Momentum, Kinetic Energy and Barotropic Vorticity online Diagnostics for CROCO

Jonathan Gula

November 15, 2020

Contents

The physical meaning and numerical implementation of the cpp key DIAGNOSTICS_UV, DIAGNOSTICS_EK and DIAGNOSTICS_VRT in CROCO are described below.

- DIAGNOSTICS_UV = outputs 3d terms from the momentum equation in a separate file.
- DIAGNOSTICS_EK = outputs vertically integrated terms from the kinetic energy equation in a separate file. The DIAGNOSTICS_EK_FULL is optional. It impacts the computation of the terms but do not impact the choice or the size of the outputted fields (see below). The key DIAGNOSTICS_EK_MLD can be added to also output terms averaged over the depth of the surface mixed-layer.
- DIAGNOSTICS_VRT = outputs 2d terms from the barotropic vorticity equation in a separate file.
- DIAGNOSTICS_PV = outputs right-hand-sides of momentum and T/S equations in 3d.
- DIAGNOSTICS_DISS = same than DIAGNOSTICS_PV, but outputs are rescaled as energy and buoyancy tendencies .
- DIAGNOSTICS_BARO = isolate contribution from the barotropic/baroclinic coupling for kinetic energy and/or momentum
- DIAGNOSTICS_EDDY = outputs (time-averaged) quadratic quantities for u,v,w,b,T,S. etc.

The options need to be activated in the cppdefs.h file:

```
# define DIAGNOSTICS_TS
# define  DIAGNOSTICS_UV
# ifdef DIAGNOSTICS_TS
#  define  DIAGNOSTICS_TS_ADV
#  undef  DIAGNOSTICS_TS_MLD
# endif

# define  DIAGNOSTICS_VRT
# define  DIAGNOSTICS_EK
# ifdef DIAGNOSTICS_EK
#  define DIAGNOSTICS_EK_FULL
#  undef DIAGNOSTICS_EK_MLD
# endif

# define DIAGNOSTICS_BARO
# define DIAGNOSTICS_PV

# define DIAGNOSTICS_DISS
# ifdef DIAGNOSTICS_DISS
#  define DIAGNOSTICS_PV
#  define DIAGNOSTICS_PV_FULL
# endif

# define DIAGNOSTICS_EDDY

# undef TENDENCY
```

```
# ifdef TENDENCY
#  define DIAGNOSTICS_UV
# endif
```

and require the following addition to the namelist (croco.in):

```
diagnosticsM:   ldefdiaM   nwrtdiaM    nrpfdiaM /filename
                  F          0          0
                                sarga_diaM.nc

diagM_avg: ldefdiaM_avg  ntsdiaM_avg  nwrtdiaM_avg  nprfdiaM_avg /filename
              F          1           0            0
                                sarga_diaM_avg.nc

diagM_history_fields: diag_momentum(1:2)
                           T T

diagM_average_fields: diag_momentum_avg(1:2)
                           T T

diags_ek:   ldefdiags_ek, nwrtdiags_ek, nrpfdiags_ek /filename
                  T          72         100
                                croco_diags_ek.nc

diags_ek_avg: ldefdiags_ek_avg  ntsdiags_ek_avg  nwrtdiags_ek_avg  nprfdiags_ek_avg /filename
              T          1           10          100
                                croco_diags_ek_avg.nc

diags_ek_history_fields: diags_ek
                            T

diags_ek_average_fields: diags_ek_avg
                            T

diags_vrt:   ldefdiags_vrt, nwrtdiags_vrt, nrpfdiags_vrt /filename
                  T          72         100
                                croco_diags_vrt.nc

diags_vrt_avg: ldefdiags_vrt_avg  ntsdiags_vrt_avg  nwrtdiags_vrt_avg  nprfdiags_vrt_avg /filename
              T          1          72          100
                                croco_diags_vrt_avg.nc

diags_vrt_history_fields: diags_vrt
                            T

diags_vrt_average_fields: diags_vrt_avg
                            T

diags_pv:   ldefdiags_pv, nwrtdiags_pv, nrpfdiags_pv /filename
                  T         4320        5
                                PV/gigatl1_diags_pv.nc

diags_pv_avg: ldefdiags_pv_avg  ntsdiags_pv_avg  nwrtdiags_pv_avg  nprfdiags_pv_avg /filename
              T          1           0          0
                                PV/gigatl1_diags_pv_avg.nc

diags_pv_history_fields: diags_pv
                           2*T

diags_pv_average_fields: diags_pv_avg
                          2*T



diags_eddy:   ldefdiags_eddy, nwrtdiags_eddy, nrpfdiags_eddy /filename
                  F          0        5
                                EDDY/gigatl1_diags_eddy.nc

diags_eddy_avg: ldefdiags_eddy_avg  ntsdiags_eddy_avg  nwrtdiags_eddy_avg  nprfdiags_eddy_avg /
     filename
              T          1           0          0
                                EDDY/gigatl1_diags_eddy_avg.nc

diags_eddy_history_fields: diags_eddy
                            T

diags_eddy_average_fields: diags_eddy_avg
                            T
```

1 Momentum equation

1.1 Continuous equation

The horizontal momentum equations in the Boussinesq approximation are:

$$\frac{\partial u}{\partial t} = -u_j \frac{\partial u}{\partial x_j} - w\frac{\partial u}{\partial z} + fv - \frac{P_x}{\rho_0} + \mathcal{V}_u + \mathcal{D}_u + \mathcal{S}_u \quad (1)$$

$$\underbrace{\frac{\partial v}{\partial t}}_{rate} = -\underbrace{u_j \frac{\partial v}{\partial x_j}}_{hadv} - \underbrace{w\frac{\partial v}{\partial z}}_{vadv} - \underbrace{fu}_{cor} - \underbrace{\frac{P_y}{\rho_0}}_{Prsgrd} + \underbrace{\mathcal{V}_v}_{vmix} + \underbrace{\mathcal{D}_v}_{hmix} + \underbrace{\mathcal{S}_v}_{nudg} \quad (2)$$

where Cartesian tensor notation with summation convention has been used for $j = 1, 2$; $\vec{u} = (u, v)$ is the horizontal velocity vector, w is the vertical velocity, f is the Coriolis parameter, P is the pressure anomaly, $\vec{\mathcal{V}} = (\mathcal{V}_u, \mathcal{V}_v) = \frac{\partial}{\partial z}\left(K_{Mv}\frac{\partial \vec{u}}{\partial z}\right)$ is the vertical mixing, $\vec{\mathcal{D}} = (\mathcal{D}_u, \mathcal{D}_v)$ the horizontal diffusion, and $\vec{\mathcal{S}} = (\mathcal{S}_u, \mathcal{S}_v)$ other sources and sinks (due to restoring, nudging, boundary conditions, etc.).

1.2 Discrete formulation

The model momentum equations computes the momentum at the time-step n+1 [*step3d_uv1.F, step3d_uv2.F*]:

$$H^{n+1}u_i^{n+1} = H^n u_i^n + \Delta t \sum_j M_{i,j}^{n+\frac{1}{2}}$$

where u_i are the horizontal components of the velocity vector ($i = 1, 2$), Δt is the baroclinic time-step of the model, H the vertical grid spacing, and $M_{i,j}^{n+\frac{1}{2}}$ are the terms from the momentum equation (with units of velocities $\times$ cell height / time) corresponding to horizontal advection, vertical advection, coriolis force, pressure gradient, vertical mixing, horizontal diffusion (implicit and/or explicit), and various sources and sinks.

The terms are divided by H^{n+1} before writing in the file such that they all have dimensions of m s^{-2}. Variables included in the croco_diags_uv.nc files are:

- u_rate, v_rate = rate of change of momentum [*step3d_uv2.F*]
- u_xadv, v_xadv = advection + implicit dissipation along xi- axis [*rhs3d.F*]

- u_yadv, v_yadv = advection + implicit dissipation along eta- axis [*rhs3d.F*]
- u_vadv, v_vadv = vertical advection [*rhs3d.F*]
- u_cor, v_cor = Coriolis term + grid curvature terms (see CURVGRID) [*rhs3d.F*]
- u_prsgrd, v_prsgrd = Pressure gradient [*prsgrd.F*]
- u_hmix, v_hmix = Horizontal mixing (explicit) [*uv3dmix4_GP.F, uv3dmix_GP.F, uv3dmix_spg.F, uv3dmix4_S.F, uv3dmix_S.F*]
- u_vmix, v_vmix = Vertical mixing [*step3d_uv2.F*]
- u_nudg, v_nudg = Nudgind, restoring, boundary conditions, etc. [*step3d_uv2.F*]
- u_hdiff, v_hdiff = Horizontal mixing (implicit) [*rhs3d.F*]; it is also included in advective terms, this term corresponds to the diffusive part of the advection, and is evaluated by taking the difference with a centered scheme (C4 for UP3 and C6 for UP5). Not implemented for WENO5.
- u_baro, v_baro = barotropic/baroclinic coupling [*step3d_uv2.F*]; this is already included in u_vmix, v_vmix and outputted as an additional term if DIAGNOSTICS_BARO is activated
- u_fast, v_fast = contribution from the fast momentum time stepping [*step3d_fast.F*] *[if M3FAST is activated, it includes in particular the bottom drag part]*

All variables are 3D on horizontal u- and v- grids and vertical rho-grid (N levels).

The following pointwise budgets are closed:

u_rate = u_xadv + u_yadv + u_vadv + u_Prsgrd + u_cor + u_vmix + u_hmix + u_nudg + u_fast
v_rate = v_xadv + v_yadv + v_vadv + v_Prsgrd + v_cor + v_vmix + v_hmix + v_nudg+ v_fast

2 Kinetic energy equation

2.1 Continuous equation

The kinetic energy equation is formed by taking the inner product of the horizontal velocities with the momentum equations:

$$\frac{1}{2}\frac{\partial u_i^2}{\partial t} + u_j \frac{\partial \frac{1}{2} u_i^2}{\partial x_j} + w \frac{\partial \frac{1}{2} u_i^2}{\partial z} = -\frac{u_i}{\rho_0}\frac{\partial P}{\partial x_i} + \mathcal{V}_i u_i + \mathcal{D}_i u_i + \mathcal{S}_i u_i \qquad (3)$$

where Cartesian tensor notation with summation convention has been used, $i = 1, 2$, $j = 1, 2$; u_i are the horizontal components of the velocity vector u_j; $u_3 = w$ is the vertical velocity.

Variables included in the croco_diags_ek.nc files are:

- ek_rate = rate of change of depth integrated kinetic energy, $\frac{\partial}{\partial t}\int_{-h}^{\zeta} \frac{1}{2} u_i^2 dz$
- ek_hadv = $\int_{-h}^{\zeta} u_i \frac{\partial \frac{1}{2} u_i^2}{\partial x_i} dz$
- ek_vadv = $\int_{-h}^{\zeta} w \frac{\partial \frac{1}{2} u_i^2}{\partial z} dz$
- ek_prsgrd = $\int_{-h}^{\zeta} -u_i \frac{\partial P}{\partial x_j} dz$
- ek_vmix = $\int_{-h}^{\zeta} \mathcal{V}_i u_i dz$
- ek_hmix = explicit part of $\int_{-h}^{\zeta} \mathcal{D}_i u_i dz$
- ek_hdiff = implicit part of $\int_{-h}^{\zeta} \mathcal{D}_i u_i dz$ [already included in ek_hadv]
- ek_nudg = $\int_{-h}^{\zeta} \mathcal{S}_i u_i dz$, other sources and sinks such as nudging and open boundary conditions
- ek_vol = depth integrated kinetic energy variation due to the grid breezing ($\frac{\partial \zeta}{\partial t}$)
- ek_cor = $\int_{-h}^{\zeta}(fuv - fvu)dz$, *i.e.,* zero at the continuous level, but not formally zero in the model due to the discretization of the grid
- ek_fast = contribution from the fast momentum time stepping [*step3d_fast.F*] *[only if M3FAST is activated, it includes in particular the bottom drag part and has to be included in the sum]*

- ek_baro [already included in ek_vmix]
- ek_wind [already included in ek_vmix]
- ek_drag [already included in ek_vmix]

Such that the full closed budget is:

ek_rate = ek_hadv + ek_vadv + ek_prsgrd + ek_vmix + ek_hmix + ek_hdiff + ek_nudg + ek_cor + ek_vol

Tems ek_vol and ek_cor should both be negligible and are kept only for consistency check. Note that the discretization of the Coriolis term does not unsure pointwise cancelation of ek_cor but should unsure area averages cancellation. Area average cancellation is perfect for closed boundary conditions but there might be a residual for open boundary conditions.

2.2 Discrete formulation

The kinetic energy equation terms correspond to the momentum equation terms multiplied by the velocity at the time-step $n+\frac{1}{2}$ [*step3d_uv2.F*]:

$$\frac{1}{2}H^{n+1}(u_i^{n+1})^2 = \frac{1}{2}H^n(u_i^n)^2 + \Delta t \sum_j u^{n+\frac{1}{2}} M_{i,j}^{n+\frac{1}{2}}$$

The output terms are $u^{n+\frac{1}{2}} M_{i,j}^{n+\frac{1}{2}}$. They are vertically integrated and multiplied by the cell surface $dm\,dn = \frac{1}{pm\,pn}$, such that they correspond to volume energy tendencies. They all have dimensions of [energy × volume / time] ($\mathrm{m^5 s^{-3}}$). The exact formulation of the terms is:

- ek_rate = $\sum_{k=1}^{N} \left(\frac{1}{2}H^{n+1}(u_i^{n+1})^2 - \frac{1}{2}H^n(u_i^n)^2\right)/\Delta t$
- ek_hadv = $\sum_{k=1}^{N} u_i^{n+1/2} M_{i,\mathrm{hadv}}^{n+1/2}$
- ek_vadv = $\sum_{k=1}^{N} u_i^{n+1/2} M_{i,\mathrm{vadv}}^{n+1/2}$
- ek_Prsgrd = $\sum_{k=1}^{N} u_i^{n+1/2} M_{i,\mathrm{Prsgrd}}^{n+1/2}$
- ek_vmix = $\sum_{k=1}^{N} u_i^{n+1/2} M_{i,\mathrm{vmix}}^{n+1/2}$
- ek_hmix = $\sum_{k=1}^{N} u_i^{n+1/2} M_{i,\mathrm{hmix}}^{n+1/2}$

- ek_hdiff $= \sum_{k=1}^{N} u_i^{n+1/2} M_{i,\text{hdiff}}^{n+1/2}$ [already included in ek_hadv]
- ek_nudg $= \sum_{k=1}^{N} u_i^{n+1/2} M_{i,\text{nudg}}^{n+1/2}$
- ek_vol $= \sum_{k=1}^{N} \left(\frac{H^n - H^{n+1}}{2} \left[\frac{H^n}{H^{n+1}} (u^n)^2 + 2u_n \sum_j \frac{M_j^{n+1/2}}{H^{n+1}} \right] \right)$
- ek_cor $= \sum_{k=1}^{N} u_i^{n+1/2} M_{i,\text{cor}}^{n+1/2}$
- ek_fast $= \sum_{k=1}^{N} u_i^{n+1/2} M_{i,\text{fast}}^{n+1/2}$
- ek_baro $= \sum_{k=1}^{N} u_i^{n+1/2} M_{i,\text{baro}}^{n+1/2}$ [already included in ek_vmix]
- ek_wind $= \tau_i^s u_i^{n+1/2}$ [already included in ek_vmix]
- ek_drag $= \tau_i^b u_i^{n+1/2}$ [already included in ek_vmix]

Spatial discretization: Momentum terms and velocities are first computed on their native u- and v-grids, then energy terms are computed on the rho-grid:

$$\begin{aligned} \text{ek\_hadv} = & \int_{-h}^{\zeta} \frac{1}{2} \left[u(i,j) M_{u,\text{hadv}}^{n+1/2}(i,j) + u(i+1,j) M_{u,\text{hadv}}^{n+1/2}(i+1,j) \right] dz \\ & + \int_{-h}^{\zeta} \frac{1}{2} \left[v(i,j) M_{v,\text{hadv}}^{n+1/2}(i,j) + v(i,j+1) M_{v,\text{hadv}}^{n+1/2}(i,j+1) \right] dz \end{aligned} \quad (4)$$

2.2.1 DIAGNOSTICS_EK_FULL

if DIAGNOSTICS_EK_FULL is defined, the terms from the momentum equations $M_i^{n+1/2}$ computed during time-step n, are multiplied by $\frac{u^{n+1}+u^n}{2}$. This option has the disadvantage to require several 3d arrays during the online computation to save the $M_i^{n+1/2}$ terms from the momentum equation until the end of the time-step (because the vertical integration requires u^{n+1}).

if DIAGNOSTICS_EK_FULL is NOT defined, the velocities used to multiply the momentum terms are the velocities computed after the predictor step $u^{n+1/2}$. This option has the advantage to only use 2d arrays during the online computation as the vertical integration is performed directly, but the balance will not be perfectly closed because $u^{n+1/2} \neq \frac{u^{n+1}+u^n}{2}$

2.2.2 DIAGNOSTICS_EK_MLD

This adds energy terms averaged over the mixed layer depth (currently defined as the KPP hbl).

2.2.3 DIAGNOSTICS_PV or DIAGNOSTICS_DISS

The option DIAGNOSTICS_PV outputs the sum of the non-conservative terms in 3d:

M_rhs = (M_vmix - M_baro) + M_fast + M_nudg + M_hmix + M_hdiff

The option DIAGNOSTICS_DISS outputs the same thing, but multiplies all terms by momentum to get kinetic energy tendencies (still in 3d):

ek_rhs = ek_vmix - ek_vmix_trans - ek_baro + ek_fast + ek_nudg + ek_hmix + ek_hdiff

The new term ek_vmix_trans corresponds to the transport part of the vertical mixing term.

The term ek_vmix, once vertically integrated, corresponds to the contribution of the wind (which can be positive), the bottom drag (always negative), and a dissipative part (which is always negative definite), as seen below:

$$\begin{aligned} ek\_vmix &= \int_{-h}^{\zeta} \mathcal{V}_i u_i dz \\ &= \int_{-h}^{\zeta} u_i \frac{\partial}{\partial z}\left(K_M \frac{\partial u_i}{\partial z}\right) dz \\ &= \left[u_i K_M \frac{\partial u_i}{\partial z}\right]_{-h}^{\zeta} - \int_{-h}^{\zeta} K_M \left(\frac{\partial u_i}{\partial z}\right)^2 dz \\ &= \tau_i^s u_i^s - \tau_i^b u_i^b - \int_{-h}^{\zeta} K_M \left(\frac{\partial u_i}{\partial z}\right)^2 dz \qquad (5) \end{aligned}$$

However if we compute it in 3d, it is not strictly negative (even without wind) and a vertical transport part is included:

$$
\begin{aligned}
\mathcal{V}_i u_i &= u_i \frac{\partial}{\partial z}\left(K_M \frac{\partial u_i}{\partial z}\right) \\
&= \frac{\partial}{\partial z}\left(u_i K_M \frac{\partial u_i}{\partial z}\right) - K_M \left(\frac{\partial u_i}{\partial z}\right)^2 \\
&= \underbrace{\frac{\partial}{\partial z}\left(K_M \frac{1}{2}\frac{\partial u_i^2}{\partial z}\right)}_{ek\_vmix\_trans} - \underbrace{K_M \left(\frac{\partial u_i}{\partial z}\right)^2}_{ek\_vmix - ek\_vmix\_trans}
\end{aligned}
\tag{6}
$$

If DIAGNOSTICS_PV_FULL is activated, the term ek_vmix_trans is not substracted from the ek_rhs, but is outputted separately (this was a temporary fix to unsure consistency with previously computed diagnostics).

2.3 Mean and Eddy

Examples of kinetic energy budget with CROCO using these diagnostics can be found in Gula *et al.* (2016), where the equation is further decomposed into mean and eddy parts following Harrison & Robinson (1978).

The mean kinetic energy of the flow KE $= \frac{1}{2}(\overline{u^2} + \overline{v^2})$ is the sum of the kinetic energy of the mean flow, MKE $= \frac{1}{2}(\overline{u}^2 + \overline{v}^2)$, and the eddy kinetic energy, EKE $= \frac{1}{2}(\overline{u'^2} + \overline{v'^2})$, where the overbar denotes a time average, and the prime denotes fluctuations relative to the time average.

The mean kinetic energy equation is formed by taking the inner product of the mean horizontal velocities with the mean terms in the momentum equations:

$$
\overline{u}_i \overline{\frac{\partial u_i}{\partial \mathrm{t}}} + \overline{u}_i \overline{u_j \frac{\partial u_i}{\partial x_j}} = -\frac{\overline{u}_i}{\rho_0}\frac{\partial \overline{P}}{\partial x_i} + \overline{\mathcal{V}_i}\overline{u_i} + \overline{\mathcal{D}_i}\overline{u_i} + \overline{\mathcal{S}_i}\overline{u_i}, \tag{7}
$$

where Cartesian tensor notation with summation convention has been used, $i = 1, 2$, $j = 1, 2, 3$; u_i are the horizontal components of the velocity vector u_j; $u_3 = w$ is the vertical velocity; p is the pressure anomaly; $b = -\frac{g\rho}{\rho_0}$ is the buoyancy anomaly; $\mathcal{V}_i$, $\mathcal{D}_i$, and $\mathcal{S}_i$ are the vertical mixing, horizontal diffusion, and forcing terms in the horizontal momentum equations.

We can further decompose the advective terms as:

$$
\begin{aligned}
\overline{u}_i \overline{u_j \frac{\partial u_i}{\partial x_j}} &= \overline{u}_i \overline{u}_j \frac{\partial \overline{u}_i}{\partial x_j} + \overline{u}_i \overline{u'_j \frac{\partial u'_i}{\partial x_j}} \\
&= \overline{u}_j \frac{\partial \frac{1}{2} \overline{u}_i^2}{\partial x_j} + \overline{u}_i \overline{u'_j \frac{\partial u'_i}{\partial x_j}} \\
&= \frac{\partial \frac{1}{2} \overline{u}_j \overline{u}_i^2}{\partial x_j} + \overline{u}_i \frac{\partial \overline{u'_j u'_i}}{\partial x_j}
\end{aligned}
$$

with summation convention $i = 1, 2,\ j = 1, 2, 3$; which shows that the contribution of the advective term can be split into the divergence of an energy flux corresponding to the advection of MKE by the mean flow ($\frac{\partial \frac{1}{2} \overline{u}_j \overline{u}_i^2}{\partial x_j}$), and another term corresponding to the conversion of eddy kinetic energy to mean kinetic energy ($\overline{u}_i \frac{\partial \overline{u'_j u'_i}}{\partial x_j}$).

In practice we can diagnose the left hand side ($\overline{u}_i \overline{u_j \frac{\partial u_i}{\partial x_j}}$) from the online diagnostics of momentum and easily compute $\frac{\partial \frac{1}{2} \overline{u}_j \overline{u}_i^2}{\partial x_j}$ from the mean fields to get the second term on the right hand side as a residual ($\overline{u}_i \frac{\partial \overline{u'_j u'_i}}{\partial x_j}$).

We can also decompose the pressure gradient terms as:

$$
-\frac{u_i}{\rho_0} \frac{\partial P}{\partial x_i} = -\frac{\partial \left(\frac{1}{\rho_0} u_j p \right)}{\partial x_j} + wb
$$

with summation convention $i = 1, 2,\ j = 1, 2, 3$; which shows that the contribution of the pressure gradient term can be split into the divergence of an energy flux at the boundaries of the domain volume ($\frac{\partial \left(\frac{1}{\rho_0} u_j p \right)}{\partial x_j}$) and conversion between potential and kinetic energy (wb).

Again, in practice it is easier to diagnose the left hand side ($\frac{\overline{u}_i}{\rho_0} \frac{\partial \overline{P}}{\partial x_i}$) from the online diagnostics of momentum and compute $\overline{w}\overline{b}$ from the mean fields to get the first term on the right hand side as a residual ($\frac{\partial \frac{1}{\rho_0} \overline{u}_j \overline{p}}{\partial x_j}$).

Finally we get the equation for MKE:

$$\overline{u}_i \overline{\frac{\partial u_i}{\partial \mathrm{t}}} + \underbrace{\frac{\partial\left(\frac{1}{2}\overline{u}_j\overline{u}_i^2 + \frac{1}{\rho_0}\overline{u}_j\overline{p}\right)}{\partial x_j}}_{\text{Boundary Transport}} =$$

$$\underbrace{-\overline{u}_i \frac{\partial \overline{u_j' u_i'}}{\partial x_j}}_{\text{EKE}\rightarrow\text{MKE}} + \underbrace{\overline{w}\overline{b}}_{\text{MPE}\rightarrow\text{MKE}} + \underbrace{\overline{\mathcal{V}_i}\overline{u_i}}_{\text{Vertical mixing}} + \underbrace{\overline{\mathcal{D}_i}\overline{u_i}}_{\text{Horizontal diffusion}} + \underbrace{\overline{\mathcal{S}_i}\overline{u_i}}_{\text{sources/sinks}}, \quad (8)$$

.

The eddy kinetic energy equation is formed by subtracting the energy equation of the mean flow from that of the total flow:

$$\overline{u_i \frac{\partial u_i}{\partial \mathrm{t}}} - \overline{u}_i \overline{\frac{\partial u_i}{\partial \mathrm{t}}} + \overline{u_i u_j \frac{\partial u_i}{\partial x_j}} - \overline{u}_i \overline{u_j \frac{\partial u_i}{\partial x_j}} = -\overline{\frac{u_i}{\rho_0}\frac{\partial P}{\partial x_i}} + \frac{\overline{u}_i}{\rho_0}\frac{\partial \overline{P}}{\partial x_i}$$
$$+ \overline{\mathcal{V}_i u_i} - \overline{\mathcal{V}_i}\overline{u_i} + \overline{\mathcal{D}_i u_i} - \overline{\mathcal{D}_i}\overline{u_i} + \overline{\mathcal{S}_i u_i} - \overline{\mathcal{S}_i}\overline{u_i}, \quad (9)$$

Where the advective term becomes:

$$\overline{u_i u_j \frac{\partial u_i}{\partial x_j}} - \overline{u}_i \overline{u_j \frac{\partial u_i}{\partial x_j}} = \frac{\partial\left(\frac{1}{2}\overline{u_j}\,\overline{u_i'^2} + \frac{1}{2}\overline{u_j' u_i'^2}\right)}{\partial x_j} + \overline{u_j' u_i'}\frac{\partial \overline{u_i}}{\partial x_j}$$

which shows that the contribution of the advective term can be split into the divergence of an energy flux corresponding to the advection of EKE by the flow ($\frac{\partial\left(\frac{1}{2}\overline{u_j}\,\overline{u_i'^2}+\frac{1}{2}\overline{u_j' u_i'^2}\right)}{\partial x_j}$), and another term corresponding to the conversion of mean kinetic energy to eddy kinetic energy ($\overline{u_j' u_i'}\frac{\partial \overline{u_i}}{\partial x_j}$).

In practice we can diagnose the left hand side ($\overline{u_i u_j \frac{\partial u_i}{\partial x_j}} - \overline{u}_i \overline{u_j \frac{\partial u_i}{\partial x_j}}$) exactly from the online diagnostics of kinetic energy and momentum. The first and third terms on the right hand side can also be recomputed using the DIAGNOSTICS_EDDY capability (which outputs $\overline{u_j' u_i'} = \overline{u_j u_i} - \overline{u_j}\overline{u_i}$), such that the second term on the right hand side can be computed as the residual.

And the pressure gradient term becomes:

$$-\overline{\frac{u_i}{\rho_0}\frac{\partial P}{\partial x_i}} + \frac{\overline{u}_i}{\rho_0}\frac{\partial \overline{P}}{\partial x_i} = \overline{w' b'} - \frac{\partial\left(\frac{1}{\rho_0}\overline{u_j' p'}\right)}{\partial x_j}$$

which shows that the contribution of the pressure gradient term on EKE can be split into the divergence of an energy flux due to pressure fluctuations ($\frac{\partial\left(\frac{1}{\rho_0}\overline{u'_j p'}\right)}{\partial x_j}$) and conversion from eddy potential to eddy kinetic energy ($\overline{w'b'}$).

In practice we can diagnose the left hand side ($-\overline{\frac{u_i}{\rho_0}\frac{\partial P}{\partial x_i}} + \frac{\overline{u}_i}{\rho_0}\frac{\partial \overline{P}}{\partial x_i}$) exactly from the online diagnostics of kinetic energy and momentum. The first term on the right hand side can also be recomputed using the DIAGNOSTICS_EDDY capability (which outputs $\overline{w'b'} = \overline{wb} - \overline{w}\overline{b}$), such that the second term on the right hand side can be computed as the residual.

So finally the EKE equations is:

$$\overline{u'_i\frac{\partial u'_i}{\partial \mathrm{t}}} + \underbrace{\frac{\partial\left(\frac{1}{2}\overline{u_j}\,\overline{u'^2_i} + \frac{1}{2}\overline{u'_j u'^2_i} + \frac{1}{\rho_0}\overline{u'_j p'}\right)}{\partial x_j}}_{\text{Boundary Transport}} =$$

$$\underbrace{-\overline{u'_j u'_i}\frac{\partial \overline{u_i}}{\partial x_j}}_{\text{MKE}\rightarrow\text{EKE}} + \underbrace{\overline{w'b'}}_{\text{EPE}\rightarrow\text{EKE}} + \underbrace{\overline{\mathcal{V}'_i u'_i}}_{\text{Vertical mixing}} + \underbrace{\overline{\mathcal{D}'_i u'_i}}_{\text{Horizontal diffusion}} + \underbrace{\overline{\mathcal{S}'_i u'_i}}_{\text{sources/sinks}} \quad (10)$$

3 Barotropic vorticity equation

3.1 Continuous equation

The full barotropic vorticity balance equation of the flow is obtained by integrating the momentum equations in the vertical and cross differentiating them:

$$\underbrace{\frac{\partial \Omega}{\partial t}}_{\text{rate}} = -\underbrace{\vec{\nabla}.(f\vec{u})}_{\text{planet. vort. adv.}} + \underbrace{\frac{\mathbf{J}(P_b, h)}{\rho_0}}_{\text{bot. pres. torque}} + \underbrace{\vec{k}.\vec{\nabla}\times\frac{\vec{\tau}^{wind}}{\rho_0}}_{\text{wind curl}} - \underbrace{\vec{k}.\vec{\nabla}\times\frac{\vec{\tau}^{bot}}{\rho_0}}_{\text{bot. drag curl}}$$

$$+ \underbrace{\mathcal{D}_\Sigma}_{\text{horiz.diffusion.}} - \underbrace{A_\Sigma}_{\text{NL advection}} \quad (11)$$

where the barotropic vorticity is defined as the vorticity of the vertically integrated velocities[1]

$$\Omega = \frac{\partial \overline{v}}{\partial x} - \frac{\partial \overline{u}}{\partial y}$$

with (u, v) the (x, y) components of the horizontal flow, and the overbar denotes a vertically integrated quantity,

$$\overline{u} = \int_{-h}^{\zeta} u \, dz,$$

where $\zeta(x, y, t)$ is the free-surface height and $h(x, y) > 0$ the depth of the resting topography. $H(i, j, t) = \int_{-h}^{\zeta} dz = \zeta(i, j, t) + h(i, j)$ is the total depth of the water column. Finally, the curl of non-linear advection terms can be written as

$$A_{\Sigma} = \frac{\partial^2(\overline{vv} - \overline{uu})}{\partial x \partial y} + \frac{\partial^2 \overline{uv}}{\partial x \partial x} - \frac{\partial^2 \overline{uv}}{\partial y \partial y},$$

and $\mathcal{D}_{\Sigma}$ is the term due to the horizontal diffusion in the model implicitly part of the advective scheme, plus eventually some explicit diffusion.

Examples of barotropic vorticity budget with CROCO using these diagnostics and interpretations can be found in Gula *et al.* (2015) and Schoonover *et al.* (2016).

3.2 Discrete formulation

Variables included in the croco_diags_vrt.nc files are:

- vrt_rate = rate of change of barotropic vorticity [*step3d_uv2.F*]
- vrt_xadv = contribution of advection + implicit dissipation along xi-axis+ grid curvature terms (see CURVGRID) [*rhs3d.F*]
- vrt_hdiff = implicit dissipation along xi- and eta- axis [*rhs3d.F*] [already included in vrt_xadv+ vrt_yadv]

[1] Note that the barotropic vorticity is not identical to the vertically integrated vorticity. The curl and the vertical integration can be interchanged at the expense of introducing terms due to the horizontal variations of the limits of the integral. The difference $\Omega - \overline{\zeta} = \vec{u_s} \times \vec{\nabla}\zeta + \vec{u_b} \times \vec{\nabla} h$, where $\vec{u_s}$ and $\vec{u_b}$ are the horizontal velocities at the surface and bottom, respectively, can be non-negligible at places where we have both significant bottom currents and large topography slopes.

- vrt_cor = planetary vorticity advection [*rhs3d.F*]
- vrt_Prsgrd = bottom Pressure torque [*prsgrd.F*]
- vrt_hmix = contribution of Horizontal diffusion (explicit) [*uv3dmix4_GP.F, uv3dmix_GP.F, uv3dmix_spg.F, uv3dmix4_S.F, uv3dmix_S.F*]
- vrt_vmix = contribution of Vertical mixing = vrt_Wind + vrt_Drag [*step3d_uv2.F*]
- vrt_nudg = contribution of Nudging, restoring, boundary conditions, etc. [*step3d_uv2.F*]
- vrt_Wind = Wind stress curl [*step3d_uv2.F*] [already included in vrt_vmix]
- vrt_Drag = Bottom drag curl [*step3d_uv2.F*] [already included in vrt_vmix]

All variables are 2D on horizontal psi-grid. The following pointwise budget is closed:

vrt_rate = vrt_xadv + vrt_yadv + vrt_Prsgrd + vrt_cor + vrt_vmix + vrt_hmix + vrt_nudg

Spatial discretization: Momentum terms are first computed and vertically averaged on their native u- and v-grids, then vorticity terms are computed on the psi-grid:

4 Tracer equation

4.1 Continuous equation

The Tracer equation is:

$$\underbrace{\frac{\partial C}{\partial t}}_{rate} = \underbrace{-u_j\frac{\partial C}{\partial x_j}}_{hadv} \underbrace{-w\frac{\partial C}{\partial z}}_{vadv} + \underbrace{\mathcal{V}_C}_{vmix} + \underbrace{\mathcal{D}_C}_{horiz.diff.} + \underbrace{\mathcal{S}_C}_{nudg} \tag{12}$$

where Cartesian tensor notation with summation convention has been used for $j = 1, 2$; $\vec{u} = (u, v)$ is the horizontal velocity vector, w is the vertical velocity, $\mathcal{V}_C = \frac{\partial}{\partial z}\left(K_C\frac{\partial \vec{C}}{\partial z}\right)$ is the vertical mixing, $\mathcal{D}_C$ the horizontal diffusion, and $\mathcal{S}_C$ other terms in the model that can act as sources or sinks (due to restoring, nudging, boundary conditions, etc.).

The equation for the tracer variance is:

$$\underbrace{\frac{\partial \frac{1}{2}C^2}{\partial t}}_{rate} = \underbrace{-u_j \frac{\partial \frac{1}{2}C^2}{\partial x_j}}_{hadv} \underbrace{-w \frac{\partial \frac{1}{2}C^2}{\partial z}}_{vadv} + \underbrace{C\mathcal{V}_C}_{vmix} + \underbrace{C\mathcal{D}_C}_{horiz.diff.} + \underbrace{C\mathcal{S}_C}_{nudg} \tag{13}$$

which we can write as:

$$\frac{\partial C^2}{\partial t} + u_j \frac{\partial C^2}{\partial x_j} + w \frac{\partial C^2}{\partial z} = 2C\mathcal{V}_C + 2C\mathcal{D}_C + 2C\mathcal{S}_C \tag{14}$$

However the terms on the right are not strictly associated to a decay of tracer variance.

4.1.1 Vertical mixing

The vertical mixing contribution can be decomposed as:

$$\begin{aligned} C\mathcal{V}_{\mathcal{C}} &= C\frac{\partial}{\partial z}\left(K_C \frac{\partial C}{\partial z}\right) \\ &= \frac{\partial}{\partial z}\left(K_C \frac{1}{2}\frac{\partial C^2}{\partial z}\right) - K_C \left(\frac{\partial C}{\partial z}\right)^2 \end{aligned} \tag{15}$$

If vertically integrated, it corresponds to the contribution of the surface/bottom forcings, and a diffusive part (which is always negative definite), as seen below:

$$\begin{aligned} C\_vmix &= \int_{-h}^{\zeta} \mathcal{V} C dz \\ &= \left[\frac{1}{2} K_C \frac{\partial C^2}{\partial z}\right]_{-h}^{\zeta} - \int_{-h}^{\zeta} K_C \left(\frac{\partial C}{\partial z}\right)^2 dz \end{aligned} \tag{16}$$

But to isolate the diffusive part, we need to decompose it at each time step, to be able to write the tracer variance decay as:

$$\frac{\partial C^2}{\partial t} + \frac{\partial u_j C^2}{\partial x_j} + \frac{\partial}{\partial z}\left(wC^2 - K_C \frac{\partial C^2}{\partial z}\right) \;=\; -2K_C \left(\frac{\partial C}{\partial z}\right)^2 + C(\mathcal{D}_C + \mathcal{S}_C)$$

4.1.2 Horizontal diffusion

The horizontal diffusive part will be a mixture of explicit and implicit diffusion. In practice we only have explicit diffusivity in the sponge layers.

References

GULA, J., MOLEMAKER, M.J. & MCWILLIAMS, JAMES C. 2015 Gulf Stream dynamics along the Southeastern U.S. Seaboard. *J. Phys. Oceanogr.* **45** (3), 690–715.

GULA, J., MOLEMAKER, M.J. & MCWILLIAMS, JAMES C. 2016 Topographic generation of submesoscale centrifugal instability and energy dissipation. *Nat. Commu.* **7**, 12811.

HARRISON, D.E. & ROBINSON, A.R. 1978 Energy analysis of open regions of turbulent flows: Mean eddy energetics of a numerical ocean circulation experiment. *Dyn. Atmos. Ocean* **2**, 185–211.

SCHOONOVER, J., DEWAR, W., WIENDERS, N., GULA, J., MCWILLIAMS, JAMES C., MOLEMAKER, M.J., BATES, S., DANABASOGLU, G. & YEAGER, S. 2016 North atlantic barotropic vorticity balances in numerical models. *J. Phys. Oceanogr.* **46**, 289–303.