# CROCO step
# for 3D fast-mode (NBQ)

*step3d_fast_zetaw Fortran routine*

June 18, 2018

## Contents

This document is related to the 2018 ZETAW version of CROCO-NBQ. It gives a step-by-step description of step3d_fast Fortran routine.

**Notations:**

- internal-mode time-step index is n, fast-mode time-step index is m,
- superscripts are written in brackets when the corresponding Fortran variable has no time dimension.

# 1 Initializations

## 1.1 Initialization of NBQ variables (first time-step)

In **initial_nbq**: initialization of $rho\_nbq^{(1)}$, $qdmu\_nbq^{(1)}$ & $qdmw\_nbq^{(1)}$:

- $rho\_nbq^{(1)} = 0$
- $qdmu\_nbq^{(1)} = u^1$
- $qdmw\_nbq^{(1)} = w^1$

**Here**, initialization of backup rho arrays:

- $rho\_bak^{(1)} = rho^{(1)}$
- $rho\_grd^{(1)} = (1.5rho^{(1)} - 0.5rho\_bak^{(1)}/rho0)$ extrapolation in time

## 1.2 Extrapolate D, $\bar{u}$ and $\bar{v}$ to $m + \frac{1}{2}$

Mass of water column and depth-averaged velocities are extrapolated to $m + \frac{1}{2}$ to compute rhs of fast-mode equations:

$$\begin{aligned}
Drhs^{(m+\frac{1}{2})} &= (\frac{3}{2} + \beta)(H + zeta^m)rhobar_nbq^m \\
&- (\frac{1}{2} + 2\beta)(H + zeta^{m-1})rhobar_nbq^{m-1} \\
&+ \beta(H + zeta^{m-2})rhobar_nbq^{m-2} \\
urhs^{(m+\frac{1}{2})} &= (\frac{3}{2} + \beta)ubar^m - (\frac{1}{2} + 2\beta)ubar^{m-1} + \beta ubar^{m-2}
\end{aligned}$$

## 1.3 Initialize grid for fast-mode (first time-step)

call **grid_nbq**:

- $Hzw_{i,j,k}^{(1)} = z\_r_{i,j,k+1}^{(1)} - z\_r_{i,j,k}^{(1)}$
- Upper & lower BC: $Hzw_{i,j,0}^{(1)} = z\_r_{i,j,1}^{(1)} - z\_w_{i,j,0}^{(1)}$ & $Hzw_{i,j,N}^{(1)} = z\_w_{i,j,N}^{(1)} - z\_r_{i,j,N}^{(1)}$
- Caution: Hzw is stored in Hzw_half_nbq
- Compute: Hzr_half_nbq_inv($Hzr$), Hzu_half_qdmu_nbq($Hzr_u$), Hzw_half_nbq_inv($Hz_w$).

## 2 External fast-mode processing

---

### 2.1 Compute $\zeta^{m+1}$ $(1^{st} guess)$

$\zeta^{m+1}$ is computed here based on surface characteristic relation before grid is updated. This is ZETAW version of CROCO-NBQ. In the older ZETA2D version, $\zeta^{m+1}$ is obtained using depth-averaged conservation of mass. Whatever the method, both relations must be satisfied by surface anomaly but in NH, this anomaly is dynamically coupled with the surface layer and should thus be computed based on the surface vertical velocity (surface characteristic relation & ZETAW version).

- Prepare computations:
  - U surface-velocity:

$$umean\_nbq_{i,j}^{(m)} = \frac{2\ qdmu\_nbq_{i,j,N}^{(m)}}{Hz_{i,j,N}^{(m)} + Hz_{i-1,j,N}^{(m)}}$$

  - W surface-velocity:

$$wmean\_nbq_{i,j}^{(m)} = \frac{2\ qdmw\_nbq_{i,j,N}^{(m)}}{Hzw_{i,j,N}^{(m)}}$$

  - Zeta for RHS:

$$zab3^{(m+\frac{1}{2})} = (\frac{3}{2}+\beta)zeta^{m} - (\frac{1}{2}+2\beta)zeta^{m-1} + \beta zeta^{m-2}$$

- Update zeta
  - $zeta^{m+1}$ is given by surface kinematic relation:

$$\begin{aligned} zeta^{m+1} = zeta^{m} + \Delta t_{fast}\ \Big[\ & wmean\_nbq_{i,j}^{(m)} \\ & - \frac{1}{2}\left( umean\_nbq_{i,j}^{(m)} \frac{zab3_{i,j}^{(m+\frac{1}{2})} - zab3_{i-1,j}^{(m+\frac{1}{2})}}{\Delta x} \right. \\ & \left. + umean\_nbq_{i+1,j}^{(m)} \frac{zab3_{i+1,j}^{(m+\frac{1}{2})} - zab3_{i,j}^{(m+\frac{1}{2})}}{\Delta x} \right) \Big] \end{aligned}$$

    Time scheme to be improved ?
  - $zeta^{m+1} = zeta^{m+1} + zonal\ nudging$
  - call **zetabc**
  - call **wetdry** compute WET/DRY masks

### 2.2 Update grid

As soon as $\zeta^{m+1}$ is known, grid can be updated at fast-mode step m+1 ("PRECISE" implementation of the algorithm). In "PERF" implementation, grid can be updated at a lower frequency and only during re-evaluation of zeta(m+1).

Caution: $Hz\_bak$ must be assigned only once every internal step in set_depth. Because in step3d_fast, set_depth is called more than once, the logic in step3d_fast must be consistent with that of set_depth code.

- call **set_depth** (update vertical grid variables):
  $z\_w^{(m+1)}$, $z\_r^{(m+1)}$, $Hz\_bak^{(m+1)}$, $Hz\_bak2^{(m+1)}$, $Hz^{(m+1)}$.

- call **grid_nbq**:

  - $Hzw_{i,j,k}^{(m+1)} = z\_r_{i,j,k+1}^{(m+1)} - z\_r_{i,j,k}^{(m+1)}$
  - Upper & lower BC:
    $Hzw_{i,j,0}^{(m+1)} = z\_r_{i,j,1}^{(m+1)} - zw_{i,j,0}^{(m+1)}$ & $Hzw_{i,j,N}^{(m+1)} = zw_{i,j,N}^{(m+1)} - z\_r_{i,j,N}^{(m+1)}$
  - Caution: variable for $Hzw^{(m+1)}$ is Hzw_half_nbq! To be changed.
  - Compute: Hzr_half_nbq_inv(Hzr), Hzu_half_qdmu_nbq(Hzr), Hzw_half_nbq_inv($Hz_w$)

### 2.3 Prepare evaluation of surface pressure-force

- $zwrk^{(m)} = zeta^m$
- $Dnew^{(m)} = (H + zeta^m)rhobar\_nbq^m$
- Compute $rzeta^{(m)}(zwrk^{(m)}),\ rzeta2^{(m)}(zwrk^{(m)}),\ rzetaSA^{(m)}(zwrk^{(m)})$

### 2.4 Compute $\overline{rhs}$ for fast-mode momentum equations

Depth-averaged (2D) forcing of fast-mode momentum equations is updated at every fast-mode time-step as it changes with surface anomaly. This depth-averaged component of forcing is latter added to the sheared component of forcing (rufrc).

$$
\begin{aligned}
rubar^{(m)} = &\ pressure\ force && \leftarrow (Zwrk^{(m)},\ rzeta^{(m)},\ rzeta2^{(m)},\ rzetaSA^{(m)}) \\
&+ horizontal\ advection && \leftarrow (DUon,\ urhs) \\
&+ Coriolis\ pseudo-force && \leftarrow (Drhs,\ urhs, ...) \\
&+ Advection\ metric\ coef && \\
&+ viscous\ stress && \leftarrow (Drhs,\ ubar) \\
&+ linear\ /\ quadratic\ bottom\ stress && \leftarrow (ubar) \\
&+ 2D-vortex\ force\ combined\ with\ advection && \leftarrow (Drhs,\ urhs)
\end{aligned}
$$

### 2.5 Int-mode to fast-mode ($1^{st}$ fast-mode time-step)

- Save rufrc: $cff = rufrc^{(n+\frac{1}{2})} - rubar^{(n)}$
- Update $rufrc_{i,j}^{(n+\frac{1}{2})}$:

$$
\begin{aligned}
rufrc_{i,j}^{(n+\frac{1}{2})} &= .281105\ cff \\
&+ \left(-\frac{1}{2} - 2 * .281105\right)\ rufrc\_bak^{(n-1)} \\
&+ \left(\frac{3}{2} - .281105\right)\ rufrc\_bak^{(n-2)}
\end{aligned}
$$

- Save rufrc: $rufrc\_bak^{(n)} = cff$
- Initialize 2D coupling arrays:
  - For time summation: ru_ext_nbq_sum=0
  - Working variable: ru_ext_nbq_old=0

### 2.6 Total forcing to fast-mode (ru_int_nbq)

- Update 2D forcing:

$$ru\_ext\_nbq^{(m)} = \frac{rufrc^{(n+\frac{1}{2})} + rubar^{(m)}}{\Delta x \Delta y \left( Drhs_i^{(m+\frac{1}{2})} + Drhs_{i-1}^{(m+\frac{1}{2})} \right)}$$

  Caution: multiplication by 2 missing to speedup computations.
  Consistency of denominator ?

- $ru\_ext\_nbq\_old = ru\_ext\_nbq^{(m)} - ru\_ext\_nbq\_old$
- $ru\_ext\_nbq\_sum = ru\_ext\_nbq\_sum + ru\_ext\_nbq^{(m)}$
- Update Int-mode 3D forcing:

$$\begin{aligned} ru\_int\_nbq^{(m)} &= ru\_int\_nbq^{(m-1)} \\ &+ ru\_ext\_nbq\_old * (Hz_{i,j}^{(m)} + Hz_{i-1,j}^{(m)}) \end{aligned}$$

### 2.7 Initializations & backups before fast-mode 3D equations

- $1^{st}$ fast-mode time-step:
  - $ru\_nbq\_avg2^{(*)} = qdmu\_nbq^{(m)}$
  - $rw\_nbq\_avg2^{(*)} = qdmw\_nbq^{(m)}$
- Backups:
  - $ru\_ext\_nbq\_old = ru\_ext\_nbq^{(m)}$
- Time-dependent grid variables used to compute time-derivatives: $zw\_nbq^* = z\_w^{(1)}$
- $work_{i,j} = \Delta x \ \Delta y$ (temporarily)
- Initializations:
  - $rubar\_nbq^{(m+1)} = 0$
  - $DU\_nbq^{(m+1)} = 0$
- Working variable used to store $qdmw\_nbq^{(m)}$:
  $rw\_nbq^{(*)} = qdmw\_nbq^{(m)}$
- Recover Hz at first fast step (if final Hz correction needed)
- Store boundary values of NBQ variables at previous time-step for use in radiation boundary conditions

## 3 Fast-mode 3D-equations

Momentum and mass-conservation equations can now be updated for fast-mode. These equations are 3D. W-momentum equation can be integrated based on explicit or implicit methods.

- **[Explicit scheme]** w-momentum is updated right after (and the same way as) u- and v-momentum.
- **[Implicit scheme]** horizontal component of divergence is first precomputed (as required by fast-mode mass conservation equation) before tridiagonal *Gauss Elimination* is carried out for $qdmw\_nbq^{(m)}$.

Compressible pressure-force and second viscosity are calculated using thetadiv_nbq. Caution: this variable contains theta in the first part of the algorithm and divergence of momentum and in the remaining.

A forward-backward scheme is implemented:

- **[Explicit scheme]** Forward: zeta, qdmu_nbq, qdmw_nbq. Backward: rho_nbq.
- **[Implicit scheme]** Forward: zeta, qdmu_nbq. Backward: qdmw_nbq, rho_nbq.

### 3.1 Compute Pressure-Viscosity component thetadiv_nbq

$$\begin{aligned} thetadiv\_nbq^{(m)} = [ & -visc2(thetadiv\_nbq^{(m)} + thetadiv3\_nbq^{(m)}) \\ & + soundspeed2 * rho\_nbq^{(m)} \\ ] & \; Hzr\_half\_nbq\_inv^{(m)} \end{aligned}$$

### 3.2 Fast-mode u-momentum

- Compute RHS at m:

$$\begin{aligned} dums^{(m)} = (compressible\ pressure\ force\ +\ second\ viscosity) \leftarrow\ & grid^{(m+1)}, \\ & qdmu\_nbq^{(m)},\ ru\_int\_nbq^{(m)}, \\ & thetadiv\_nbq^{(m)} \end{aligned}$$

- Update $qdmu\_nbq_{i,j,k}^{(m+1)}$:

$$\begin{aligned} qdmu\_nbq_{i,j,k}^{(m+1)} = & qdmu\_nbq_{i,j,k}^{(m)} \\ & + \Delta t_{fast}(dums^{(m)} + ru\_int\_nbq^{(m)}) \end{aligned}$$

- $DU\_nbq_{i,j}^{(m+1)} = DU\_nbq_{i,j}^{(m+1)} + \sum_{k=1}^{N} qdmu\_nbq_{i,j,k}^{(m+1)}$
- $ru\_nbq_{i,j,k}^{(m)} = dums^{(m)}/work_{i,j}$
- $rubar\_nbq_{i,j}^{(m+1)} = rubar\_nbq_{i,j}^{(m+1)} + \sum_{k=1}^{N} ru\_nbq\_ext_{i,j,k}^{(m)}$
- Add Fast-mode nudging
- Lateral boundary conditions: call **u2dbc** & call **unbq_bc**

### 3.3 Fast-mode w-momentum (explicit scheme)

- Compute RHS at m:

$$\begin{aligned} dums^{(m)} =& (compressible\ pressure\ force\ +\ second\ viscosity \leftarrow grid^{(m+1)}, \\ & \qquad qdmw\_nbq^{(m)},\ rw\_int\_nbq^{(m)}, \\ & \qquad thetadiv\_nbq^{(m)} \\ & +gravity) \leftarrow rho\_nbq^{(m)} \end{aligned}$$

- Update $qdmw\_nbq_{i,j,k}^{(m+1)}$ :

$$\begin{aligned} qdmw\_nbq_{i,j,k}^{(m+1)} =& qdmw\_nbq_{i,j,k}^{(m)} \\ & + \Delta t_{fast}(dums^{(m+1)} + rw\_int\_nbq^{(m)}) \end{aligned}$$

- Add fast-mode nudging
- Lateral boundary conditions: call **wnbq_bc**

### 3.4 Fast-mode conservation of mass and w-momentum (implicit scheme)

From now on, thetadiv_nbq is used for divergence, it is not theta anymore.

- $thetadiv\_nbq^{(m+1)} = div_s^{(h)}(qdmu\_nbq^{(m+1)})$
- Backup: $zw\_nbq^m = z\_w^{(m)}$
- $thetadiv2\_nbq^{(m+1)} = reduced\ slope\ derivatives + time\ derivatives$
- $thetadiv3\_nbq^{(m+1)} = slope\ derivatives$
- if implicit scheme, compute here $qdmw\_nbq^{(m+1)}$ by Gaussian elimination.
- $thetadiv\_nbq^{(m+1)} = thetadiv\_nbq^{(m+1)} + div_s^{(v)}(qdmw\_nbq^{(m+1)})$
- Update $rho\_nbq_{i,j,k}^{(m+1)}$:

$$\begin{aligned} rho\_nbq_{i,j,k}^{(m+1)} = rho\_nbq_{i,j,k}^{(m)} - \Delta t_{fast}(&\ thetadiv\_nbq^{(m+1)} \\ & + thetadiv2\_nbq^{(m+1)}) \end{aligned}$$

- Add fast-mode nudging.

# 4 Fast-mode post-processing

## 4.1 Update total mass of water column

$$Dnew^{(m+1)} = (H + zeta^{m+1})rhobar\_nbq^{m+1}$$

## 4.2 Backup slow-mode density (rho_bak)

$$rho\_bak = rho^{(n)}$$

## 4.3 Update rho for mode coupling

- Update $rhobar\_nbq^{m+1}$ for 2D computations (fast-mode time-step):

$$rhobar\_nbq^{m+1} = \frac{\sum_{k=1}^{N} \left(1 + rho^{(n)}/\rho_0 + rho\_nbq^{(m+1)}\right) Hzr^{(m+1)}}{z\_w_{i,j,N}^{(m+1)} - z\_w_{i,j,0}^{(m+1)}}$$

- Update $rho\_nbq\_ext^{m+1}$ for Int-mode computations:

$$rho\_nbq\_ext^{m+1} = 1 + \frac{rho^{(n)}}{\rho_0} + rho\_nbq^{(m+1)}$$

## 4.4 Last fast-mode time-step

In the present algorithm, $ru\_nbq\_ext^{(m+1)}$ and $rw\_nbq\_ext^{(m+1)}$ are not updated the same way. This is inherited from an old action to speed-up code... To be homogenized?

- Update $rw\_nbq^{(m+1)}$:

$$\begin{aligned} rw\_nbq^{(m+1)} = [&(qdmw\_nbq^{(m+1)} \\ &- rw\_nbq^{(*)})/\Delta t_{fast} \\ &- N_q * rw\_int\_nbq^{(m)}]/work_{i,j} \end{aligned}$$

- AVG1 averages:
  - $rhobar\_nbq\_avg1^{(n+1)} = rhobar\_nbq^{m+1}$
  - $rho\_nbq\_avg1^{(n+1)} = 1 + rho^{(n)}/\rho_0 + rho\_nbq^{(m+1)}/Hzr^{(m+1)}$
- Update $ru\_int\_nbq^{(n+1)}$:

$$\begin{aligned} ru\_int\_nbq^{(n+1)} = &ru\_int\_nbq^{(n+1)} \\ &- ruext\_nbq\_2d\_old\ (Hz_{i,j}^{(m+1)} + Hz_{i-1,j}^{(m+1)}) \end{aligned}$$

- AVG2 averages:
  - u-velocity:

$$\begin{aligned} ru\_nbq\_avg2^{(n+\frac{1}{2})} = \Delta x \Delta y[ \\ &(qdmu\_nbq^{(m+1)} - ru\_nbq\_avg2^{(*)})/\Delta t \\ &- ru\_int\_nbq^{(n+1)} \\ &- ru\_ext\_nbq\_sum\ \frac{Hz_{i,j}^{(m+1)} + Hz_{i-1,j}^{(m+1)}}{N_q}] \end{aligned}$$

  - w-velocity:

$$\begin{aligned} rw\_nbq\_avg2^{(n+\frac{1}{2})} = \Delta x \Delta y[ \\ &\frac{qdmw\_nbq^{(m+1)} - rw\_nbq\_avg2^{(*)}}{\Delta t} \\ &- rw\_int\_nbq^{(m+1)}] \end{aligned}$$

### 4.5 Depth-averaged velocity & forcing from fast-mode

- $DUnew^{(m+1)} = 2\ DU\_nbq^{(m+1)}$
- (2D) depth-averaged velocities:
  - Update $ubar^{m+1}$. This variable is the hydrostatic and ZETA2D pronostic variable for depth-averaged velocity. With ZETAW algorithm, depth-averaged velocity is diagnosed from new pronostic variable $qdmu\_nbq^{(m+1)}$:

$$ubar^{m+1} = \frac{DUnew^{(m+1)}}{Dnew_{i,j}^{(m+1)} + Dnew_{i-1,j}^{(m+1)}}$$

  - $DU\_avg1^{n+1} = DU\_avg1^{n+1} + \frac{1}{2} weight^{(avg1)} * \Delta x\ DUnew^{(m+1)}$
  - $DU\_avg2^{n+1} = DU\_avg2^{n+1} + \frac{1}{2} weight^{(avg2)} * \Delta x\ DUnew^{(m+1)}$
- Depth-averaged momentum nudging & Body force.
- Lateral boundary conditions:
  call **u2dbc**, $BC\ for\ Dnew^{(m+1)}$, $DU\_avg1^{n+1}$, $DU\_avg2^{n+1}$.

### 4.6 Adjust $\zeta^{m+1}$ using $div\bar{u}$ & update grid

Once rhobar_nbq and depth-averaged momentum has been updated, surface anomaly can be adjusted to satisfy the low frequency (internal-mode time-step) version of mass conservation equation. This adjustment is however not sufficient to satisfy conservation at machine precision and a final correction (Hz) is needed.

This re-computation of $\zeta^{m+1}$ using $div\bar{u}$ is only done for the PRECISE option. In PERF option, $\zeta^{m+1}$ is computed only once based on the surface characteristic relation. But the final numerical correction is applied in all cases.

- $Dnew^{m+1} = (H + zeta^{m+1})rhobar\_nbq^{m+1}$
- Update Zeta:

$$
\begin{aligned}
zeta^{m+1} = \frac{1}{rhobar\_nbq^{m+1}} \Big[ & (H + zeta^{m})rhobar\_nbq^{m} \\
& + \Delta t_{fast} \frac{Dnew^{m+1}_{i,j} + Dnew^{m+1}_{i-1,j}}{2\Delta x \Delta y} ubar^{m+1} \Delta y^{(u)} \Big] \\
& - H
\end{aligned}
$$

- Set closed boundary conditions for $zeta^{m+1}$
- $Zt\_avg1^{n+1} = zeta^{m+1}$
- call **set_depth**: update $Hz^{(m+1)}$
- call **grid_nbq**:
  - $Hzw^{m+1}_{i,j,k} = z\_r^{m+1}_{i,j,k+1} - z\_r^{m+1}_{i,j,k}$
  - Upper & lower BC: $Hzw^{m+1}_{i,j,0} = z\_r^{m+1}_{i,j,1} - z\_w^{m+1}_{i,j,0}$ & $Hzw^{m+1}_{i,j,N} = z\_w^{m+1}_{i,j,N} - z\_r^{m+1}_{i,j,N}$
  - Caution: $Hzw^{m+1}$ is stored in Hzw_half_nbq ( "half" is usually for step $m + \frac{1}{2}$)
  - Compute: Hzr_half_nbq_inv(Hzr), Hzu_half_qdmu_nbq($Hzr_u$), Hzw_half_nbq_inv($Hz_w$)
- Point source for river runoff.
- Copy density for extrapolation (last fast step): $rho\_bak = rho$
- Correct $Hz^{n+1}$ (for internal mode only) by inverting internal continuity equation.