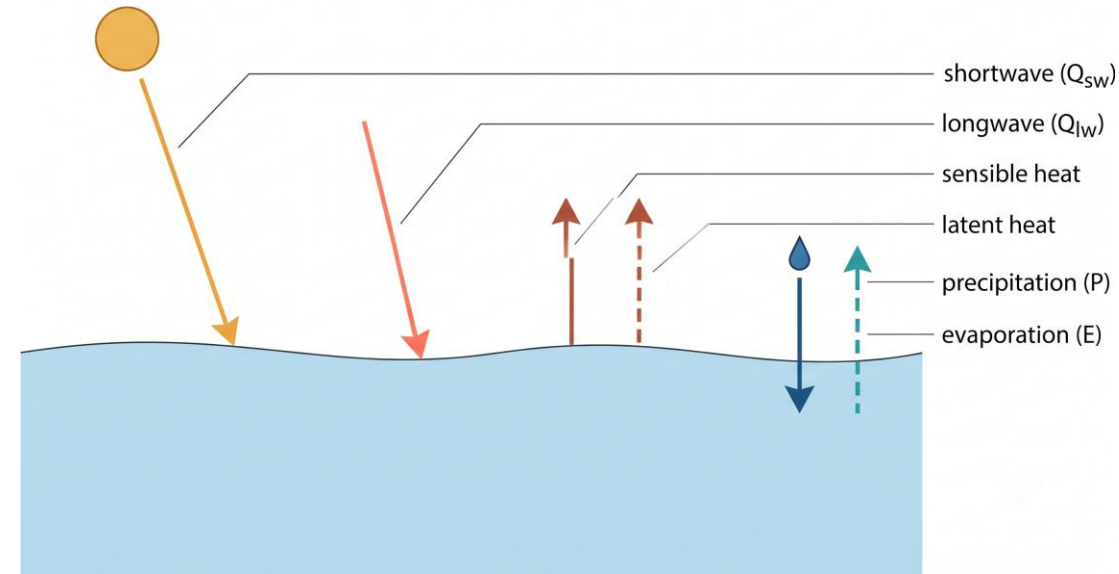
Heat and Freshwater Flux

Heat and freshwater exchange at the surface alter density and can drive convection, independent of wind.

$$Q_{net} = Q_{sw} + Q_{lw} + Q_{sensible} + Q_{latent}$$

$$F_w = P - E$$

- Net heat flux combines incoming **shortwave radiation**, **net longwave radiation**, and the **turbulent sensible** and **latent heat** exchanges between ocean and atmosphere.
- Freshwater flux, the balance between **precipitation and evaporation**, enters the salinity equation directly as a surface boundary condition.
- Act as buoyancy forcing that can trigger convective mixing, independent of wind-driven motion.

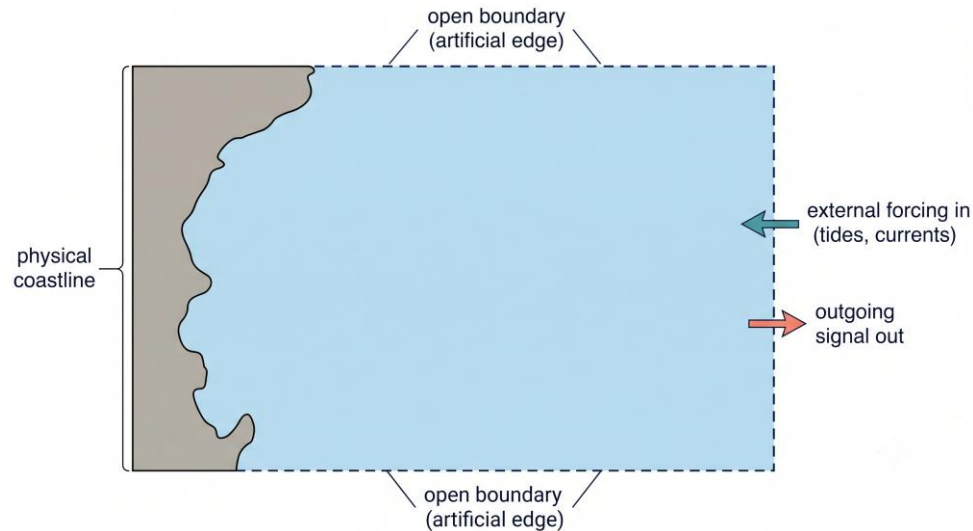


Open Boundaries

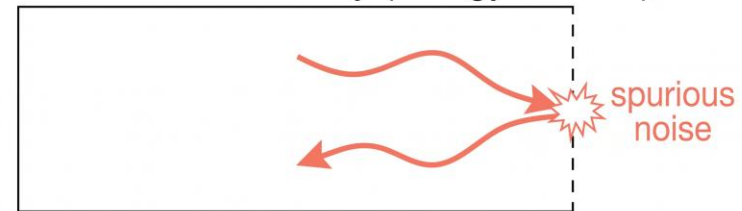
Why Open Boundaries Are Necessary?

A regional model simulates only part of the ocean, so its artificial edges demand careful mathematical treatment.

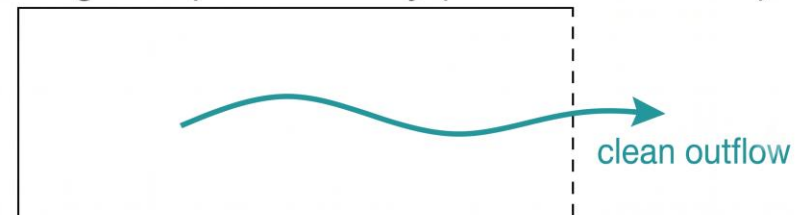
- Regional models cover only part of the ocean, so their edges are artificial, not real coastlines.
- Left untreated, these edges reflect energy back into the domain, creating spurious noise and instability.
- A good open boundary condition lets outgoing signals leave freely, while still allowing external forcing, such as tides or large-scale currents, to enter.



Untreated boundary (energy reflects)



Well-designed open boundary (radiation condition)



Nudging: Relaxing Toward External Data

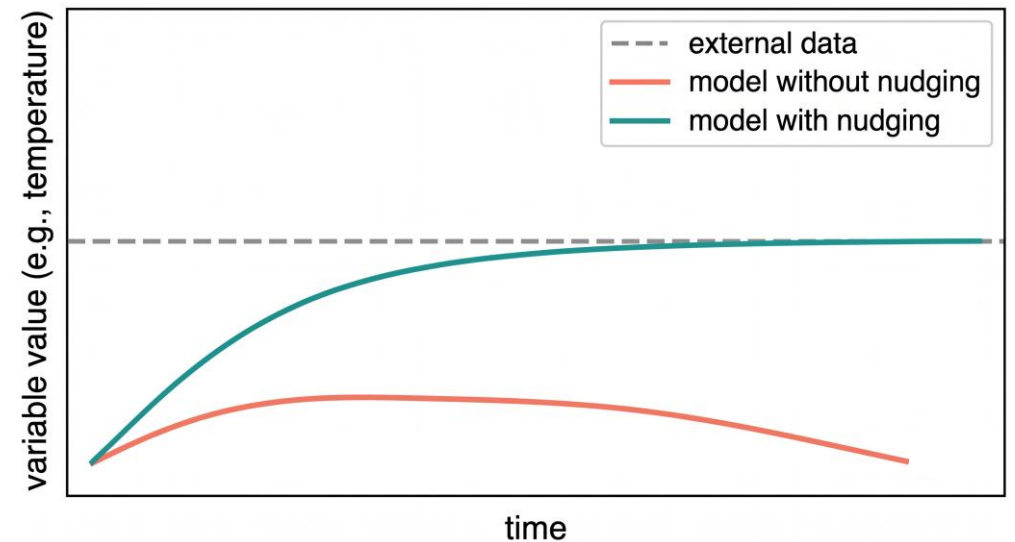
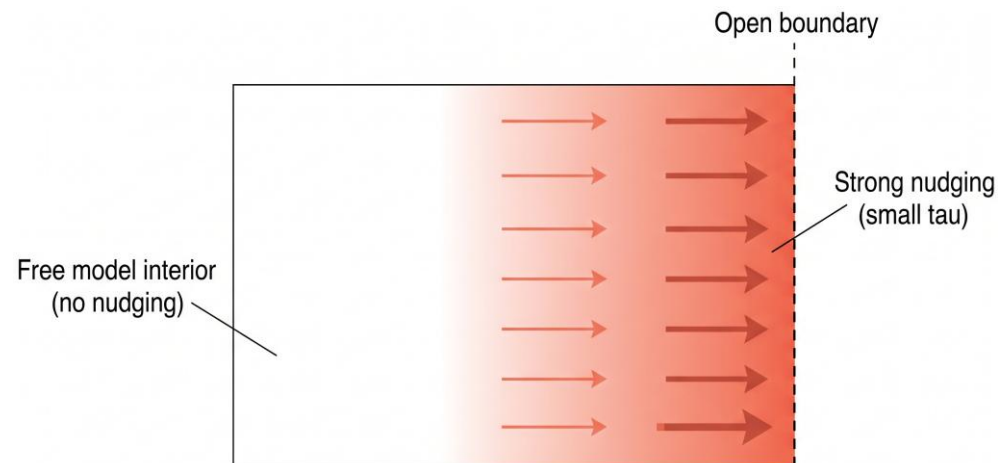
Near the boundary, the model solution is gently pulled toward external data rather than evolving freely

$$\left. \frac{\partial \phi}{\partial t} \right|_{nudging} = -\frac{1}{\tau(x)} (\phi_{model} - \phi_{external})$$

ϕ = model variable

$\tau(x)$ = nudging timescale (position-dependent, shorter means stronger relaxation)

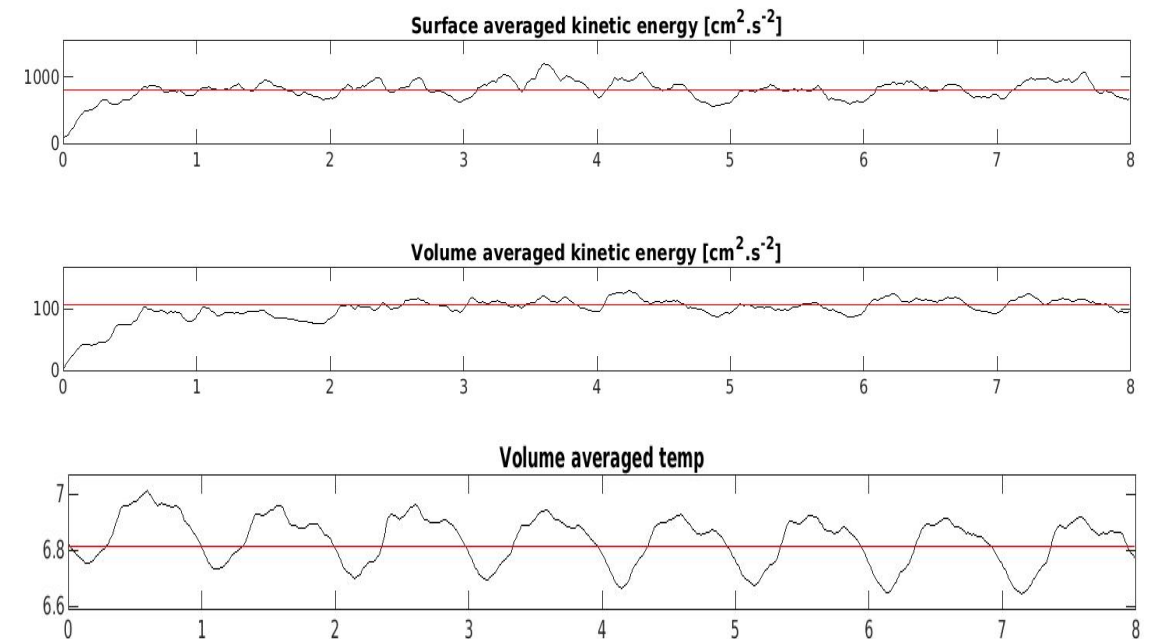
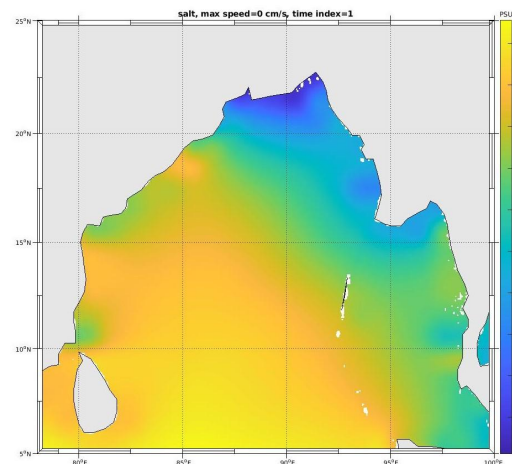
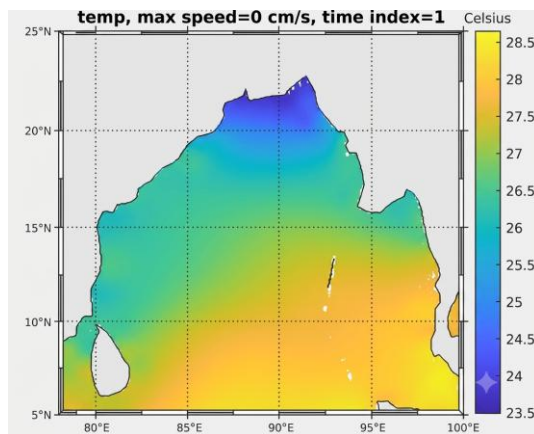
- Nudging adds a relaxation term that pulls the model solution toward external data, such as a parent model or observations, near the boundary.
- The nudging timescale $\tau(x)$ sets the relaxation strength, short near the boundary for strong forcing, and long toward the interior, where forcing fades to near zero.
- A properly tuned nudging zone stabilizes the boundary without suppressing the interior's own independent dynamics.



Initial Conditions

Every model needs a starting snapshot of the ocean before it can move forward in time.

- Initial conditions define the **ocean's state, temperature, salinity, currents, and sea surface height**, at every grid point **at time zero**, giving the equations a starting point to evolve from.
- If the starting state is not well balanced, the model needs a spin-up period, time to adjust away from that imbalance, before its output becomes physically realistic.
- These conditions usually come from a coarser global reanalysis or climatology, interpolated onto the model's own grid, resolution, and vertical coordinate.



Vertical Mixing

Turbulent mixing happens at scales far smaller than the grid can resolve, so it must be parameterized.

$$\overline{w'\phi'} = -K_v \frac{\partial \phi}{\partial z}$$

$w'\phi'$ = turbulent vertical flux
 K_v = vertical eddy diffusivity
 ϕ = tracer or momentum variable

The turbulent flux is modeled as a diffusive process, but K_v itself is not constant, it must come from a closure scheme.

Feature	KPP (K-Profile Parameterization)	GLS / Mellor–Yamada Family
Type & Approach	Diagnostic; computes turbulence profile directly from surface forcing	Prognostic; solves equations for turbulent kinetic energy (TKE) and turbulence length scale
Physics	Empirical boundary-layer parameterization	More physically based turbulence closure
Computational Cost	Low (fast)	Higher (more computationally expensive)
Accuracy	Good for most large-scale ocean simulations	Better for representing complex turbulent processes
Main Advantage	Fast, stable, and widely used (default in ROMS/CROCO)	More realistic and physically consistent turbulence representation

- All three schemes simulate mixed layer depth deeper than observed, with the largest bias during the Indian Summer Monsoon (July–August).
- The schemes cluster close to each other but consistently diverge from observations, showing the bias comes from shared modeling assumptions, not one scheme's flaw.
- No single scheme performs best across all four regions or all months, the "best" choice depends on location and season.

No Single Scheme Wins Everywhere

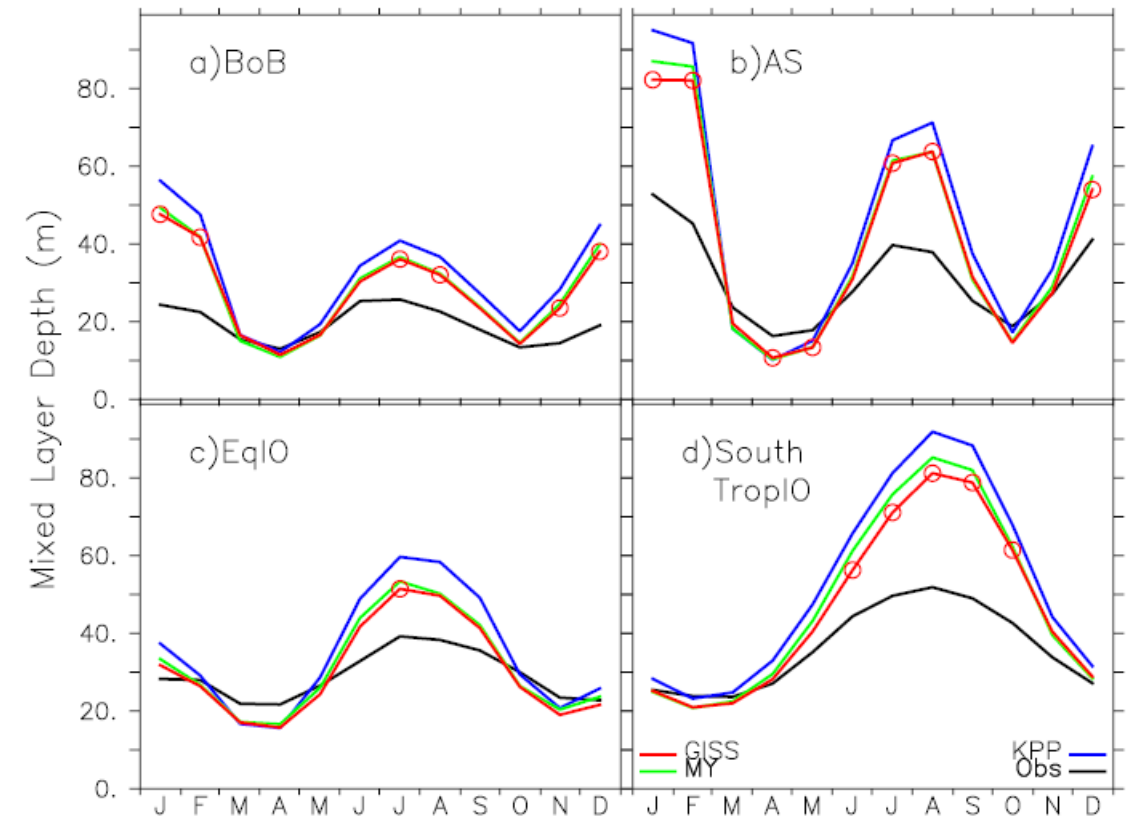


Figure: Seasonal cycle of mixed layer depth simulated by three mixing schemes (KPP: blue; GISS: red; MY: green) versus observations (black) across four regions of the tropical Indian Ocean (Pottapinjara & Joseph, 2022)

Module 04

Introduction to CROCO Model



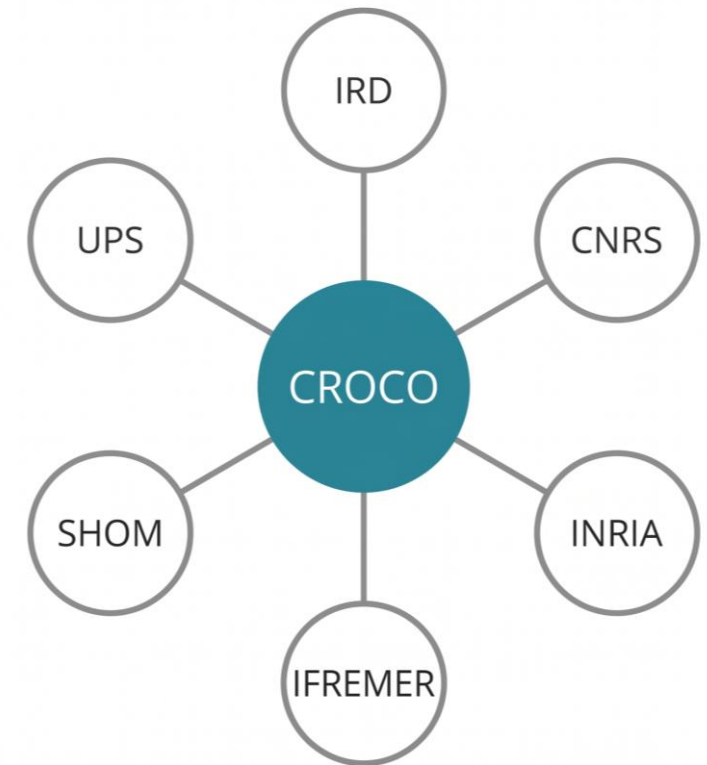
Coastal & Regional Ocean Community Model



Coastal & Regional Ocean Community Model

A community-driven model built to unify coastal ocean modeling in France and beyond.

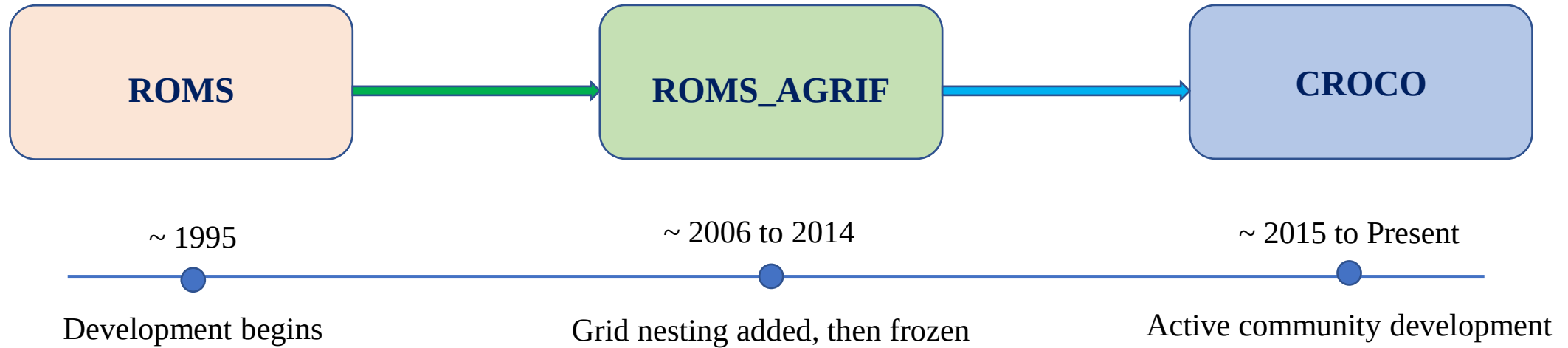
- Born from a need to unify fragmented coastal modeling efforts across French research institutions.
- Developed by a consortium: IRD, CNRS, INRIA, IFREMER, SHOM, and UPS, with historical ties to UCLA and Rutgers (original ROMS developers).
- Philosophy: high-level numerical schemes, realistic complex modeling, and openness to the broader research community.



** Historical ties to UCLA and Rutgers, the original ROMS developers*

The Lineage of CROCO

CROCO did not start from scratch, it evolved through two decades of community development.



- **ROMS** (Regional Ocean Modeling System) began in the mid 1990s as a terrain-following, split-explicit ocean model.
- **ROMS_AGRIF** added adaptive mesh refinement (AGRIF), enabling nested, multi-resolution grids within a single simulation.
- **CROCO** extended this lineage further, adding non-hydrostatic capability, sediment dynamics, and coupling with atmosphere and wave models.

CROCO Ocean Community Forum

CROCO is not just a model, it's an active, expanding research community.

- More than a thousand registered users worldwide, with around half coming from countries in the Global South.
- Active users span Europe, the Americas, Africa, and Asia, drawn by the model's performance, accuracy, and ease of access.
- Support is community-driven through an open forum, alongside regular training courses

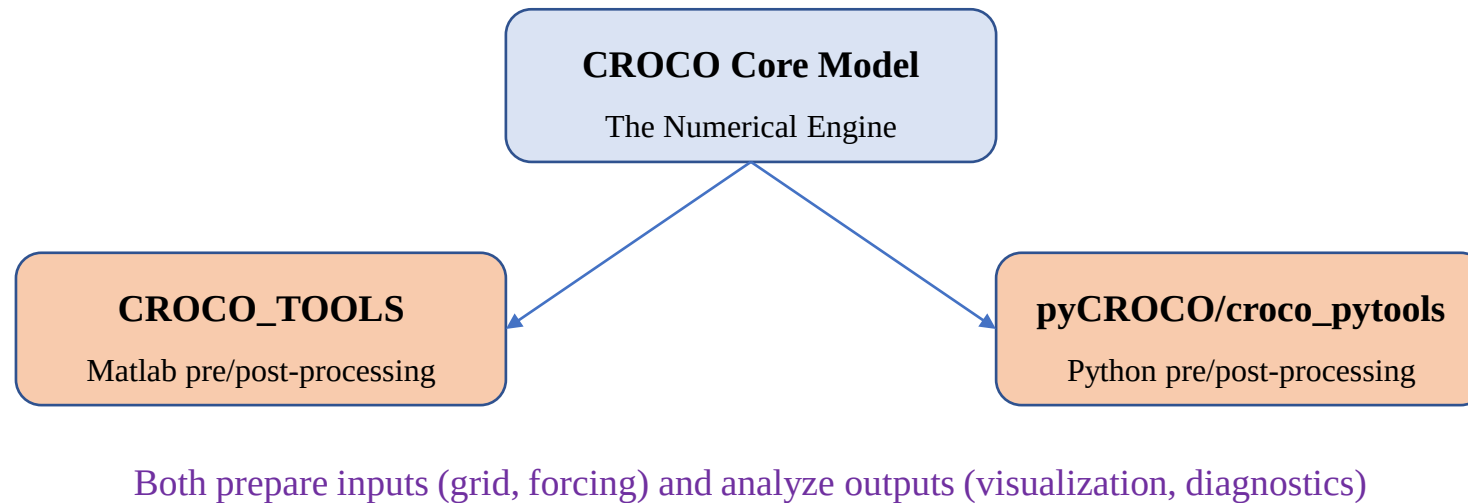
The image shows two side-by-side screenshots. The left screenshot is the CROCO website homepage, featuring a large ocean map background with the text 'CROCO Coastal and Regional Ocean COmmunity model'. The navigation bar includes links for HOME, ABOUT, DOWNLOAD & DOCUMENTATION, TRAININGS, WEBINARS AND MEETINGS, CONTACTS, HOW TO CITE?, and a 'FORUM' link which is circled in red. A red arrow points from this 'FORUM' link to the right screenshot. The right screenshot is the CROCO community forum interface, showing a list of topics with columns for Topic, Replies, Views, and Activity. The topics listed include a welcome message, a question about RIVER_RAMP, a case study about BENGUELA, a postdoc position, and questions about model parameters and file formats.

Topic	Replies	Views	Activity
Welcome to CROCO-Ocean community forum! 🎉	0	674	Dec 2023
Option RIVER_RAMP	4	213	2h
BENGUELA case fails with croco-GPU with CROCO_QH or NBQ	0	10	2d
Postdoc position on ocean mixing	0	8	3d
Use wkb_wwave and parameters in croco_inter.in	1	24	10d
How use different files as fluxes	8	261	16d

Community Forum Link: <https://forum.croco-ocean.org/>

CROCO Ecosystem

CROCO is a complete ecosystem, the **core model** plus tools that **prepare, run, and analyze simulations**.

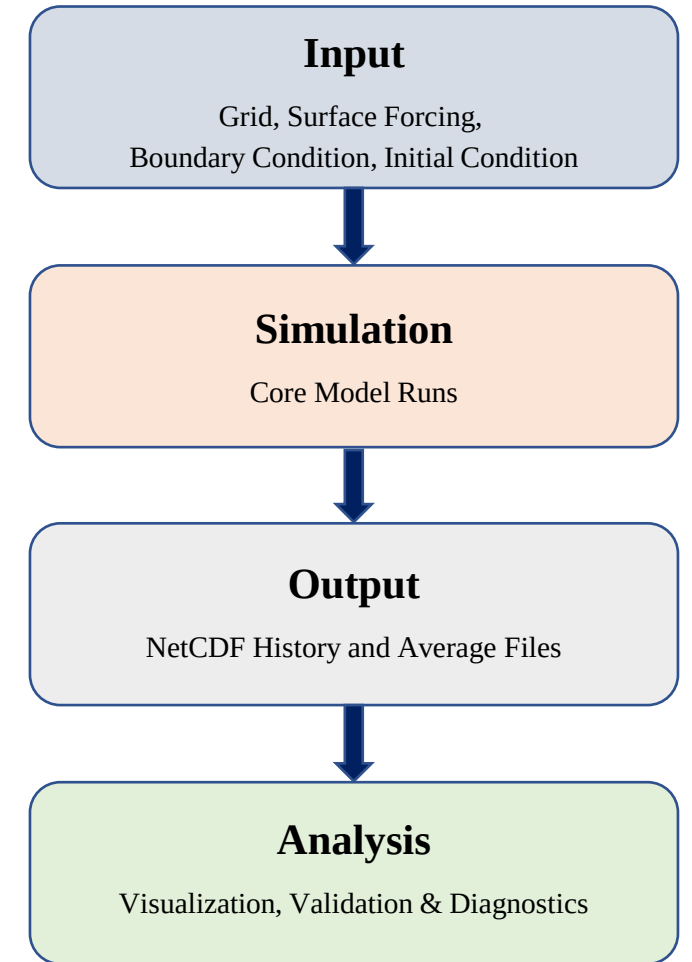


- The **CROCO core model** is the numerical engine which solves the governing equations
- **CROCO_TOOLS** (Matlab-based) and **pyCROCO/croco_pytools** (Python-based) handle **pre-processing** (grid, forcing, boundary files) and **post-processing** (visualization, analysis).
- Together, these tools turn raw configuration into a runnable simulation, and simulation output into interpretable results.

CROCO Workflow

Every CROCO simulation follows the same four-stage pipeline, regardless of the application.

1. **Input:** Grid, bathymetry, initial and boundary conditions, atmospheric forcing, all prepared before the model runs.
2. **Simulation:** The core model integrates the governing equations forward in time, producing history and average files.
3. **Output:** The model produces raw NetCDF files, history and average outputs containing the simulated ocean state.
4. **Analysis:** This output is visualized, validated, and interpreted using CROCO_TOOLS/pyCROCO or through other analyzing tools.

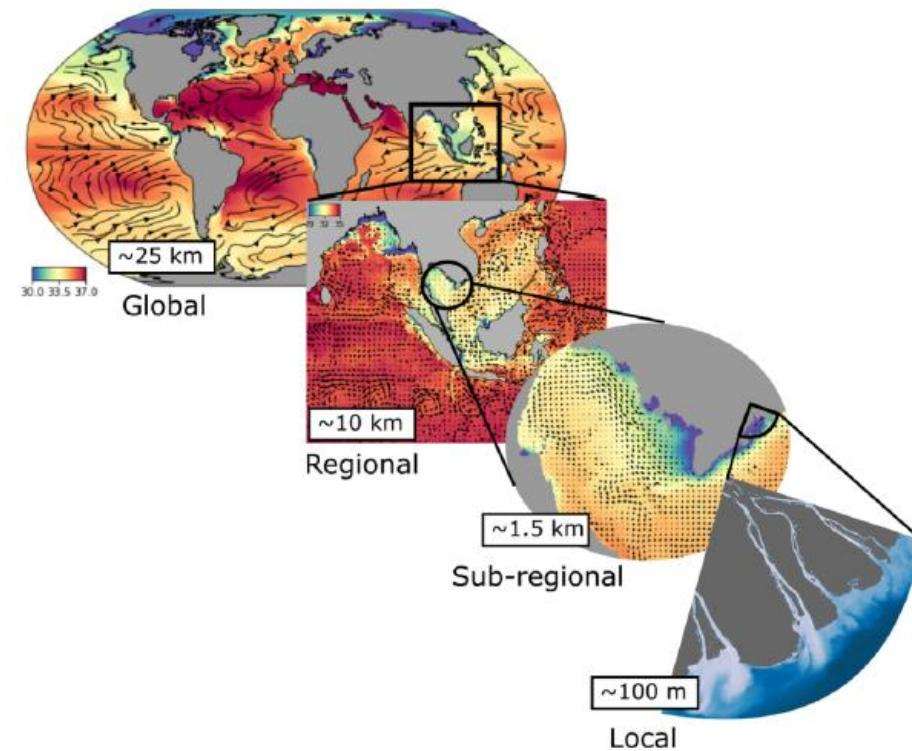
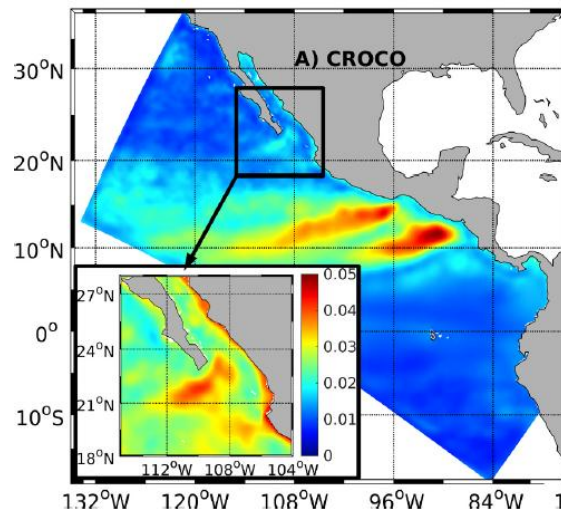
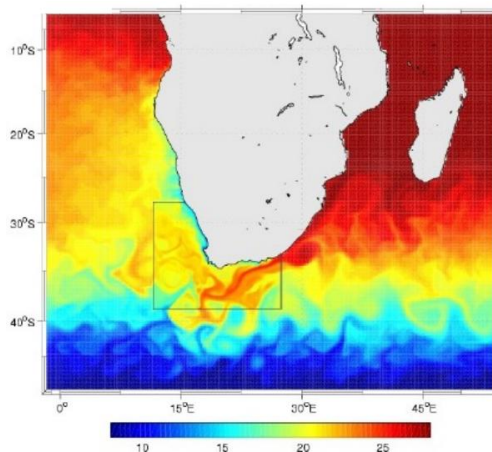


High-resolution Coastal & Regional Modeling

Global ocean models typically run at tens of kilometers resolution, too coarse to capture coastal currents, estuaries, or narrow straits.

CROCO is designed to resolve the fine-scale processes that coarse global models miss

- CROCO targets regional and coastal domains, from a few kilometers down to sub-100-meter resolution when needed.
- Finer resolution resolves mesoscale eddies, upwelling filaments, and coastal boundary currents that coarser models smooth over.
- This focus on coastal and regional scales, rather than global coverage, is what CROCO is fundamentally built for.



(Herrera-Becerril et al., 2025)

(Holt et al., 2025)

Non-hydrostatic Capability

When vertical motion matters too much to ignore, CROCO can solve the full equations of motion.

The hydrostatic approximation assumes vertical acceleration is negligible, valid when horizontal scales dominate over vertical ones.

CROCO also offers a non-hydrostatic, non-Boussinesq mode (NBQ) for situations where that assumption breaks down.

$$\text{Hydrostatic: } \frac{\partial p}{\partial z} = -\rho g$$

$$\text{Non-hydrostatic (NBQ): } \frac{\partial \rho w}{\partial t} + \nabla \cdot (\rho w \vec{v}) - \rho f u = -\frac{\partial P}{\partial z} - \rho g + \lambda \frac{\partial(\nabla \cdot \vec{v})}{\partial z} + \mathcal{F}_w + \mathcal{D}_w$$

w = vertical velocity

λ = bulk viscosity

NBQ = Non-Boussinesq,
non-hydrostatic mode

(the vertical momentum equation is solved in full, rather than reduced to a balance between pressure and gravity)

- NBQ mode retains the full vertical momentum equation, capturing vertical accelerations that the hydrostatic approximation discards.
- To avoid the costly 3D pressure-solving system this normally requires, CROCO integrates acoustic waves explicitly using a split time-stepping approach, separating fast and slow dynamics.
- This makes non-hydrostatic simulation computationally practical, useful for small-scale, strongly vertical processes like internal waves, convection, or flow over steep topography.

Nesting Capability

A finer grid embedded inside a coarser one, focusing computational power where it's needed most

Why Nesting Is Needed

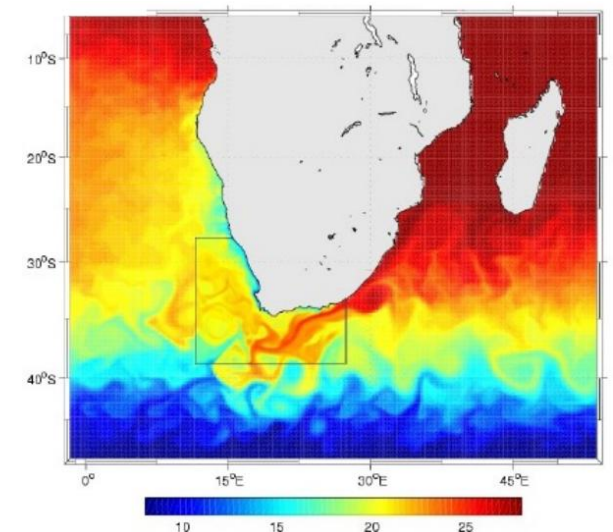
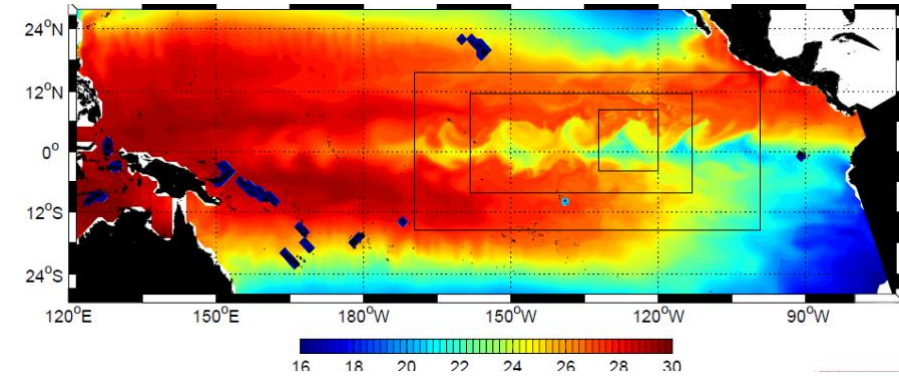
- ✓ Coastal processes require much higher resolution than open-ocean models
- ✓ Nesting enables local refinement without increasing the cost of the whole domain

Nesting Method

- ❑ CROCO uses AGRIF (Adaptive Grid Refinement in Fortran)
- ❑ Supports multiple nested grids and multiple nesting levels
- ❑ Employs static grid embedding with flexible refinement ratios

Working Procedure

- A **parent (coarse) grid** provides boundary conditions to a **child (fine) grid**
- Child grid runs with **smaller time steps** to satisfy numerical stability
- Nesting can be one-way (parent feeds child only) or two-way (child feeds information back into the parent)



Parallel Computing

Realistic ocean simulations are too large for a single processor, so the domain is divided and solved in parallel.

Why Parallelization?

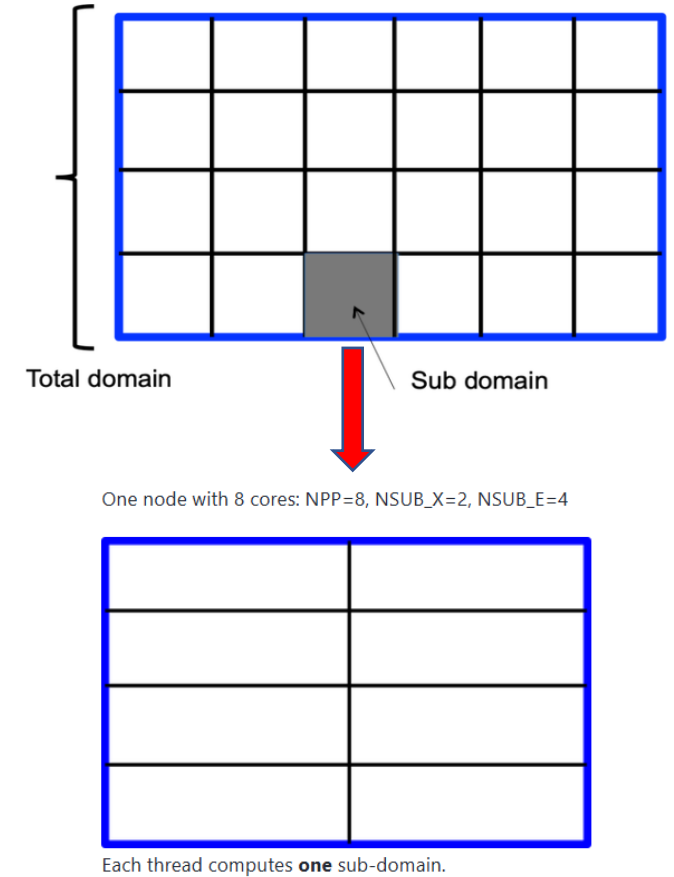
- Coastal models require high resolution and large domains
- Parallel computing reduces runtime and improves scalability

How it Works?

- The model domain is split into smaller sub-domains (tiles), each assigned to a separate processor core.
- Processors exchange boundary information at every time step, keeping neighboring tiles physically consistent.

Supported Parallel Strategies

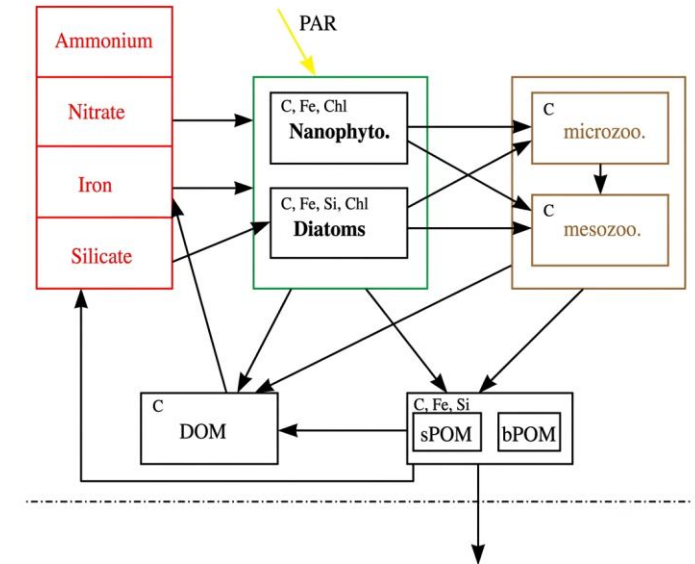
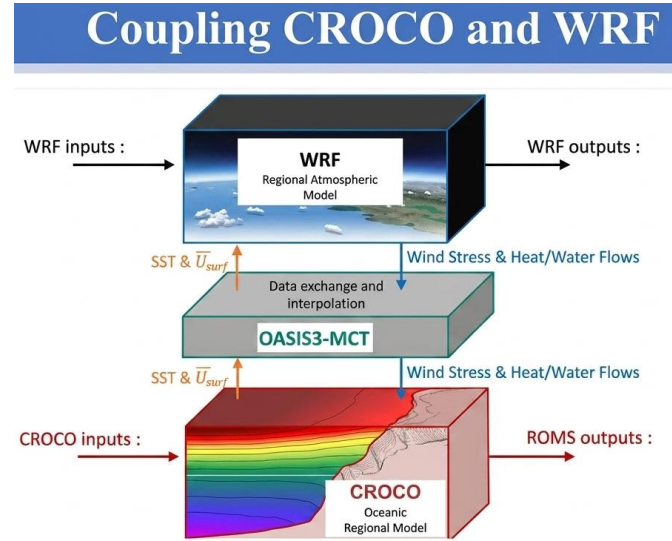
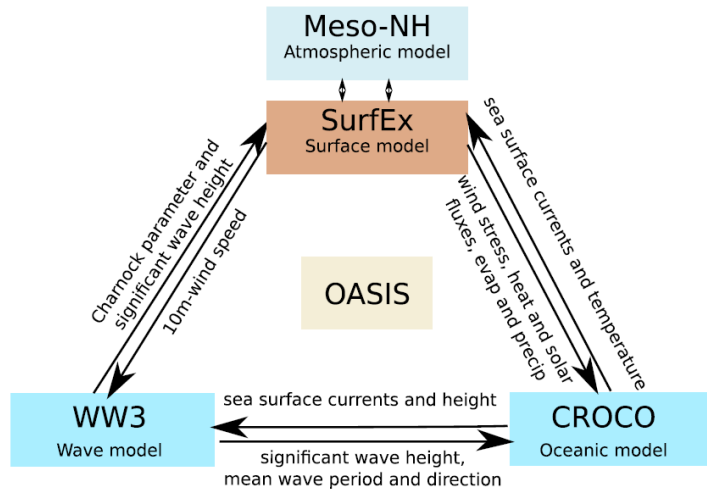
- i. MPI – Distributed memory parallelism
- ii. OpenMP – Shared memory parallelism
- iii. OpenACC – GPU acceleration



This scalability is what makes high-resolution, basin-scale, or long climate simulations computationally feasible.

Coupling with Other Model

CROCO can exchange information with atmosphere, wave, and biogeochemical models to capture processes ocean physics alone cannot.



- **WRF** (atmosphere) and **WW3** (waves) exchange fields with CROCO at runtime, coupling ocean-atmosphere and ocean-wave dynamics beyond simple one-way forcing.
- **PISCES** and **NPZD** are biogeochemical models coupled to CROCO, simulating nutrients, plankton, and carbon cycling driven by the physical circulation.
- Coupling is handled through a standard framework (OASIS), allowing each component model to run independently while exchanging fields at set intervals.

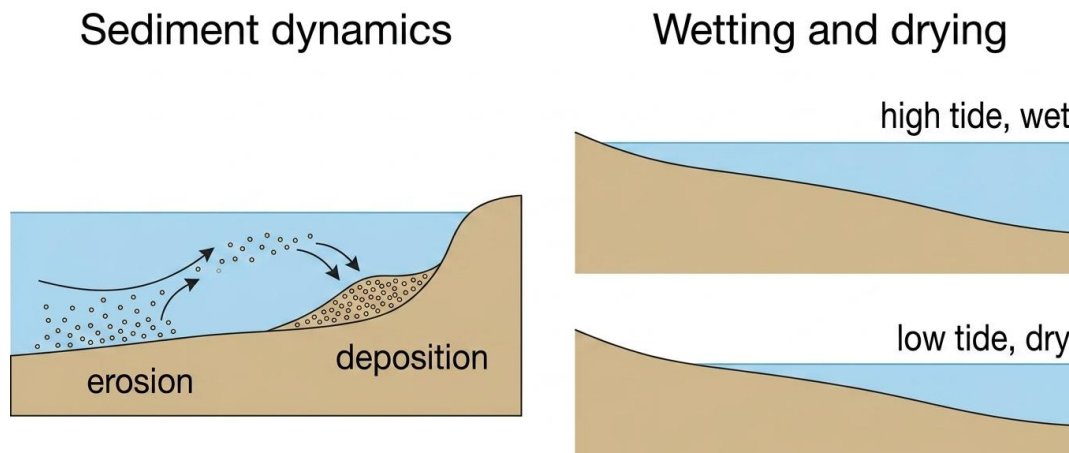
Sediment Dynamics & Wetting-Drying

Sediment Dynamics

- Coastal flows carry and deposit sediment, gradually reshaping the seabed itself.
- CROCO simulates erosion, transport, and deposition of sediment, feeding back into bathymetry over time.
- Useful for studying river deltas, harbor siltation, and coastal erosion under storms or long-term climate change.

Wetting and Drying

- Tidal flats and estuaries are sometimes underwater, sometimes exposed, a boundary that moves with the tide.
- Grid cells can dynamically switch between "wet" and "dry" as water depth drops below a threshold, rather than assuming a fixed coastline.
- Essential for accurately simulating tidal flats, mangroves, and shallow estuarine systems.



Scientific Applications of CROCO

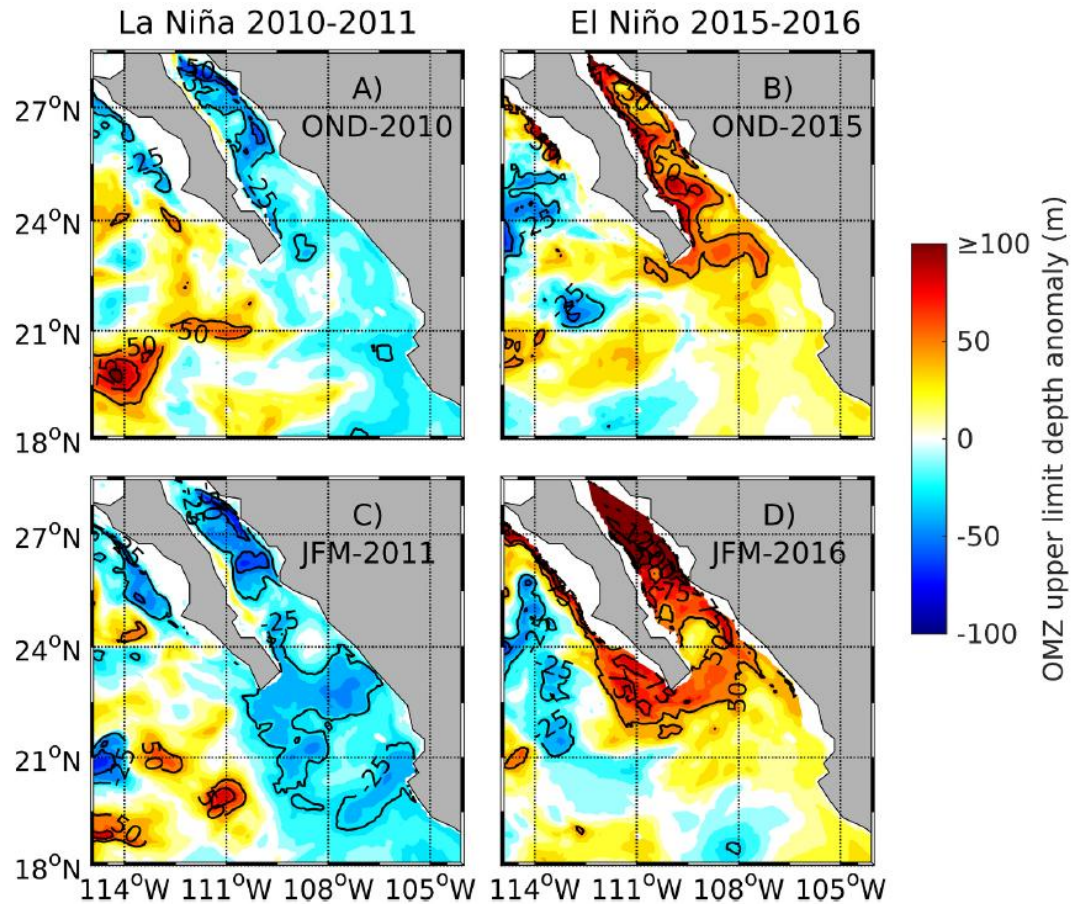


Figure-1: Comparison of the OMZ upper limit ($\text{DO} = 44 \mu\text{mol L}^{-1}$) depth anomaly between La Niña 2010–2011 (left column: a,c) and El Niño 2015–2016 (right column: b,d) during fall and winter using croco+pisces couple model. (Herrera-Becerril et al., 2025)

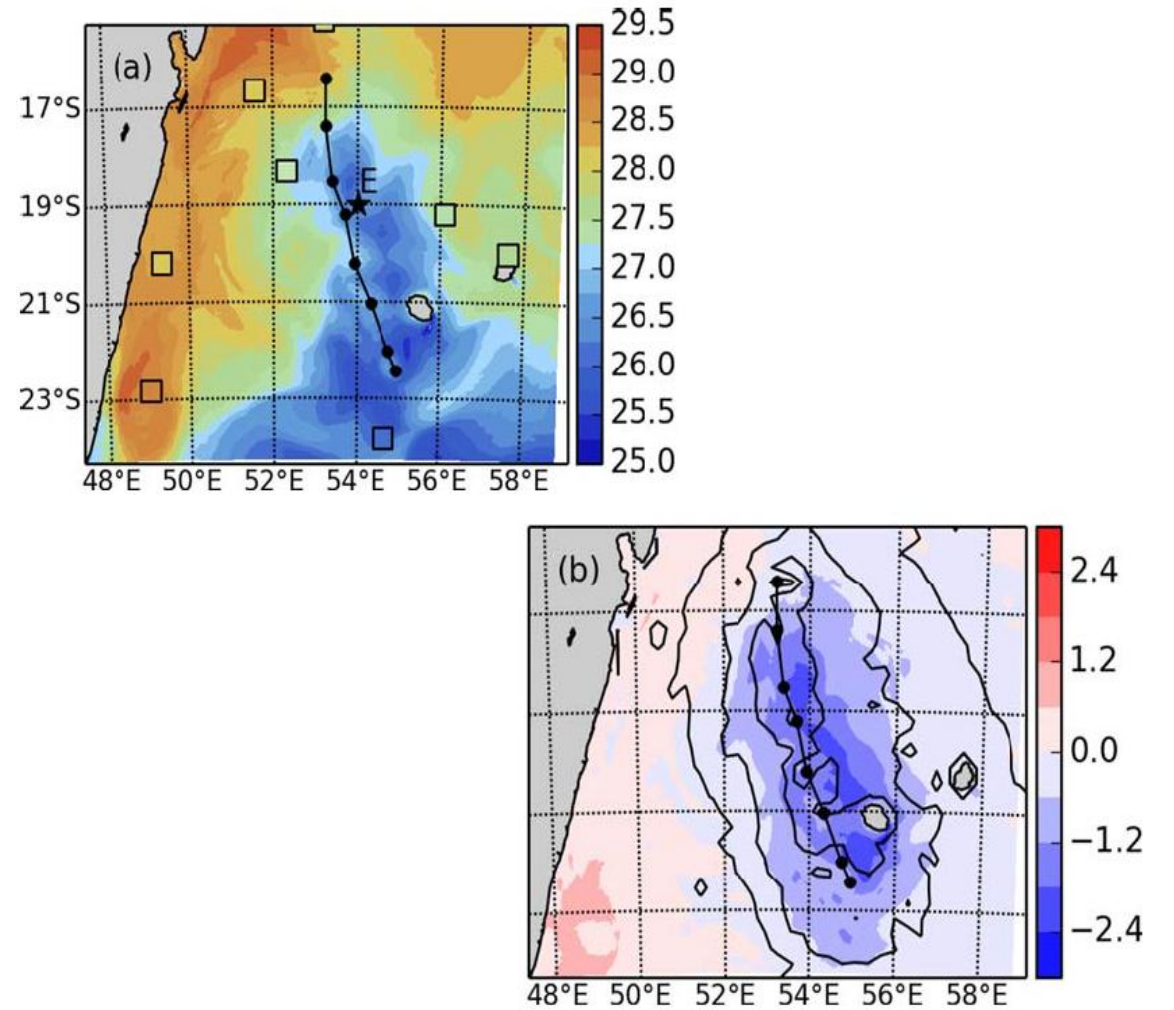


Figure-2: CROCO-simulated sea surface cooling wake along the track of tropical cyclone Bejisa, with the coldest anomalies exceeding 2°C on the left side of the storm track (Pianezze et al., 2018).

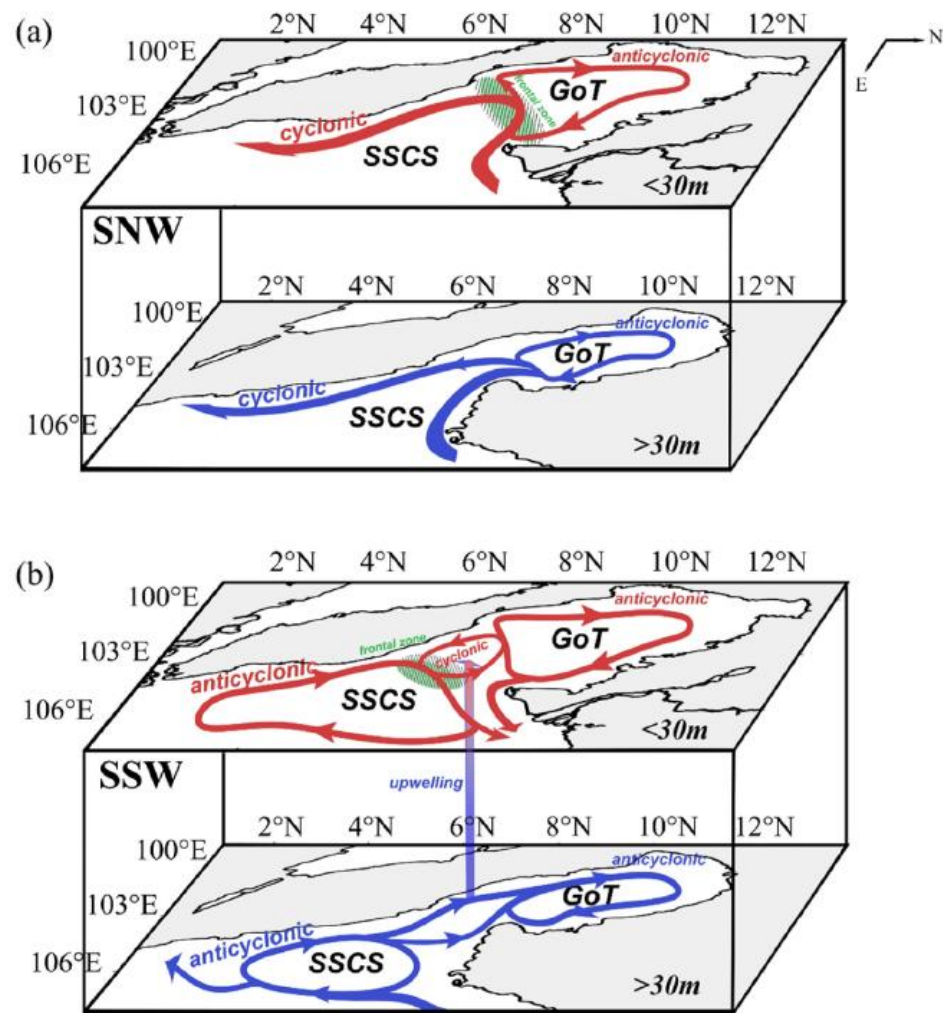


Figure-3: Schematic of CROCO-simulated three-dimensional circulation in the Gulf of Thailand and Southern South China Sea during the northeast (SNW) and southwest (SSW) monsoon periods, showing reversing upper- and lower-layer water exchange (Wu et al., 2026).

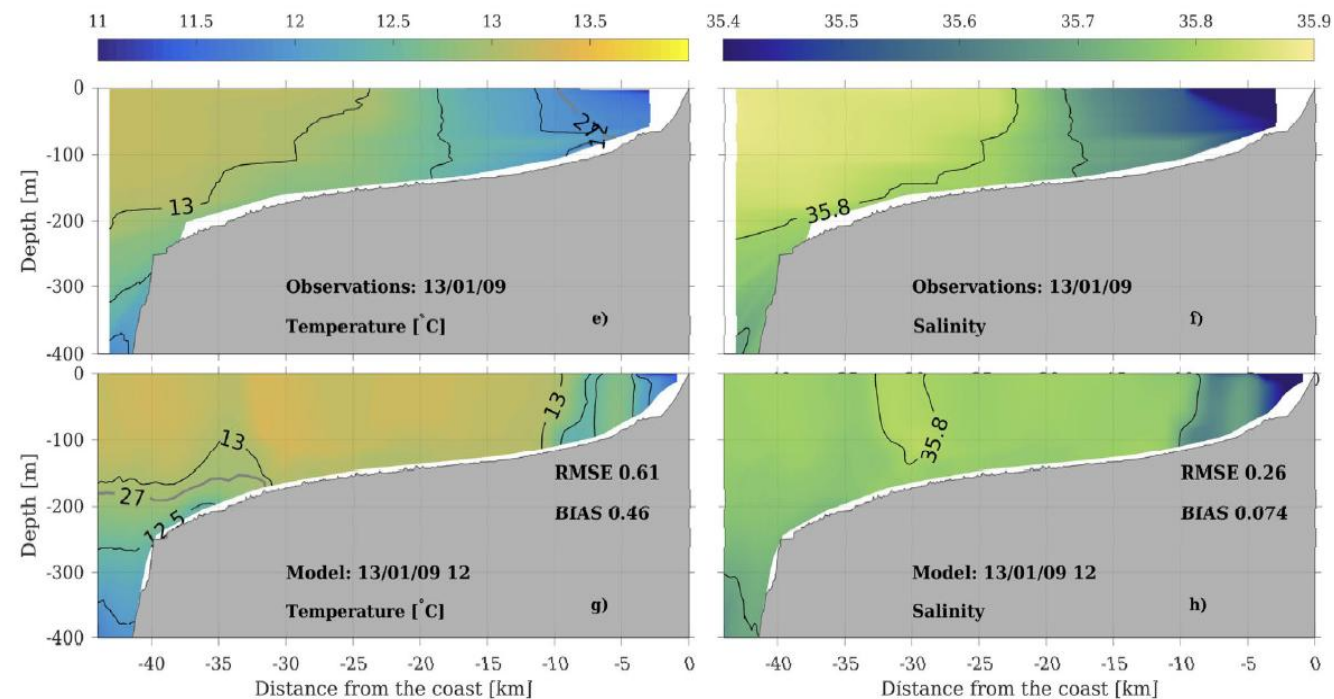


Figure-4: CROCO-simulated temperature and salinity cross-section during a coastal upwelling event, showing realistic doming of the 27.0 kg/m³ isopycnal in NW Iberian continental shelf (Viitak et al., 2026)

Module 05

CROCO Model Configuration & Input

Model Domain Design

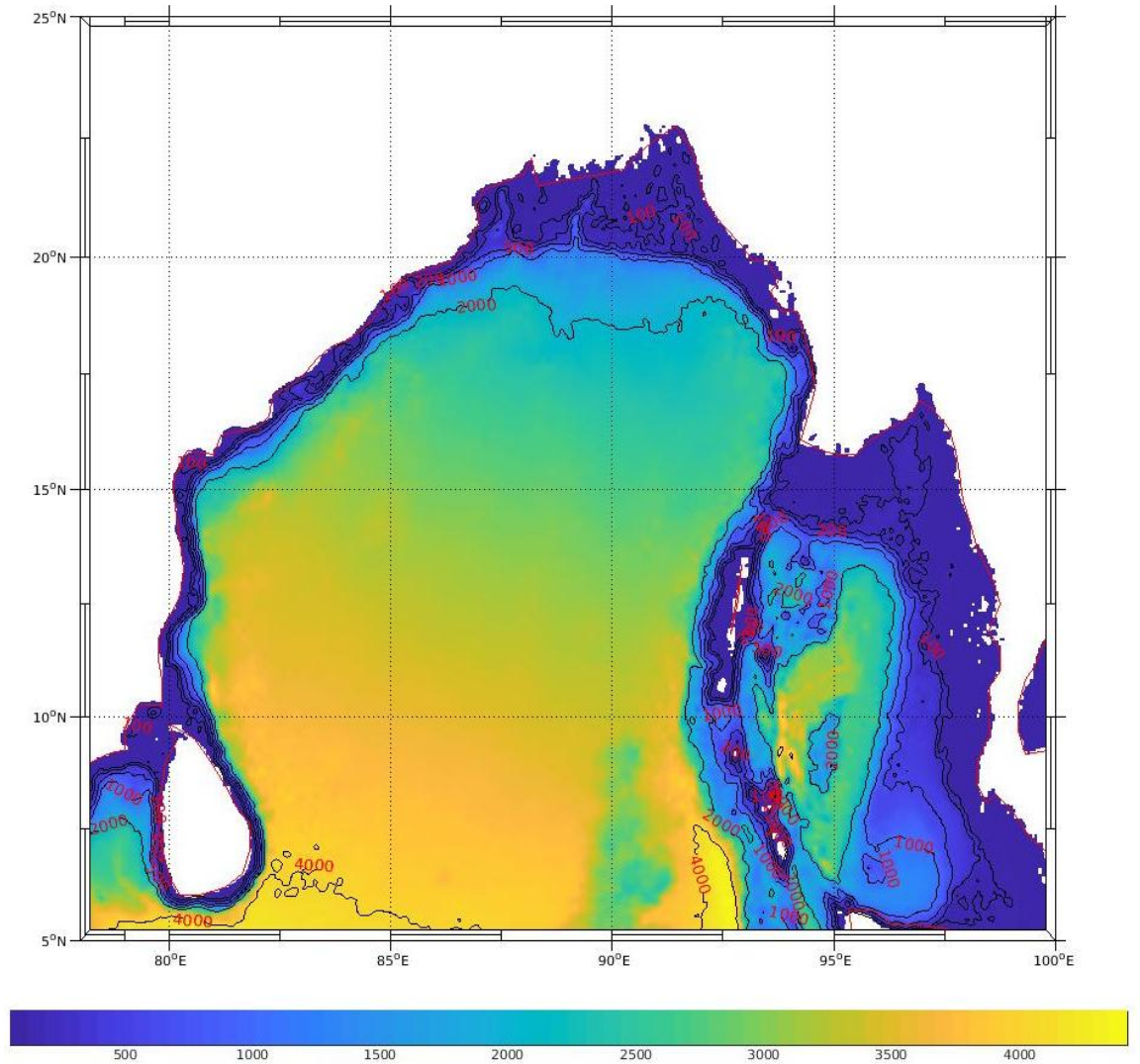
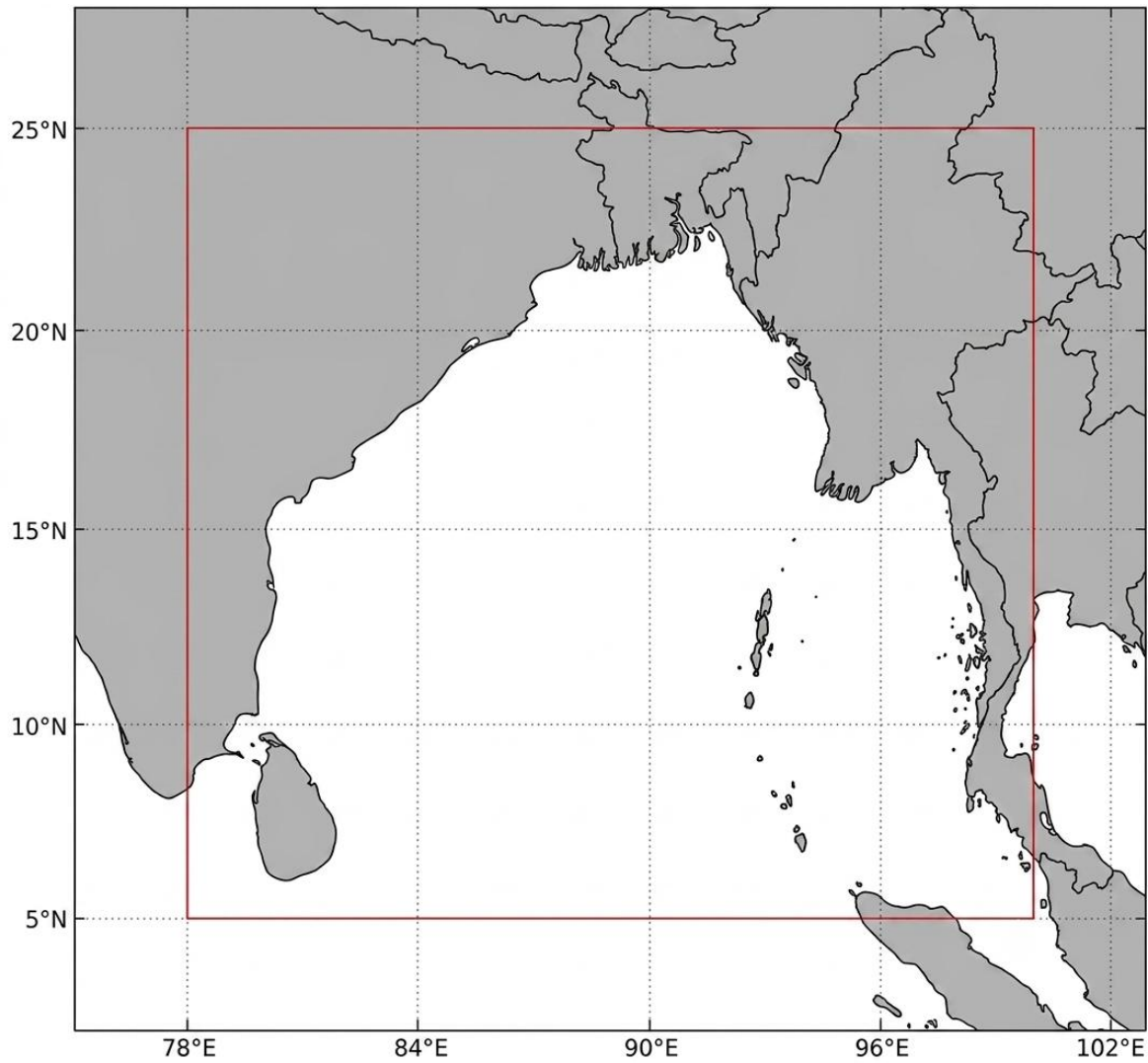
Domain Selection

Every configuration decision starts with one question, what area does the simulation actually need to cover.

A domain that is too small misses important large-scale influences.

Too large wastes computational resources on areas outside the study interest.

- Domain boundaries should sit far enough from the region of scientific interest that open boundary artifacts (covered in Module 3), don't contaminate the results.
- Boundaries are best placed in relatively quiet, deep water away from complex topography or strong currents, where nudging and radiation conditions behave more reliably.
- The domain must be large enough to include the physical processes driving local dynamics, such as a nearby current system, monsoon wind field, or river plume source.

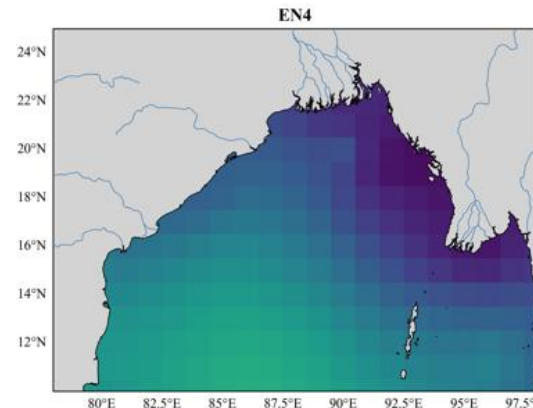
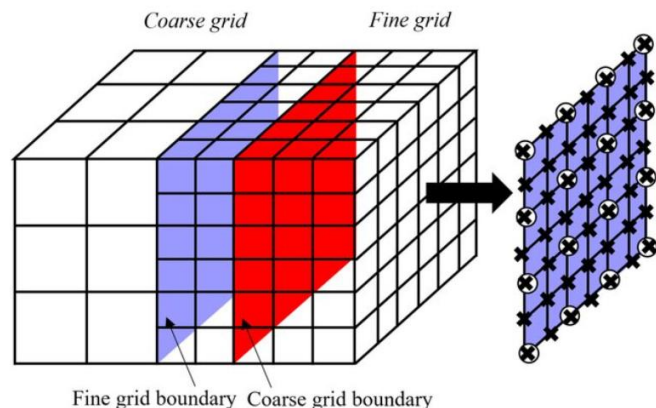


CROCO Model Domain Selection (Bay of Bengal Region)

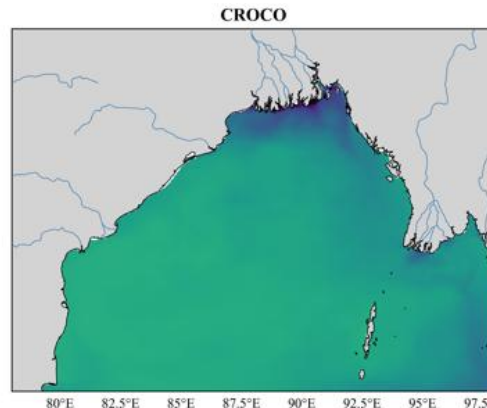
Horizontal Resolution & Vertical Levels

Once the domain is set, resolution decides what physical processes the model can actually see.

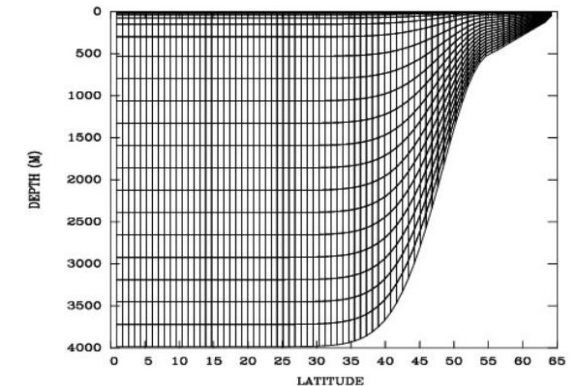
- **Horizontal resolution** determines whether the model resolves mesoscale eddies ($\sim 10\text{-}100\text{ km}$), coastal upwelling filaments, or narrow straits, finer grids capture smaller features but cost more, as covered in the CFL condition.
- **Vertical levels** are typically concentrated near the surface, where stratification, mixing, and forcing gradients are strongest, using the sigma-coordinate stretching.
- Resolution is not chosen in isolation, it depends directly on the bathymetric complexity and dynamical scales of the region selected in the previous step.



$1/4^\circ$ Horizontal Resolution



$1/20^\circ$ Horizontal Resolution

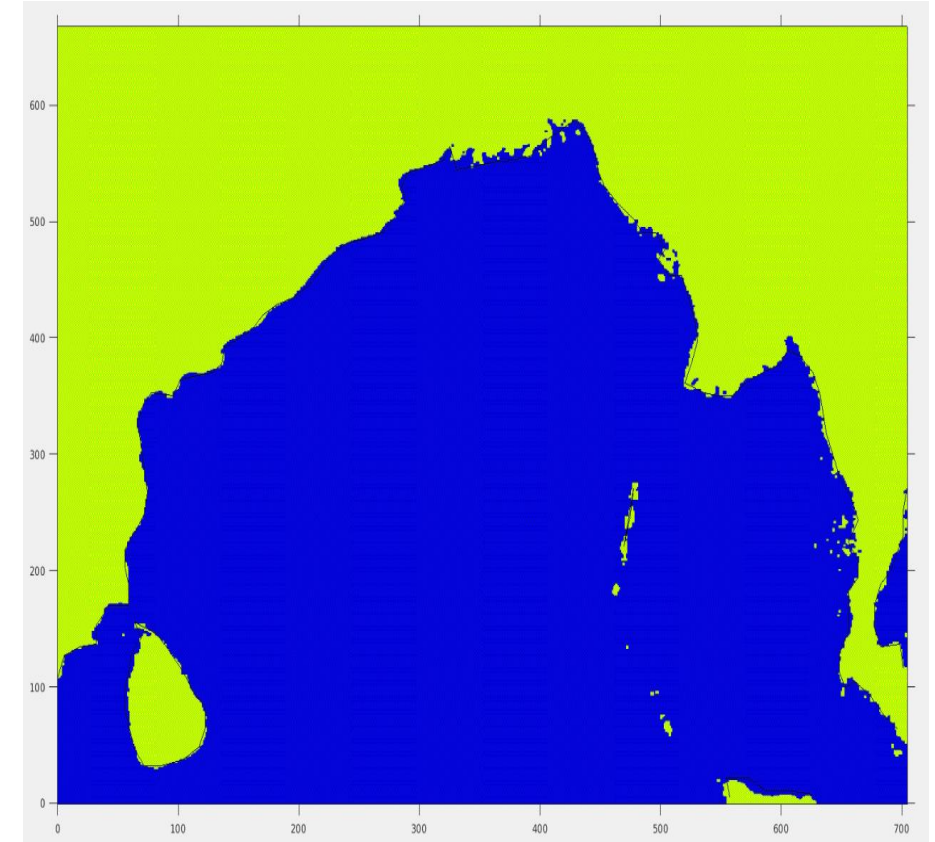


Sigma Vertical Levels

Computational Grid

Three components turn domain and resolution choices into an actual, usable model grid.

- **Horizontal grid generation:** Curvilinear or orthogonal coordinates are generated across the domain, following the C-grid staggering from Module 3, often with variable spacing to concentrate resolution near the coast.
- **Vertical sigma grid:** the terrain-following coordinate system is set up with stretching parameters controlling how levels concentrate near the surface versus the seabed.
- **Land-sea mask:** every grid cell is classified as water or land based on bathymetry, defining which cells the model actually computes.

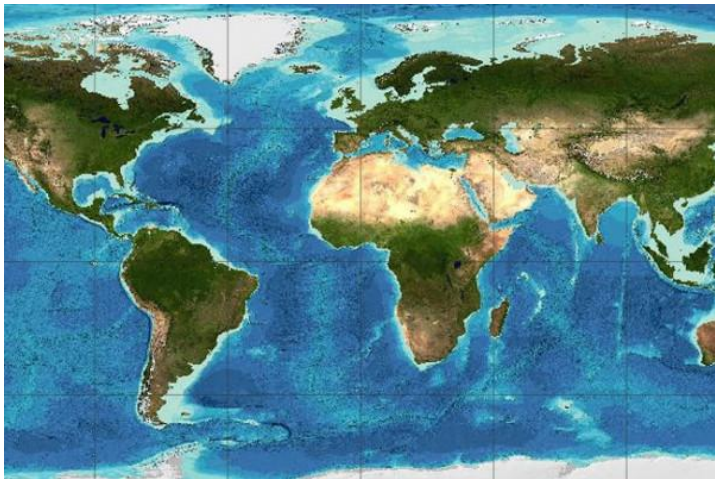


Land (Lime) –Sea (Dark Blue) Masking in CROCO Grid

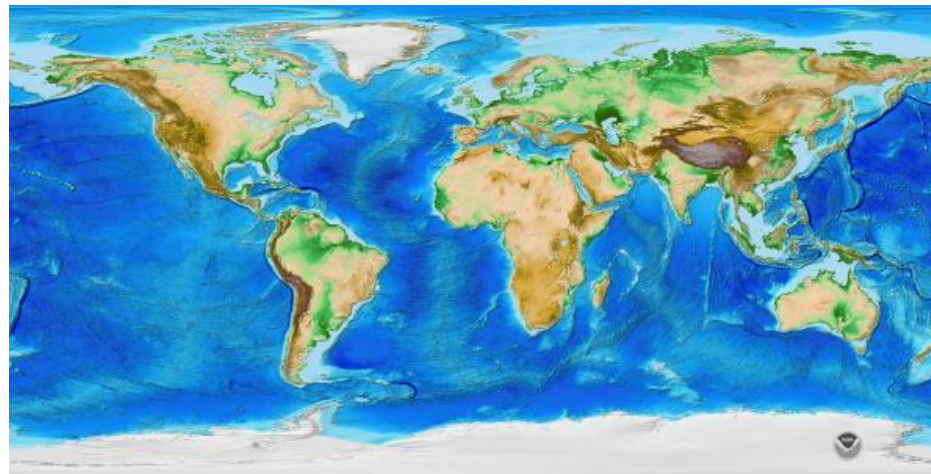
Bathymetry Preparation

Global depth datasets must be interpolated, smoothed, and checked before they're usable in a regional model.

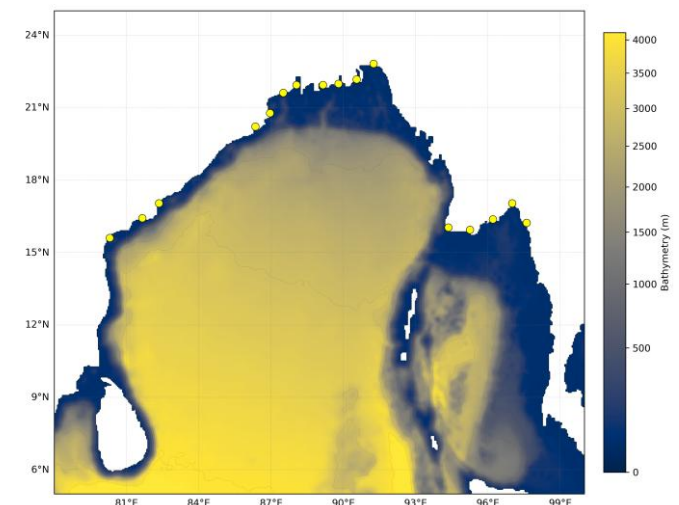
- Raw bathymetry comes from global datasets like **GEBCO** or **ETOPO**, then interpolated onto the model's specific curvilinear grid, exactly as covered when this topic first appeared.
- The interpolated bathymetry is smoothed to control steepness between adjacent grid points, since sigma coordinates amplify pressure gradient errors on steep slopes.
- The **r-factor** quantifies this steepness and is checked at every grid cell, values are typically kept below roughly 0.2 to 0.25 for numerical stability.



GEBCO bathymetric charts of the ocean
(<https://www.gebco.net/data-products/gridded-bathymetry-data>)



ETOPO Global Relief Model
(<https://www.ncei.noaa.gov/products/etopo-global-relief-model>)

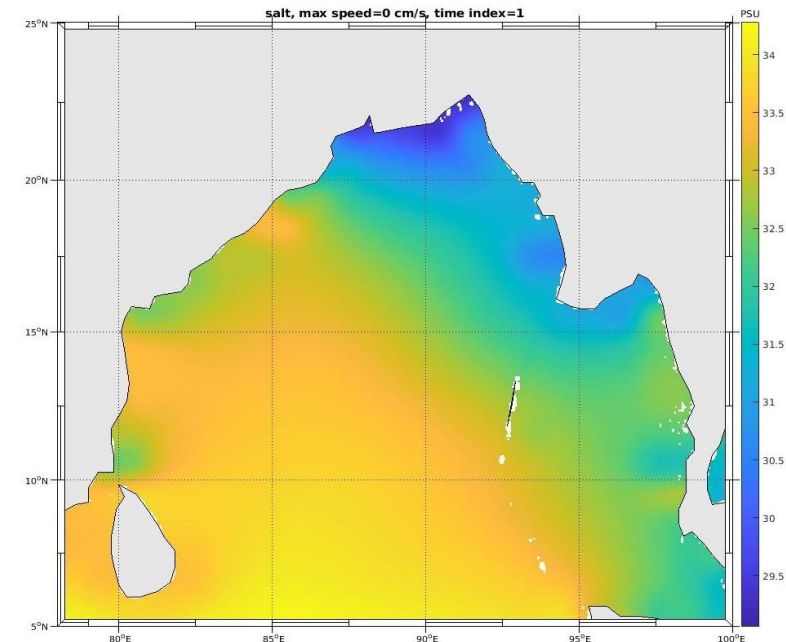
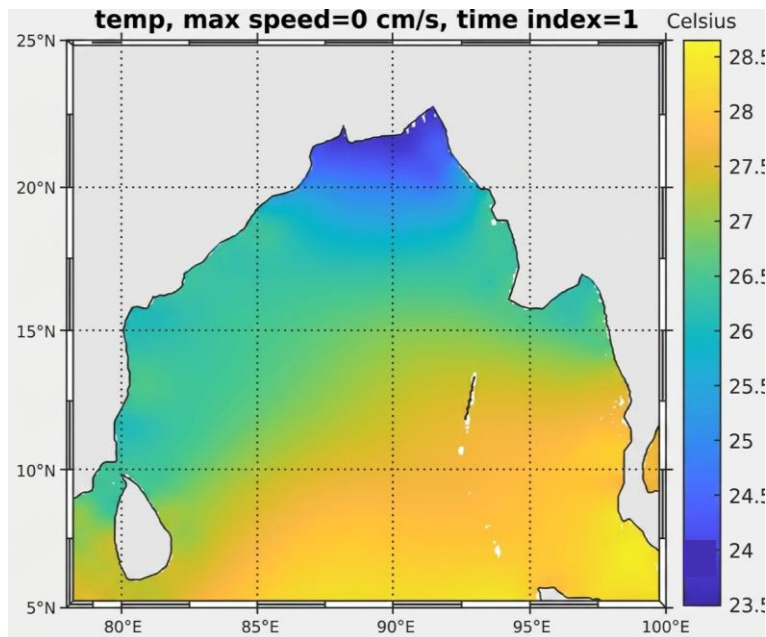


Smoothed Bathymetry from ETOPO2
Bathymetry Dataset for CROCO

Initial and Boundary Condition

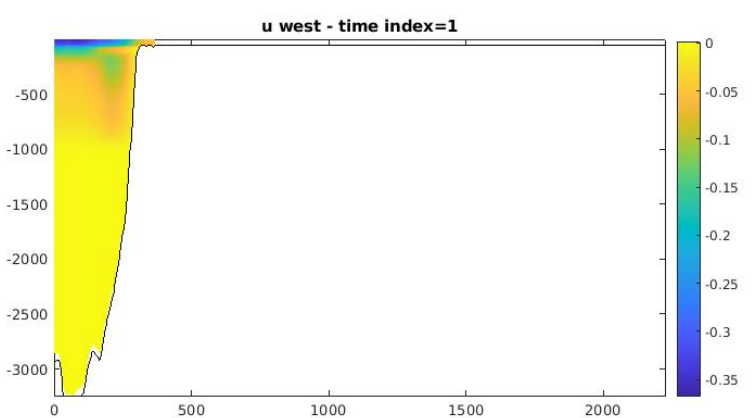
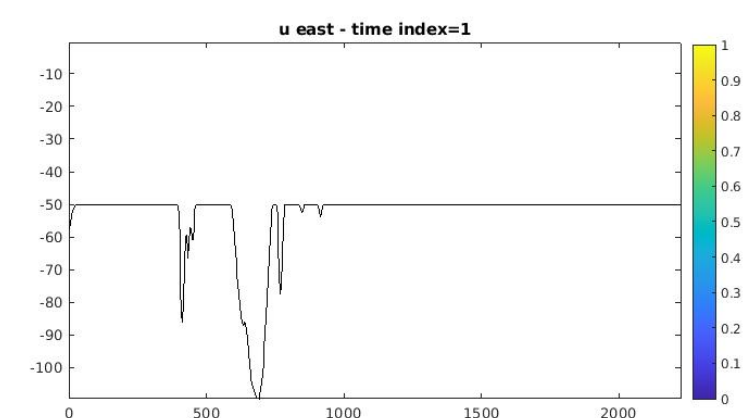
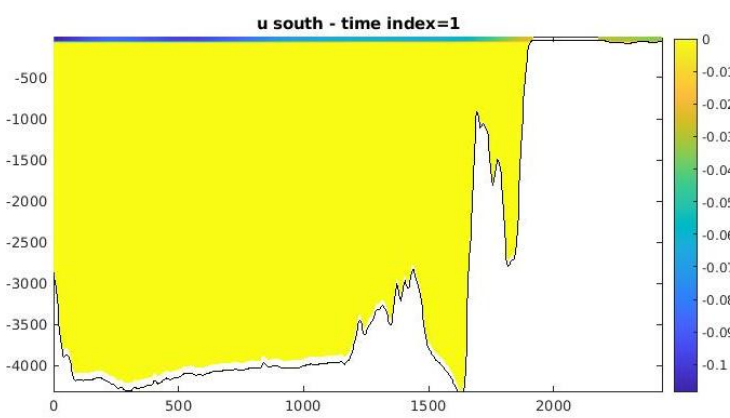
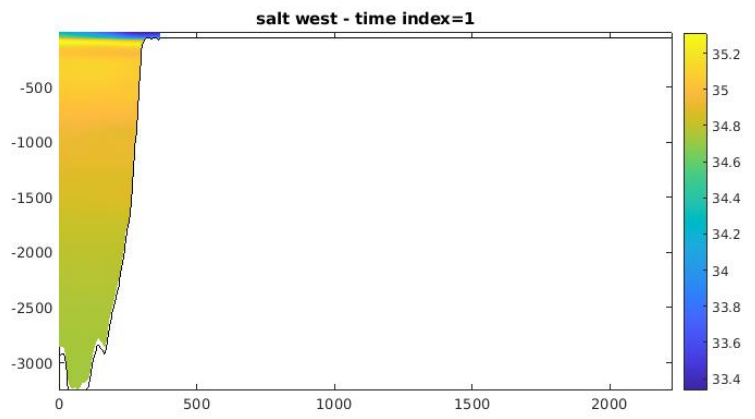
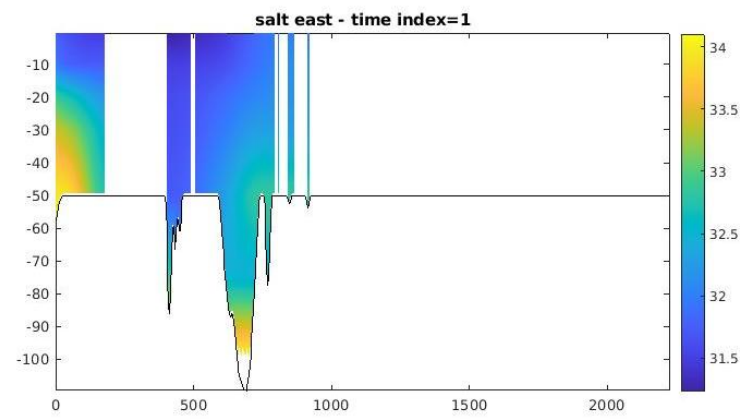
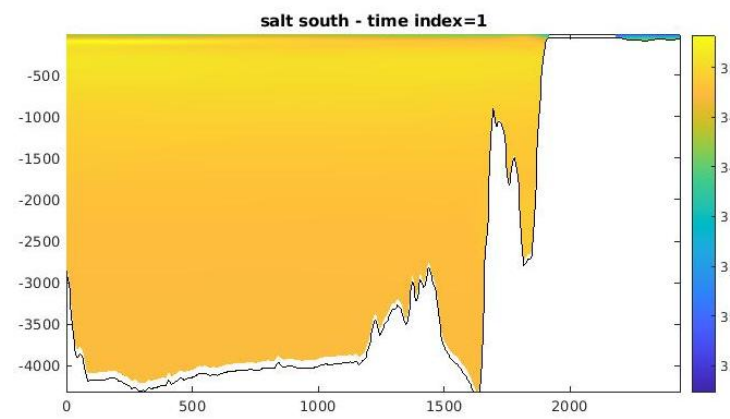
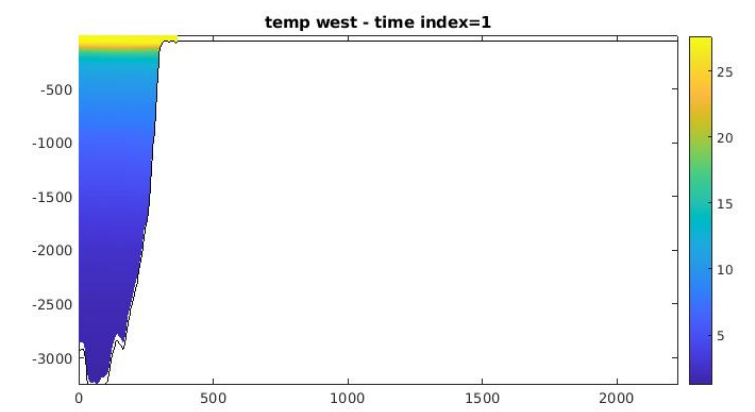
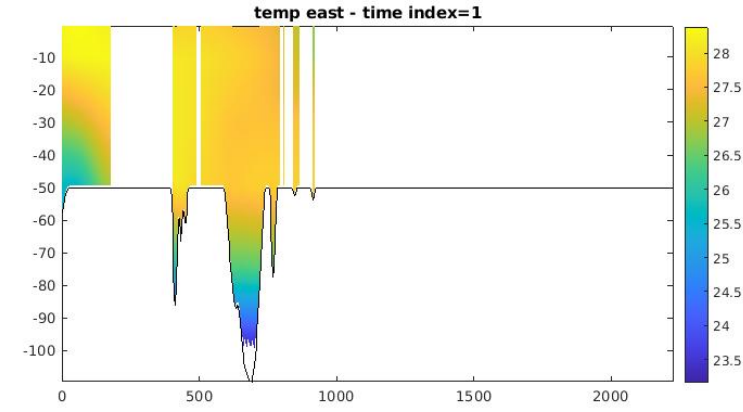
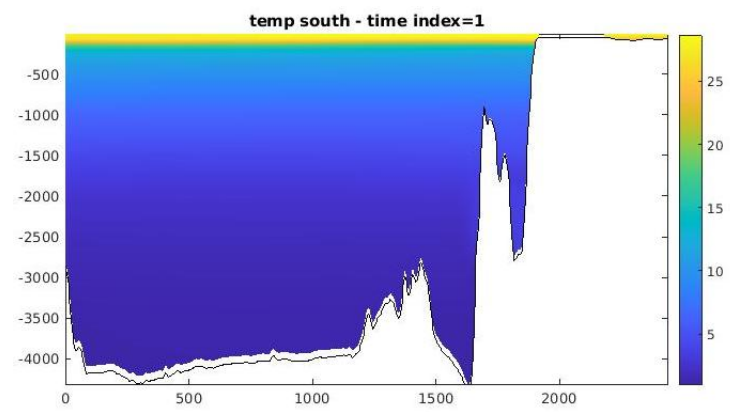
Temperature, salinity, currents, and sea surface height must be specified both at the start and at the boundaries.

- **Initial conditions** set the ocean's state everywhere in the domain at time zero, interpolated from a global reanalysis or climatology (e.g., CMEMS, HYCOM, FIOCOM, WOA) onto the model grid.
- **Boundary conditions** supply the same four fields, temperature, salinity, currents, and sea surface height, continuously at the open edges. This is the external data that the nudging technique relaxes the model toward.
- Both sources must be dynamically consistent, temperature-salinity-density relationships and current-SSH balance need to match physically, or the model will spend early simulation time adjusting away from unrealistic initial imbalances.



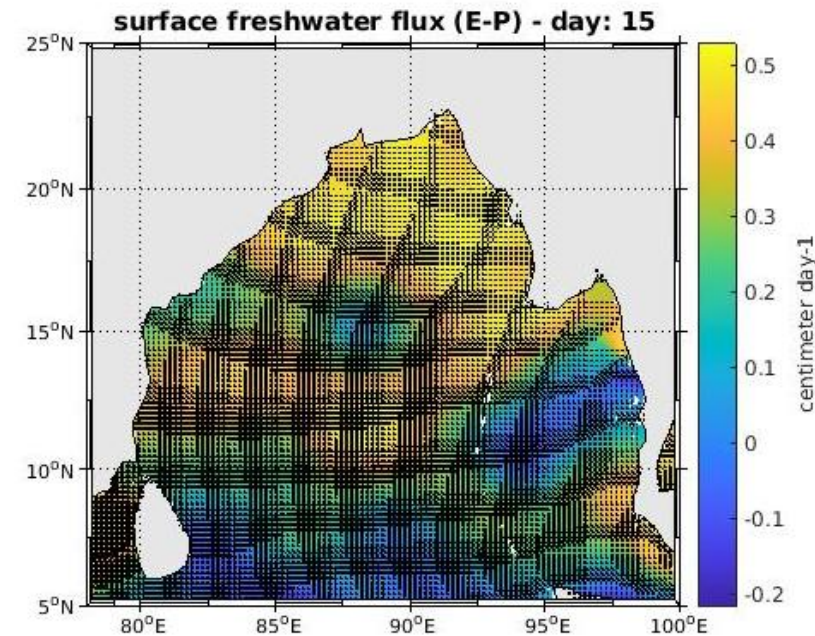
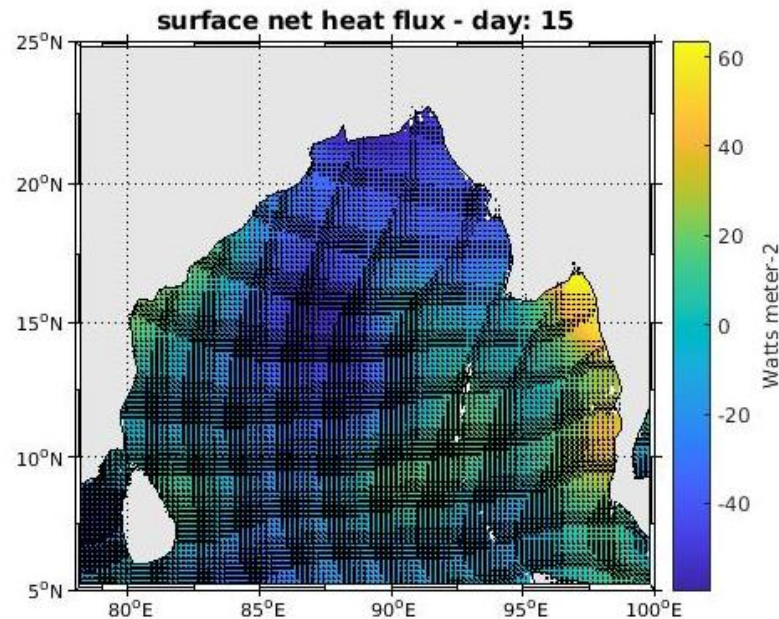
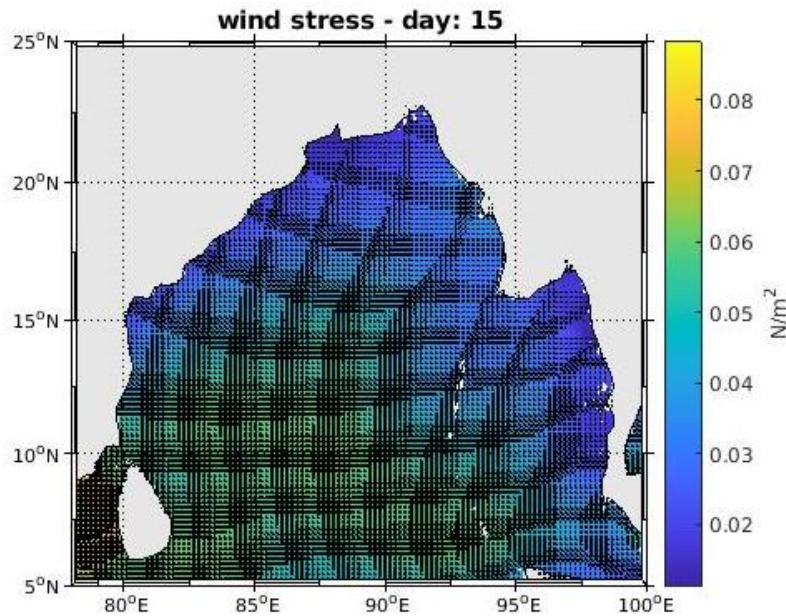
CROCO Initial Condition

CROCO Open Boundary Condition



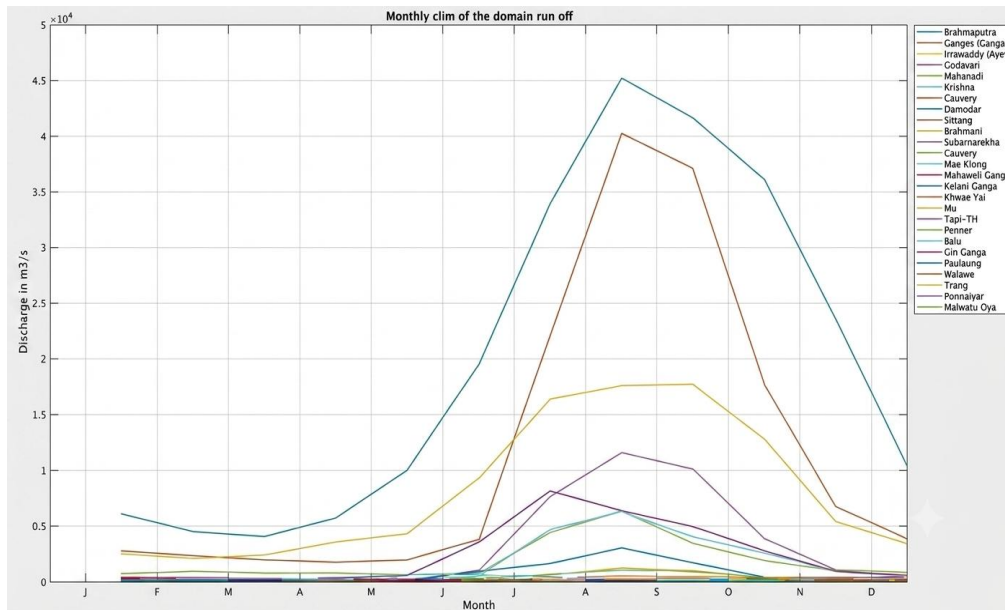
Atmospheric Forcing

- **Wind stress** and **heat flux** are computed from atmospheric reanalysis products (e.g., COADS05, ERA5, CFSv2), providing the momentum and thermal input covered earlier as bulk formulas.
- **Freshwater flux (P-E)** combines precipitation and evaporation fields into the net surface salinity forcing.
- **Surface pressure** is included for its effect on sea level (the inverse barometer effect) and, in coupled or high-resolution setups, for more accurate flux calculations.

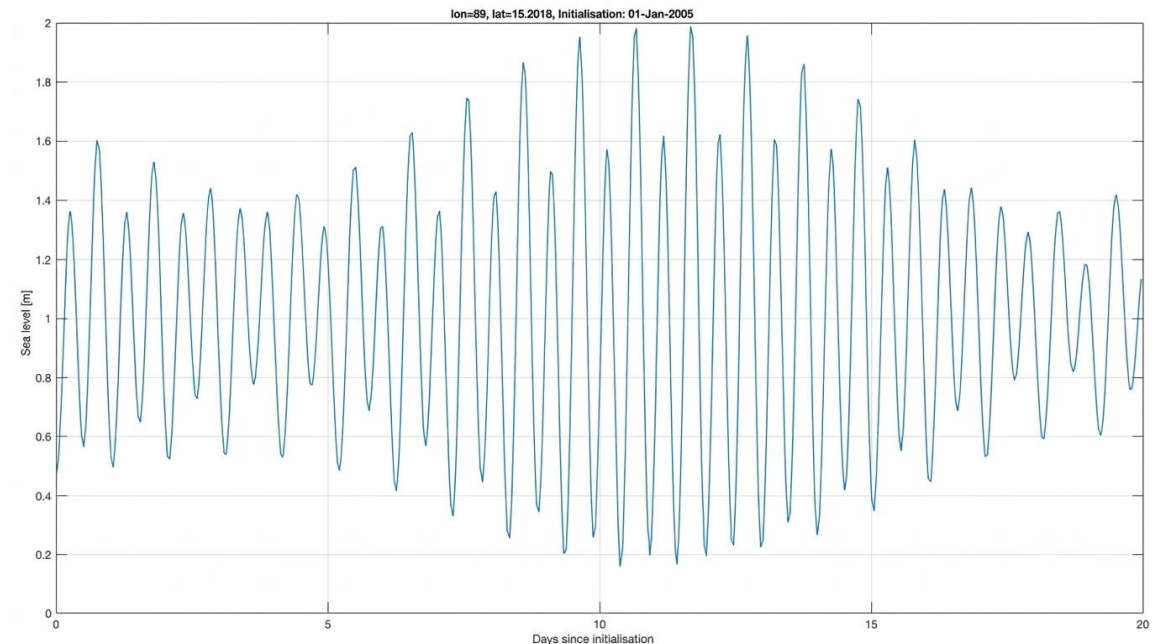


Additional Forcing

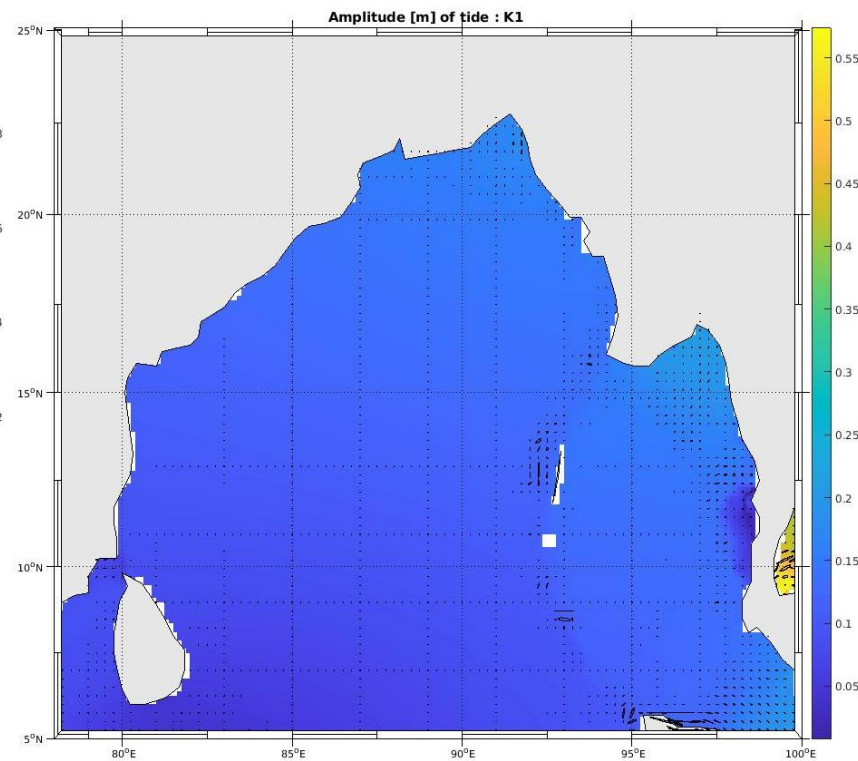
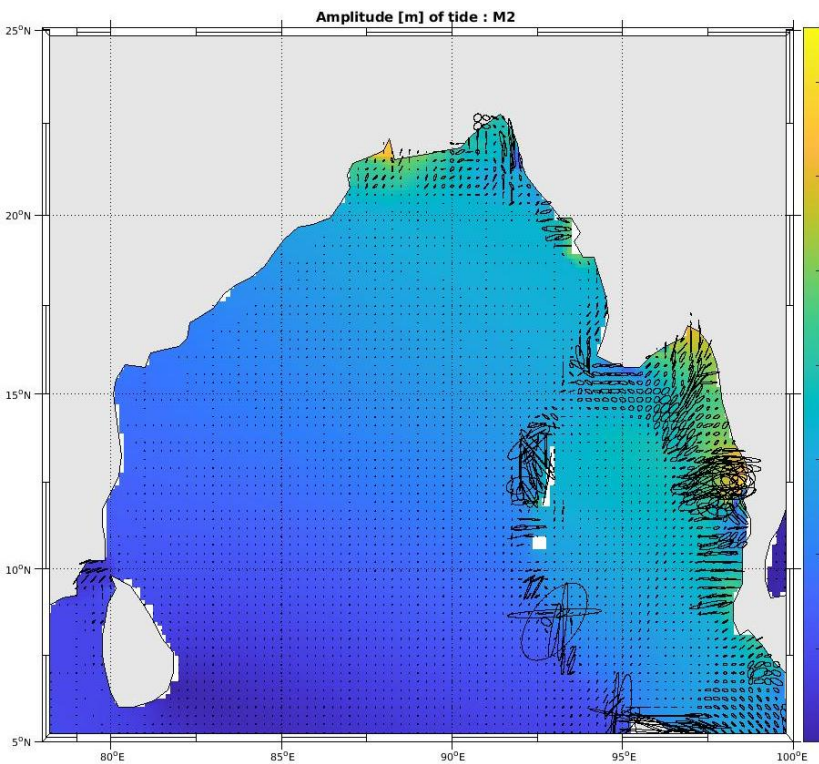
- **River input** adds freshwater discharge at specific grid points, using climatological or observed monthly discharge data from sources such as the Dai and Trenberth global river discharge dataset, the Global Runoff Data Centre (GRDC), or national hydrological agencies. This is critical for deltas and estuaries where salinity gradients drive local dynamics.
- **Tidal forcing** introduces tidal constituents (e.g., M2, S2, K1, O1) at the open boundaries, typically derived from a global tide model such as TPXO.
- Both are especially important in shallow, semi-enclosed regions like the Bay of Bengal or Gulf of Thailand, where tidal currents and river plumes dominate local water exchange.



Monthly Climatological River Discharge from
Dai and Trenberth for CROCO Domain



Tidal Amplitude from TPXO7 for CROCO Simulation



Spatial Distribution of Tidal Amplitude M2 and K1 in CROCO Domain

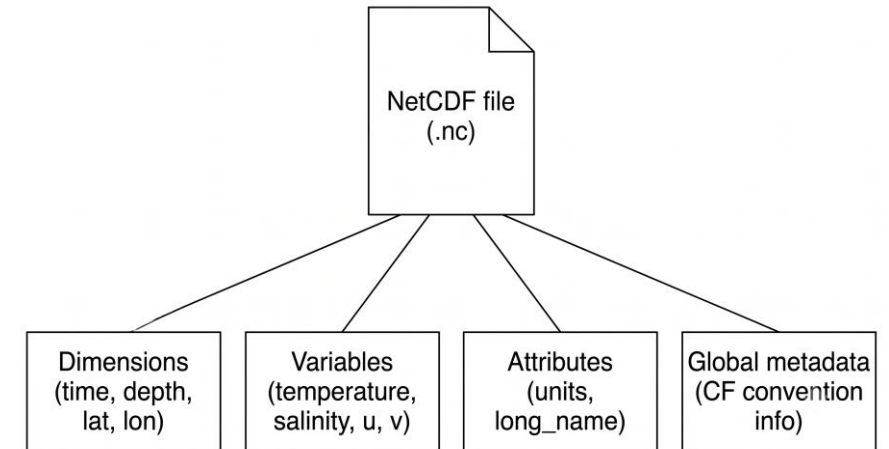
River Discharge Measuring Station



Input Data Formats

Every input discussed so far, grid, bathymetry, forcing, boundary data, ultimately becomes a standardized file the model can read.

- **NetCDF** (Network Common Data Form) is the standard file format for all CROCO inputs and outputs, storing multi-dimensional data (time, depth, latitude, longitude) efficiently in a single self-describing file.
- **CROCO input files** follow a specific structure: grid file, forcing file, boundary file, and initial condition file, each with required variable names the model expects exactly.
- **CF Conventions** (Climate and Forecast metadata standards) define naming, units, and attribute rules, ensuring files are interpretable across different models and tools, not just CROCO.



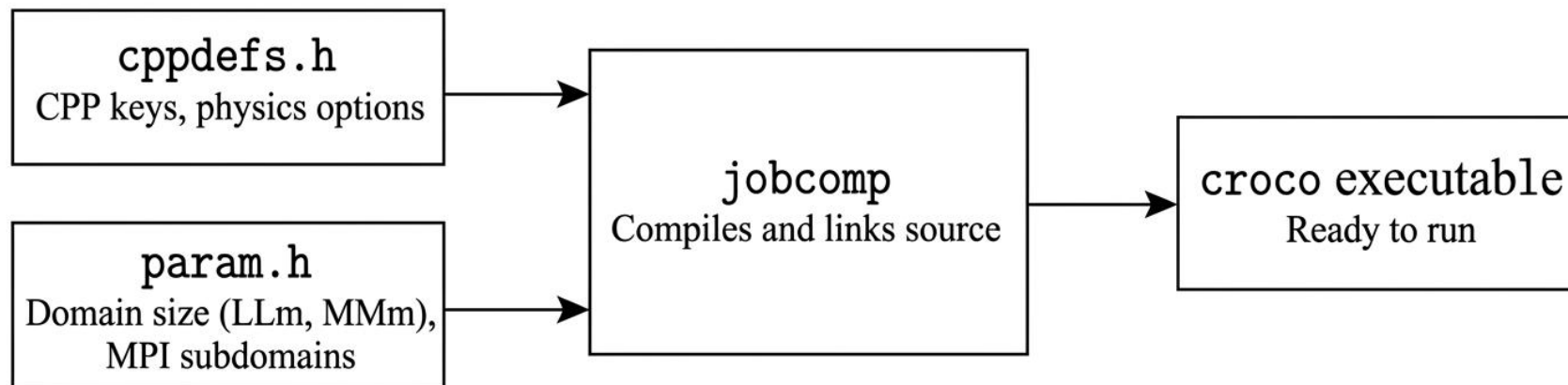
Module 06

Simulation, Post-processing & Validation

Compilation and Configuration Files

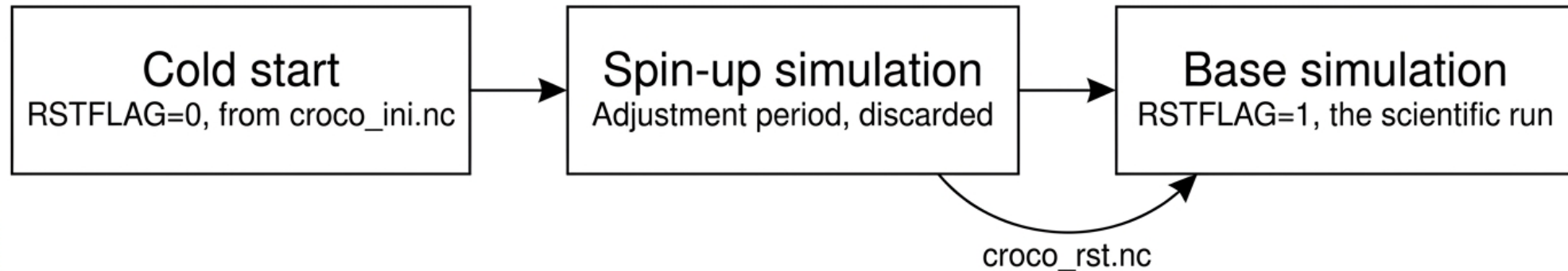
Before any simulation, CROCO's Fortran source, sized to a specific domain, must be compiled into one executable.

- **cppdefs.h** switches physical and numerical options on or off at compile time, non-hydrostatic mode, sediment dynamics, coupling, and more, each controlled by a CPP key.
- **param.h** hardcodes the domain size (LLm, MMm grid points, N vertical levels) and, if MPI is active, the number of subdomains (NP_XI, NP_ETA) the domain is split into, all fixed at compile time, not changeable at runtime and other essential functions.
- **jobcomp** ties it together: it copies the source, applies cppdefs.h and param.h settings, links NetCDF (and MPI if active), and compiles everything into a single executable, **croco**.



Cold Start, Restart & Simulation Phases

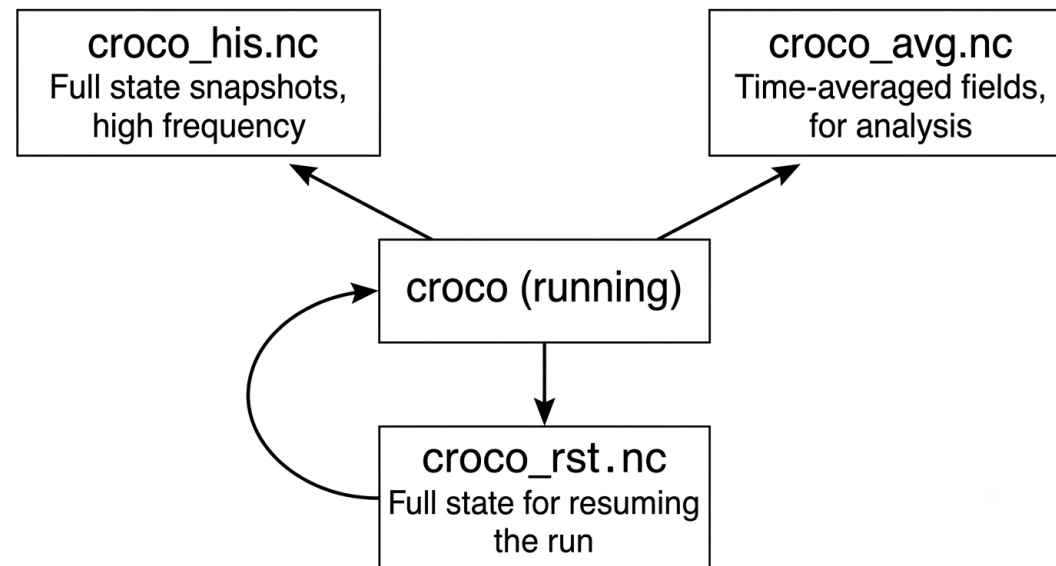
- **Cold start:** The model begins entirely from the initial condition file with no prior model history, this is how every simulation begins.
- **Spin-up:** An initial run, often forced with climatological conditions, allowed to run until the model reaches a stable, physically consistent state.
- **Restart:** subsequent runs continue from a saved restart file rather than the initial condition, letting long simulations be broken into manageable chunks, and the final spin-up state feeds directly into the **base simulation**, the actual scientific run.



CROCO Outputs

Every simulation produces three distinct types of NetCDF output, each serving a different purpose.

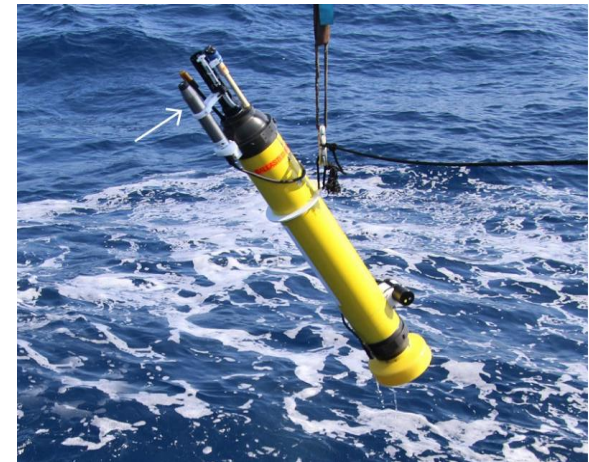
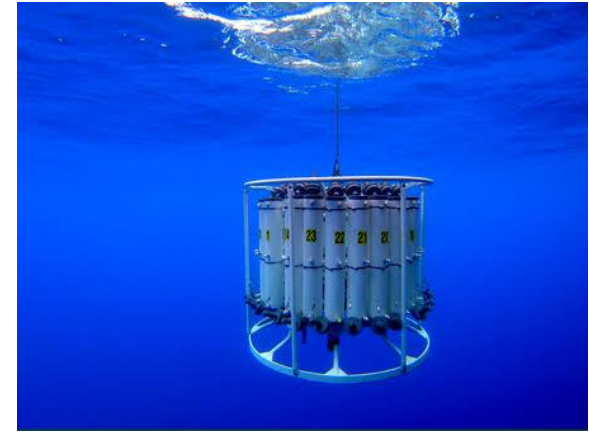
- **History files** (croco_his.nc) store the full model state (all variables, all grid points) at a chosen frequency, snapshots in time, useful for detailed process studies but large in size.
- **Average files** (croco_avg.nc) store time-averaged fields over a chosen window, smoothing out fast oscillations (tides, inertial waves) to reveal the mean circulation, typically the primary file used for scientific analysis.
- **Restart files** (croco_rst.nc) store the complete model state needed to resume the simulation exactly where it left off, this is what chains months together, as seen in run_croco.bash.



Model Validation

A simulation is only useful if it's been tested against independent, real-world observations.

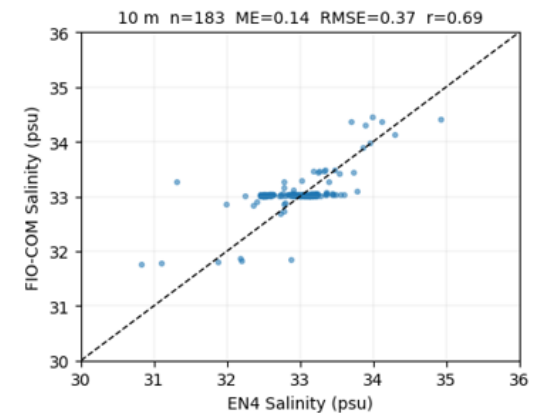
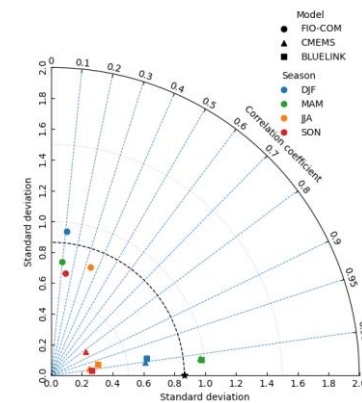
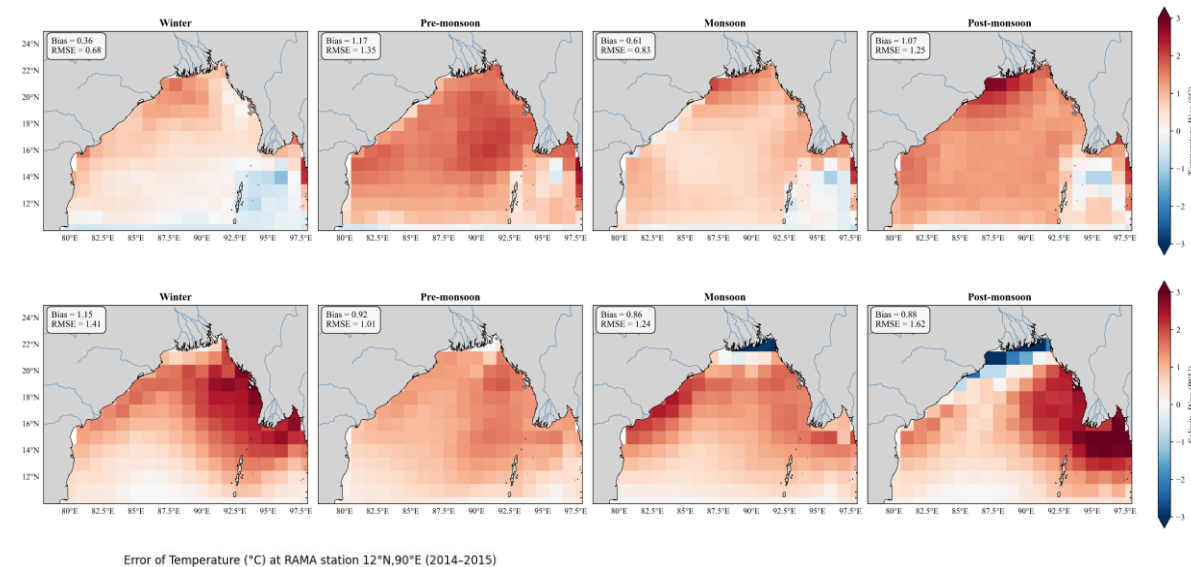
- **Satellite observations:** Broad spatial coverage of SST, SSS, and sea surface height.
- **Argo Floats:** Autonomous vertical profiles of temperature and salinity.
- **CTD Measurements:** High-precision shipboard casts at specific transects.
- **Tide Gauges:** Coastal sea level and tidal validation
- **Oceanic Buoys:** Fixed-point time series of currents, temperature, and other variables



Statistical Evaluation

Visual comparison isn't enough, model performance needs to be quantified.

- **Bias** shows the direction of error, whether the model runs systematically warmer, saltier, or higher than observed.
- **MAE (Mean Absolute Error)** captures error magnitude without letting positive and negative errors cancel out, a clearer picture than bias alone.
- **RMSE (Root Mean Square Error)** also measures error magnitude, but penalizes large deviations more heavily than small ones.
- **Correlation (r)** measures how well the model captures the pattern and timing of variability, independent of absolute magnitude.
- **Taylor diagram** brings correlation, RMSE, and standard deviation together into a single plot, a compact way to compare multiple models or variables at once.



Best Practices

Beyond getting a simulation to run, three habits determine whether the results can be trusted and reused.

- **Convergence:** Results shouldn't change meaningfully with small changes in resolution or time step, if they do, the simulation likely hasn't reached a stable, reliable solution.
- **Reproducibility:** Every configuration file, CPP key, parameter, and input dataset version should be tracked, so the exact same run can be repeated by you or anyone else, months or years later.
- **Documentation:** Recording what was changed, why, and what the effect was turns a one-off simulation into a foundation other studies (and other researchers) can build on.

Reference

1. Fairall, C. W., Bradley, E. F., Hare, J. E., Grachev, A. A., & Edson, J. B. (2003). Bulk parameterization of air-sea fluxes: Updates and verification for the COARE algorithm. *Journal of Climate*, 16(4), 571–591.
2. Apriansyah, A., Atmadipoera, A. S., Jaya, I., Nugroho, D., & Akhir, M. F. (2023). An Evaluation of a 1/18° Resolution Regional Ocean Circulation Model of CROCO in the Southern Sunda Shelf. *ILMU KELAUTAN: Indonesian Journal of Marine Sciences*, 28(1), 12–26. <https://doi.org/10.14710/ik.ijms.28.1.12-26>
[Untan University](#)
3. Jullien, S., Marchesiello, P., Menkes, C. E., Lefèvre, J., Jourdain, N. C., Samson, G., & Lengaigne, M. (2014). Ocean feedback to tropical cyclones: climatology and processes. *Climate Dynamics*, 43, 2831–2854. <https://doi.org/10.1007/s00382-014-2096-6>
4. Large, W. G., & Pond, S. (1981). Open ocean momentum flux measurements in moderate to strong winds. *Journal of Physical Oceanography*, 11(3), 324–336.
5. Pianezze, J., Barthe, C., Bielli, S., Tulet, P., Jullien, S., Cambon, G., Bousquet, O., Claeys, M., & Cordier, E. (2018). A new coupled ocean-waves-atmosphere model designed for tropical storm studies: Example of tropical cyclone Bejisa (2013–2014) in the South-West Indian Ocean. *Journal of Advances in Modeling Earth Systems*, 10(3), 801–825. <https://doi.org/10.1002/2017MS001177>
6. Pottapinjara, V., & Joseph, S. (2022). Evaluation of mixing schemes in the HYbrid Coordinate Ocean Model (HYCOM) in the tropical Indian Ocean. *Ocean Dynamics*, 72, 341–359. <https://doi.org/10.1007/s10236-022-01510-2>
7. Silva, P. A., Castro, C. G., et al. Implementation and validation of a 3D regional ocean model CROCO for sediment transport purposes in NW Iberian continental shelf. *Ocean Dynamics* (2026). <https://doi.org/10.1007/s10236-026-01792-w> [ResearchGate](#)
8. Wu, Y., Xia, C., Qiao, F., Guo, J., Qu, D., & Fang, Y. (2026). Seasonal circulation and water exchange with the Southern South China Sea of the Gulf of Thailand. *Continental Shelf Research*, 301, 105736. <https://doi.org/10.1016/j.csr.2026.105736>



Any Question?



**Thank
You...**