

NEMO-Nordic code changes

This document describes the code changes made in the NEMO-Nordic code (local revision 339) relative to the standard NEMO code (nemo_v3.6_STABLE, revision 5628). The full subroutines are not supplied, but rather the introduced new code blocks and line numbers (in the unmodified code) in the respective subroutines. The code below shows how we introduce the fast ice mask and the spatially varying background viscosity and diffusivity, where the latter is read in from a netcdf file.

Fast ice mask

In `dom_oce.F90` at line 257 we introduced

```
REAL(wp), PUBLIC, ALLOCATABLE, SAVE, DIMENSION(:,:) :: fasticemask
```

and at lines 392–396

```
ALLOCATE( mbathy(jpi,jpj) , bathy(jpi,jpj) , &  
& tmask_i(jpi,jpj) , umask_i(jpi,jpj), vmask_i(jpi,jpj), fmask_i(jpi,jpj), &  
& bmask(jpi,jpj) , &  
& fasticemask(jpi,jpj) , &  
& mbkt (jpi,jpj) , mbku (jpi,jpj) , mbkv(jpi,jpj) , STAT=ierr(9) )
```

to define the fast ice mask.

In `domzgr.F90` at lines 986–987

```
fasticemask(:,:)=1._wp  
WHERE( bathy(:,:) <=15._wp ) fasticemask(:,:) = 0._wp
```

we calculate the fast ice mask based on the ocean depth.

In `limdyn.F90` at lines 87–88

```
u_ice_b(:,:) = u_ice(:,:) * umask(:,:,1) * fasticemask(:,:) &  
v_ice_b(:,:) = v_ice(:,:) * vmask(:,:,1) * fasticemask(:,:) &
```

and in `limrhg.F90` at lines 329–330

```
zu_ice(:,jj) = u_ice(:,jj) * fasticemask(:,jj) ! velocity at previous time step
zv_ice(:,jj) = v_ice(:,jj) * fasticemask(:,jj)
```

and lines 544–547

```
u_ice(:,:) = u_ice(:,:) * fasticemask(:,:)
v_ice(:,:) = v_ice(:,:) * fasticemask(:,:)
CALL lbc_lnk( u_ice(:,:), 'U', -1. )
CALL lbc_lnk( v_ice(:,:), 'V', -1. )
```

and lines 575–578

```
u_ice(:,:) = u_ice(:,:) * fasticemask(:,:)
v_ice(:,:) = v_ice(:,:) * fasticemask(:,:)
CALL lbc_lnk( u_ice(:,:), 'U', -1. )
CALL lbc_lnk( v_ice(:,:), 'V', -1. )
```

2

we apply the fast ice mask to the sea ice drift.

Spatially varying background viscosity/diffusivity

In `ldfdyn_c3d.h90` at lines 29–32

```
#if defined key_nordic
    USE iom
    INTEGER :: inum
#endif
```

and lines 93–111

```
#if defined key_nordic

    !! Read pre-computed background horizontal diffusion
    IF (lwp) WRITE(numout,*)
    IF (lwp) WRITE(numout,*) 'ldfdyn_c3d::3D_eddy_viscosity_coefficient'
    IF (lwp) WRITE(numout,*) '~~~~~',
    IF (lwp) WRITE(numout,*) '~~~~~VARIABLE_COEF_FOR_NORDIC'

    CALL iom_open( 'nordic.nc', inum )
    CALL iom_get ( inum, jpdom_data, 'ahm', ahm1)
    CALL iom_close (inum)

    CALL lbc_lnk( ahm1, 'T', 1. )    ! Lateral boundary conditions

    ahm2(:,:,:) = ahm1(:,:,:)
    ahm3(:,:,:) = ahm1(:,:,:)
    ahm4(:,:,:) = ahm1(:,:,:)

#endif
```

we read in a spatially variable background viscosity from a file called `nordic.nc`.

In `ldftrad_c3d.h90` at lines 32–35

```
#if defined key_nordic
    USE iom
    INTEGER :: inum
```

```
#endif
```

and lines 76–98

```
#if defined key_nordic

    !! Read pre-computed background horizontal diffusion

    IF(lwp) WRITE(numout,*)
    IF(lwp) WRITE(numout,*) '_ldf_tra_c3d:_3D_eddy_diffusivity_coefficient'
    IF(lwp) WRITE(numout,*) '-----',
    IF(lwp) WRITE(numout,*) '-----VARIABLE_COEF_FOR_NORDIC'

    CALL iom_open( 'nordic.nc', inum )
    CALL iom_get ( inum, jpdom_data, 'aht', ahtt)
    CALL iom_close (inum)

    ahtu(:,:,:) = ahtt(:,:,:)
    ahtv(:,:,:) = ahtt(:,:,:)
    ahtw(:,:,:) = ahtt(:,:,:)

    CALL lbc_lnk( ahtt, 'T', 1. )    ! Lateral boundary conditions
    CALL lbc_lnk( ahtu, 'U', 1. )
    CALL lbc_lnk( ahtv, 'V', 1. )
    CALL lbc_lnk( ahtw, 'W', 1. )

#endif
```

we read in a spatially variable background diffusivity from a file called `nordic.nc`.