Kinetic energy backscatter (KEB) parameterizations for NEMO: documentation

Perezhogin P.A.[1]

[1]Marchuk Institute of Numerical Mathematics, Moscow, Russia

Abstract

This document describes the code of KEB parameterizations developed by the author for NEMO ocean model Perezhogin [2020]. Compared to publication, code was significantly refactored, adopted to curvilinear coordinates and partially tested in NNATL12 configuration.

Contents

1 Fast start

1.1 Install NEMO

- Let you have installed libraries `nectdf4, hdf5, zlib` and `xios2`, and they are in system `PATH`. Download NEMO 3.6:

```
svn co http://forge.ipsl.jussieu.fr/nemo/svn/NEMO/releases/release-3.6/NEMOGCM
```

- Go to folder `ARCH` and create file `arch-myarch.fcm` similar to ones presented in this folder, where set compilers (like `mpiifort`) and directly set path to mentioned libraries, for example:

```
%NCDF_INC -I/netcdf4/include
%NCDF_LIB -L/netcdf4/lib64 -L/hdf5/lib64 -lnetcdf -lnetcdff -lhdf5 -lhdf5_hl -lz
%XIOS_INC -I/xios-2.0/inc
%XIOS_LIB -L/xios-2.0/lib -lxios
```

- Compile clean GYRE configuration in folder `GYRE_XIOS`, i.e. in folder `CONFIG` execute:

```
./makenemo -m myarch -r GYRE_XIOS -n GYRE_XIOS -j 8
```

1.2 Import KEB library

- Copy folder `GYRE_KEB` from `this` repository to folder `CONFIG`.
- Compile with command:

```
./makenemo -m myarch -r GYRE_XIOS -n GYRE_KEB -j 8
```

- In folder `GYRE_KEB` there are three experiments:

```
/EXP_bare    - configuration without backscatter
/EXP_AR1     - autoregressive stochastic KEB
/EXP_negvisc - negative viscosity KEB
```

- You need to make link to executable file:

```
ln -s /direct/path/to/GYRE_KEB/BLD/bin/nemo.exe EXP_bare/opa
ln -s /direct/path/to/GYRE_KEB/BLD/bin/nemo.exe EXP_AR1/opa
ln -s /direct/path/to/GYRE_KEB/BLD/bin/nemo.exe EXP_negvisc/opa
```

- Each experiment can be executed with

```
mpirun -n 32 ./opa
```

  This takes approximately 2 hours. Experiments are launched for 30 years at resolution R4 from snapshot of R9 model in statistical equilibrium.

1.3 Check results

- There are `5d` files and time series just to look at with `ncview`.
- If you follow instruction, results are saved as single netcdf files. Copy content of folder `plot_figures` from `this` repository to each experiment folder. Run `Python` script:

```
python processing.py
```

  This gives `result.pdf` with figures which should be compared to figures in Perezhogin [2020]. Scripts

```
processing_AR1.py, processing_negvisc.py
```

being applied to respective experiments will give KEB statistics. Examples of these figures together with paper Perezhogin [2020] are given in `this` repository in folder `materials`.

Troubleshooting

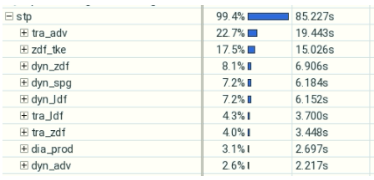
I suppose you have python 3 installed. Most likely, you will need to load additional modules using command (no need for `root`):

```
python -m pip install netCDF4 --user
python -m pip install matplotlib --user
python -m pip install numpy --user
```

Possibly, some libraries else... see error if it exists.

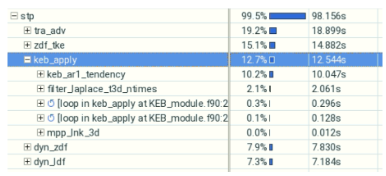
2 Code overview

- KEB parameterizations consume no more than 15% of runtime, see figure 1. Additionally 10% may be consumed by routine collecting statistics `KEB_statistics()`, depending on how many fields defined in `file_def.xml`.
- Energy dissipation is collected in `dynldf_bilap.F90` and then processed in module `KEB_module.F90`. Main routines of this module are shown in orange and main variables in green, see figure 2.



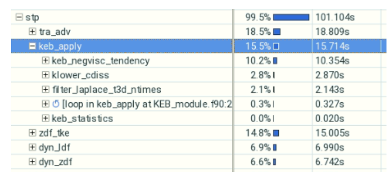
stp	99.4%	85.227s
tra_adv	22.7%	19.443s
zdf_tke	17.5%	15.026s
dyn_zdf	8.1%	6.906s
dyn_spg	7.2%	6.184s
dyn_ldf	7.2%	6.152s
tra_ldf	4.3%	3.700s
tra_zdf	4.0%	3.448s
dia_prod	3.1%	2.697s
dyn_adv	2.6%	2.217s

(a) `EXP_bare`



stp	99.5%	98.156s
tra_adv	19.2%	18.899s
zdf_tke	15.1%	14.882s
keb_apply	12.7%	12.544s
keb_ar1_tendency	10.2%	10.047s
filter_laplace_t3d_ntimes	2.1%	2.061s
[loop in keb_apply at KEB_module.f90:2	0.3%	0.296s
[loop in keb_apply at KEB_module.f90:2	0.1%	0.128s
mpp_lnk_3d	0.0%	0.012s
dyn_zdf	7.9%	7.830s
dyn_ldf	7.3%	7.184s

(b) `EXP_AR1`



stp	99.5%	101.104s
tra_adv	18.5%	18.809s
keb_apply	15.5%	15.714s
keb_negvisc_tendency	10.2%	10.354s
klower_cdiss	2.8%	2.870s
filter_laplace_t3d_ntimes	2.1%	2.143s
[loop in keb_apply at KEB_module.f90:2	0.3%	0.327s
keb_statistics	0.0%	0.020s
zdf_tke	14.8%	15.005s
dyn_ldf	6.9%	6.990s
dyn_zdf	6.6%	6.742s

(c) `EXP_negvisc`

Figure 1: Model runtime according to `Intel Advisor` for 500 steps on 1 core.

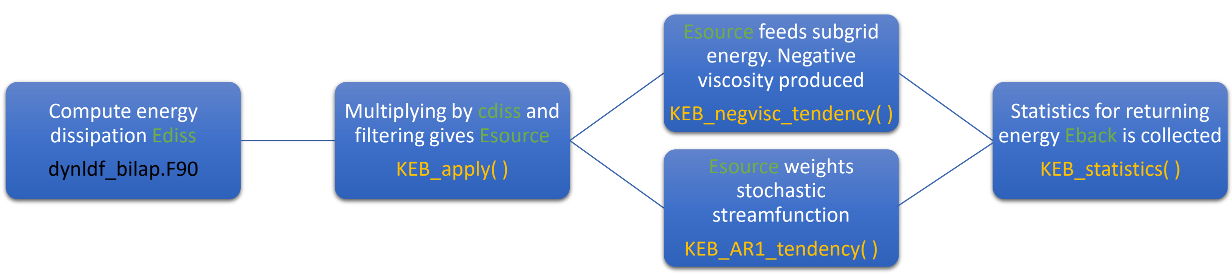


Figure 2: How KEB library works.

3 Modifications of clean NEMO code

Already existing files were modified. You can see them by checking difference between commits `Clean NEMO` and `Modifications of clean NEMO`.

Main modifications:

- `nemogcm.F90` – run `KEB_init`
- `step.F90` – run `KEB_apply` and `KEB_rst`
- `dynldf_bilap.F90` – compute `Ediss`

To reproduce my experiments (KEB can be compiled without these modifications):

- `istate.F90` – start from R9 snapshot
- `diaprod.F90, diawri.F90` – statistics
- `dom_oce.F90` – fix `integer4` sec counter
- `iom.F90, iodef.xml, field_def.xml` – fix scalars in xios

KEB library:

- `KEB_module.F90` – routines `KEB_init, KEB_rst, KEB_apply`
- `KEB_operators.F90` – all spatial operators
- `KEB_testing.F90` – test operators

4 KEB namelist

Code behaviour is set in `namelist_cfg`:

```
!-----------------------------------------------------------------------
&KEB_prms !   parameters of backscatter parameterization
!-----------------------------------------------------------------------
KEB_on      = .true.              ! turn on backscatter
KEB_test    = .false.             ! turn on testing KEB operators

! Ediss parameters
constant_cdiss = .true.           ! true = constant cdiss, false = cdiss varies
c_diss          =  0.8            ! constant backscatter rate
R_diss          =  0.5            ! non-constant backscatter rate, from Klower formula
ndiss           =  2              ! number of filters for Ediss
dirichlet_filter = .false.        ! True: zero Dirichlet, false: zero Neumann for Ediss

! negative viscosity KEB parameters
KEB_negvisc     = .false.         ! turn on negative viscosity KEB
tke_adv         = .true.          ! turn on advection of TKE
tke_diff        = .true.          ! turn on diffusion of TKE
dirichlet_TKE   = .true.          ! True: zero Dirichlet, false: zero Neumann for TKE
nu_TKE          =  300.           ! diffusivity of subgrid TKE, m2/s
tke_filter      = .false.         ! filtering TKE in z direction (for numerical stabil.)
c_back          =  0.1            ! negative viscosity ~ c_back * sqrt(TKE)
nback           =  4              ! number of filters for negvisc. tendency.

! stochastic KEB parameters
KEB_AR1     = .true.              ! turn on autoregressive in time KEB
nstoch      =  6                  ! number of filters for stochastic streamfunction
T_decorr    =  86400.             ! decorrelation time in sec, for AR1
/
```

`KEB_on` turns on backscatter, `KEB_test` is described in appendix A.

4.1 Ediss parameters

Block `Ediss parameters` describes how `Esource` is computed. General formula:

```
Esource = filter(local_cdiss * Ediss)
```

Filter is applied `ndiss` times. Its boundary conditions set by `dirichlet_filter`.

If `constant_cdiss = .true.`, then

```
local_cdiss = c_diss
```

otherwise Klower formula Klöwer et al. [2018] is used:

```
local_cdiss = 1 / (1+R_local/R_diss)
```

where `R_local` – local deformation rate scaled by Coriolis parameter (see `Klower_cdiss()` in `KEB_operators.F90`).

4.2 stochastic KEB parameters

Only number of filters for stochastic field `nstoch` and decorrelation time `T_decorr` are specified. General formula for stochastic streamfunction:

$$\mathrm{psi}(i,j,k) \sim \mathrm{filter}(\mathrm{phi}(i,j)\sqrt{\mathrm{Esource}(i,j,k)}) \tag{1}$$

where phi(i,j) – Gaussian white noise. Exact equality (1) treated similarly to dissertation O'Neill [2016].

After that time correlation introduced which does not change variance Schumann [1995]:

```
psi = psib * (1._wp - rdt / T_decorr) +
             sqrt(rdt / T_decorr * (2._wp - rdt / T_decorr)) * psi
```

Aposteriori correction averages energy input from stochastic streamfunction `Estoch` and compares it to desired energy input `Esource`. Correction of amplitude:

```
amp_increase = Esource_mean / Estoch_mean
```

Parameters of averaging are hardcoded. In `aposteriori_correction()` in `KEB_operators.F90` time window set to 30 days, initial values of `Esource_mean` and `Estoch_mean` set to `1e-9` (typical values) in `KEB_module.F90`. Min-max values of `amp_increase` are bounded by $\sqrt{2}$ and $1/\sqrt{2}$.

4.3 negative viscosity KEB parameters

Parameters `tke_adv`, `tke_diff`, `dirichlet_TKE`, `nu_TKE` set advection-diffusion of subgrid energy `TKE`, `nback` sets number of filters for momentum tendency (analogous to Juricke et al. [2020]). Negative viscosity is controlled by `c_back` as:

```
nu2t = sqrt(2) * c_back * sqrt(e12t * max(TKE,0))
```

Equation for subgrid energy reads:

```
TKE = TKE + (rhs_adv + rhs_diff + Esource - Eback) * rdt
```

Finally, `tke_filter` is vertical diffusion. Use only if you face with numerical stability issues of advection.

5 Backscatter parameters and interpretation of results

5.1 General comments

- How to set backscatter ratio `cdiss`?
  Estimate resolved energy in model with twice finer resolution and increase `cdiss` until this level will be reached in coarse model. In case of Klower formula and dirichlet B.C. for filters, resulting backscatter ratio is unclear a priori and can be found in `avr_cdiss` in output time series. If there is no additional dissipation in the model (i.e. conservative numerical schemes and no isopycnal diffusion/Gent-McWilliams), acceptable values lie in interval $(0.7, 0.8)$. Otherwise, I suppose, `cdiss=1` will be ok.

- What dissipative operators supported?
  In this code only biharmonic viscosity. Although, in NNATL12 I produced similar code for biharmonic Smagorinsky. Some estimates of dissipation are already contained in the clean NEMO code under flags like `dispkexyfo`.

- Can this code be launched with isopycnal diffusion?
  Yes, results in model R4 are even better (in North latitudes), and less restrictions on backscatter ratio `cdiss`.

- With Gent-Mcwilliams?
  I am not sure, although Jansen et al. [2019] do it, but with some modifications of subgrid energy equation.

- In some parts of the ocean, eddies are fully resolved, on the others they are fully underresolved. Where to turn off backscatter?
  Jansen et al. [2019] suggests to turn off backscatter if eddies are resolved. Also there is a belief that backscatter should amplify existing eddy filaments, so it should be turned off for underresolved simulations. In my experience, backscatter returns energy in energy-containing scale for any resolution Perezhogin and Glazunov [2021]. However, as a starting point it is reasonable to turn on backscatter only for midlatitudes (multply `local_cdiss` by some latitude function).

5.2 `negvisc` parameterization

- What default parameters correspond to?
  They are as close as possible to Juricke et al. [2020], but with advection of subgrid energy.

- How well model simulates subgrid energy?
  There is an estimate of subgrid energy for barotropic fluid Perezhogin and Glazunov [2021]:

$$\mathrm{tke\_est} = \frac{\Delta^2}{48}(|D|^2 + \omega^2), \tag{2}$$

  where $|D|$ – deformation rate, ω – vorticity, Δ – filter width of the coarse model. You can compare simulated `tke` and estimation (`tke_est`) in output files. They are different in spatial distribution, spatial scale and may be different in amount of 10 times (see time series). An amount of `tke` in statistical equilibrium is mainly controlled by parameter `c_back`. Changing it in range from `c_back = 0.4` Jansen et al. [2015] to `c_back=0.1` Juricke et al. [2020] leads to an increase of `tke` in 10 times. Also I found that in decaying turbulence an optimal value of this parameter is significantly resolution-dependent.

- Should we optimize simulation of subgrid energy?
  My beliefe is that its amount does not matter. However, its amount increases together with subgrid energy live time, and it again can range from 2 days for `c_back = 0.4` to 60 days for `c_back=0.1`. Live time is estimated as:

```
tke_time = TKE / Eback
```

and can be seen in output.

- Do we know subgrid energy live time?
  We know that subgrid energy can be generated elsewhere in the basin and be returned in intergyre jet Grooms et al. [2013]. This should take several months. Usually we can optimize either live time of subgrid energy, either its amount, but not both. My belief is that live time is more important.
- Finally, how to set `c_back`?
  Use reference values and check energy live time in output (`tke_time`). If it is about 2 days, energy returned at the same location where it is dissipated. If 60 days, it can spread over the basin.
- How to choose model for subgrid energy, `tke_adv`?
  Often it is argued that we do not know model for subgrid energy and its advection is omitted. However, we know that subgrid closure must be galilean invariant, and advection guarantees it. Galilean not-invarianse means that energy will be returned behind the mesoscale eddy path. If subgrid energy live time about 2 days, advection does not matter, if 60 days it matters. Especially, for large live time mean `Eback` is located downstream of the intergyre jet, whle `Ediss` is located near the boundary (see `materials/EXP_negvisc/result_KEB.pdf`).
- How to choose model for subgrid energy, `nu_TKE`?
  I suggest to set TKE diffusivity same as average diffusion coefficient for active tracers. An importance of this parameter mostly depends on subgrid energy live time.
- How to choose B.C. for filter (`dirichlet_filter`) and for TKE (`dirichlet_TKE`)?
  In Perezhogin [2020] both B.C. are Neumann, in Juricke et al. [2020] both are Dirichlet (private communication).

  Pros of Neumann B.C.:

  > An amount of backscattered energy is directly controlled by `c_diss`.

  Cons of Neumann B.C.:

  > Subgrid energy accumulates near boundaries, where most dissipation occurs. This accumulation is unphysical (as compared to `tke_est`).
- Does Klower formula for `c_diss` prevent from accumulation of subgrid energy near the boundary?
  In my experience – no. Dirichlet B.C. much more efficient.
- How to choose number of filters for `negvisc` tendency, `nback`?
  Following Juricke, I suggest to use this parameter mainly to prevent numerical instabilities. Improvement of the physics of backscatter with this parameter for me is unclear.
- Can subgrid energy be negative?
  In this library `Eback` and `Ediss` are positive definite. So, negative values can came only from time integration, non-positivity of advection and so on. You can check `tke_min` in the output.
- Why not to filter `Eback` as in Juricke et al. [2020]?
  This filtering leads to negative values of subgrid energy, because sink of subgrid energy spreads to regions where there is not subgrid energy.
- Why negative subgrid energy is bad?
  It can accumulate in some very unphysical locations. For example, at the second grid layer to the boundary.

5.3 AR1 parameterization

Some comments above also relevant to stochastic parameterization. Additionally:

- What default parameters correspond to?
  They are as close as possible to Perezhogin [2020], but with correlation in time.

- How to choose decorrelation time, `T_decorr`?
  This parameter should be used to filter out 1 day time scale where inertial oscillations reside. I recommend to use default value. Larger decorrelation time will lead to significant inaccuracy in formula for energy inflow
  $$\mathrm{Eback} = \mathrm{T\_decorr}\langle(\nabla^{\perp}\psi)^2\rangle \tag{3}$$
  because time scale of linear operators (like biharmonic viscosity) is smaller.

- How to choose number of filters, `nstoch`?
  In my experience, eddies arising in turbulent field are of spatial scale of stochastic forcing. So, spatial scale of stochastic forcing should be similar to eddy size (check spatial scale of stochastic forcing in file `_psi`). Unfortunately, turbulent statistics almost insensitive to `nstoch` parameter, though turbulent field may look differently. So, I recommend default value.

- Why a posteriori correction is needed?
  Formula for energy inflow (3) is correct only for dynamical system without any other RHS. Usually, a priori estimate gives insufficient energy inflow.

- When a posteriori correction work?
  It is hard coded that amplitude can be changed within a range $[1/\sqrt{2}, \sqrt{2}]$. It is easy to make a configuration when this range is insufficient: use long time scale together with small spatial scale.

- How to check that a posteriori correction works well?
  Check in output series increase in amplitude (`SKEB_amp`) and mean backscatter ratio (`avr_cdiss`). The last parameter should approach specified value of `c_diss`.

- Can we use Klower formula for `cdiss` together with `AR1` parameterization?
  It can be done in the library. But I do not recommend. An actual energy inflow from stochastic parameterization can be estimated only when there is sufficiently large number of statistical samples. Klower formula may give locally or in average large values of `cdiss` which are close to 1. This may lead to significant increase in total energy. And we cannot control it because a posteriori correction itself works on time scale of 30 days.

- Is this parameterization Galilean-invariant?
  No. And most of the stochastic backscatter parameterizations with AR1 process are not galilean-invariant, but in different extent. True invariant parameterization must include advection of stochastic field Schumann [1995]. My parameterization may return energy behind eddy path for large decorrelation time. Thats why I again recommend to use decorrelation time no more than 1 day.

- Why don't to filter only 2d stochastic field?
  Because in this case spatial scale of `Esource` will dominate in resulting parameterization. This filtering is useless after some time, and we must filter `Esource` then. So, filtering of 3D field is inevitable.

References

I. Grooms, L.-P. Nadeau, and K. S. Smith. Mesoscale eddy energy locality in an idealized ocean model. *Journal of physical oceanography*, 43(9):1911–1923, 2013.

M. F. Jansen, I. M. Held, A. Adcroft, and R. Hallberg. Energy budget-based backscatter in an eddy permitting primitive equation model. *Ocean Modelling*, 94:15–26, 2015.

M. F. Jansen, A. Adcroft, S. Khani, and H. Kong. Toward an energetically consistent, resolution aware parameterization of ocean mesoscale eddies. *Journal of Advances in Modeling Earth Systems*, 11(8):2844–2860, 2019.

S. Juricke, S. Danilov, N. Koldunov, M. Oliver, and D. Sidorenko. Ocean kinetic energy backscatter parametrization on unstructured grids: Impact on global eddy-permitting simulations. *Journal of Advances in Modeling Earth Systems*, 12(1):e2019MS001855, 2020.

M. Klöwer, M. F. Jansen, M. Claus, R. J. Greatbatch, and S. Thomsen. Energy budget-based backscatter in a shallow water model of a double gyre basin. *Ocean Modelling*, 132:1–11, 2018.

J. J. O'Neill. *A new stochastic backscatter model for large-eddy simulation of neutral atmospheric flows*. PhD thesis, University of Birmingham, 2016.

P. Perezhogin and A. Glazunov. A priori and a posteriori analysis in large eddy simulation of the two-dimensional decaying turbulence using an explicit filtering approach. *rejected JFM :)*, 2021.

P. A. Perezhogin. Testing of kinetic energy backscatter parameterizations in the nemo ocean model. *Russian Journal of Numerical Analysis and Mathematical Modelling*, 35(2):69–82, 2020.

U. Schumann. Stochastic backscatter of turbulence energy and scalar variance by random subgrid-scale fluxes. *Proceedings of the Royal Society of London. Series A: Mathematical and Physical Sciences*, 451(1941):293–318, 1995.

A Appendix. KEB_test

Before first use in new configuration, you must check that all operators works well and there are no numerical noise.

I suggest to check for numerical noise in 5d files with endings _KEB, _surf, _check_noise. Examples of numerical noise in NNATL12 configuration are given in figure 3.

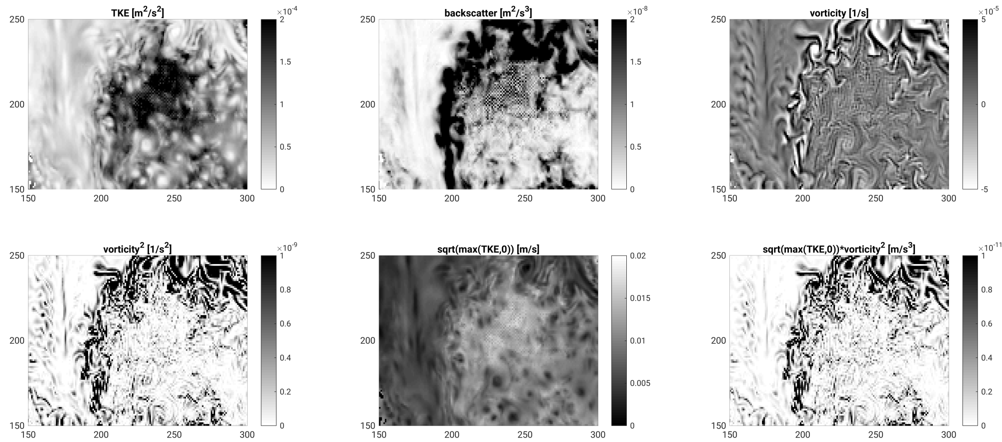


Figure 3: Numerical noise for negvisc KEB in NNATL12.

First, check that all numerics are well. Set KEB_test = .true. For negvisc there should be in ocean.output like:

```
  ~~~~~~~~~~~ KEB negvisc testing ~~~~~~~~~~~

! -------------------------- advection ------------------------ !
positiveness:
min ok:   1.850456085375350E-007  1.864908854464843E-007
max ok:   8.217884550423646E-002  8.073860079089856E-002

conservation:
no-flux            (1e-20 ok): -1.589053089423563E-020
linear free-surface (1e-8 ok):  1.929998521663570E-008

CFL:
CFL (<1 ok) =   0.234377548802137

! --------------------- Eback integration ---------------------- !
Eback ok

! -------------------------- diffusion ------------------------ !
conservation:
no-flux  (1e-20 ok): -1.064180091878790E-021
Dirichlet (1e-4 ok): -1.252268632356181E-004

positiveness:
min ok:   1.850456085375350E-007  1.856420384649364E-007
```

```
max ok:   8.217884550423646E-002  8.206124773202066E-002

! --------------------------- z-filter ------------------------ !
positiveness:
min ok:   1.850456085375350E-007  1.850865339105703E-007
max ok:   8.217884550423646E-002  8.066954583786658E-002
conservation  (1e-20 ok): -7.289300561377718E-020


! --------------------- local c_diss -------------------------- !
min value for c_diss (0 ok):  0.220658658338855
max value for c_diss (1 ok):  0.999355240226306

! -------------------- laplace filter -------------------------- !
filter_laplace_T3D_ntimes, no-flux:
conservation  (1e-20 ok): -9.063963607859050E-022
min ok:   9.955149248888630E-013  8.190910945193417E-012
max ok:   1.660738531266926E-006  9.642326610140910E-007

filter_laplace_T3D_ntimes, Dirichlet:
conservation (1e-4  ok):  1.169620783571850E-004
min ok:   9.955149248888630E-013  7.332608706254202E-012
max ok:   1.660738531266926E-006  6.331578763746706E-007

filter_laplace_f3D_ntimes, Dirichlet:
conservation (1e-4  ok):  1.754927424015464E-004
min ok:  -6.504060701236480E-005 -2.154674296598676E-005
max ok:   7.392464158314793E-005  2.079623471246499E-005

! -------  conservation of momentum for negvisc KEB  ------------ !
KEB_ldf_lap u trend (1e-20 ok): -3.694647365733425E-020
KEB_ldf_lap v trend (1e-20 ok): -3.575109685101534E-020

! -------  conservation of vorticity for negvisc KEB  ----------- !
KEB_ldf_lap (1e-5 ok):  2.933273928166026E-005

! -----------------  divergence of negvisc KEB  ----------------- !
KEB_ldf_lap (1e-26 ok):  7.425251269832589E-027

! -----------------  accuracy of Eback estimate  ---------------- !
KEB_ldf_lap, %  (1-3% ok) =    1.73893261182348

! -----------------  accuracy of Ediss estimate  ---------------- !
dynldf_bilap.F90, %    (1% ok) =   -1.41936718676091

~~~~~~~~~~~ end of KEB negvisc testing ~~~~~~~~~~~
```

For `AR1` there should be like:

```
 ~~~~~~~~~~~ KEB AR1 testing ~~~~~~~~~~~

! -----  conservation of momentum for curl of scalar field  ----- !
horizontal_curl u trend (1e-20 ok):  2.778398219086012E-019
```

```
horizontal_curl v trend (1e-20 ok): -3.462986254015147E-019

! -------------------- laplace filter --------------------------- !
filter_laplace_T3D_ntimes, no-flux:
conservation  (1e-20 ok): -3.544553728101773E-021
min ok:   5.139279367434660E-013  8.546036738813375E-012
max ok:   5.526657690610851E-006  2.128308641583762E-006

filter_laplace_T3D_ntimes, Dirichlet:
conservation (1e-4  ok):  4.496564765648987E-005
min ok:   5.139279367434660E-013  6.378930541417978E-012
max ok:   5.526657690610851E-006  2.128308641583762E-006

filter_laplace_f3D_ntimes, Dirichlet:
conservation (1e-4  ok):  1.229041937804671E-004
min ok:  -4.327175452894625E-005 -2.043950665604553E-005
max ok:   1.087831942419077E-004  1.811167495622946E-005

! ---------------------- local c_diss --------------------------- !
min value for c_diss (0 ok):  0.800000000000000
max value for c_diss (1 ok):  0.800000000000000

! -----------------  accuracy of Ediss estimate  ---------------- !
dynldf_bilap.F90, %    (1% ok) =   -1.39592065029580

~~~~~~~~~~~ end of KEB AR1 testing ~~~~~~~~~~~
```

If noise still arise with `negvisc`:

- Check CFL and positiveness of advection in `KEB_test` output. Check minimal value of TKE (`tke_min` in time series). It should not be too negative.
- Does not help? Try vertical filter `tke_filter=.true.`
- Does not help? Turn off advection `tke_adv=.false.`
- Does not help? Increase filtering of momentum tendency `nback`.
- Does not help? Reduce backscatter power by `c_diss` or `R_diss`.
- Does not help? Turn off backscatter in problem regions (for example in equatorial waveguide) by multiplying `local_cdiss` by some latitude function in `KEB_module.F90`.