Supporting Information

for

Simulations of water transport over the Iceland–Scotland Ridge improved by local grid refinement around the Faroe Islands

Andrea M. U. Gierisch, Bogi Hansen, Karin M. H. Larsen, Steffen M. Olsen

S1: Code changes in NEMO-AGRIF

The model simulations are conducted with NEMO version 4.0.4, revision r13658 (https://forge.ipsl.jussieu.fr/nemo/svn/NEMO/releases/r4.0/r4.0.4/, downloaded on 2020-10-21).
A few changes to the original code were required in order to run the model with non-linear free surface and the AGRIF nest. These changes concern the files `agrif_oce_update.F90`, `agrif_top_update.F90` and `agrif_user.F90`, and they are shown below as "`svn diff`" comparison of the adapted model code ("working copy") and the original revision r13658.

```
===================================================================
Index: NST/agrif_oce_update.F90
===================================================================
--- NST/agrif_oce_update.F90 (revision 13658)
+++ NST/agrif_oce_update.F90 (working copy)
@@ -412,7 +412,8 @@
                DO jj=j1,j2
                   DO ji=i1,i2
 !> jc tmp
-                     tabres(ji,jj,jk,jn) = tsn(ji,jj,jk,jn)  * e3t_n(ji,jj,jk) / e3t_0(ji,jj,jk)
+                     tabres(ji,jj,jk,jn) = tsn(ji,jj,jk,jn)
 !                     tabres(ji,jj,jk,jn) = tsn(ji,jj,jk,jn)  * e3t_n(ji,jj,jk)
 !< jc tmp
                   END DO
@@ -422,7 +423,9 @@
       ELSE
 !> jc tmp
         DO jn = 1,jpts
-           tabres(i1:i2,j1:j2,k1:k2,jn) =  tabres(i1:i2,j1:j2,k1:k2,jn) * e3t_0(i1:i2,j1:j2,k1:k2) &
+           tabres(i1:i2,j1:j2,k1:k2,jn) =  tabres(i1:i2,j1:j2,k1:k2,jn)  &
+                                        & * tmask(i1:i2,j1:j2,k1:k2)
         ENDDO
 !< jc tmp
@@ -433,11 +436,16 @@
                   DO jj = j1, j2
                      DO ji = i1, i2
                         IF( tabres(ji,jj,jk,jn) /= 0._wp ) THEN
-                           ztb  = tsb(ji,jj,jk,jn) * e3t_b(ji,jj,jk) ! fse3t_b prior update should be used
+                           ztb  = tsb(ji,jj,jk,jn)
+                           ztnu = tabres(ji,jj,jk,jn)
-                           ztno = tsn(ji,jj,jk,jn) * e3t_a(ji,jj,jk)
+                           ztno = tsn(ji,jj,jk,jn)
```

```
                           tsb(ji,jj,jk,jn) = ( ztb + atfp * ( ztnu - ztno) )  &
-                                    &        * tmask(ji,jj,jk) / e3t_b(ji,jj,jk)
+                                    &        * tmask(ji,jj,jk)
                         ENDIF
                      END DO
                   END DO
@@ -449,7 +457,8 @@
                DO jj=j1,j2
                   DO ji=i1,i2
                      IF( tabres(ji,jj,jk,jn) /= 0._wp ) THEN
-                        tsn(ji,jj,jk,jn) = tabres(ji,jj,jk,jn) / e3t_n(ji,jj,jk)
+                        tsn(ji,jj,jk,jn) = tabres(ji,jj,jk,jn)
                      END IF
                   END DO
                END DO
@@ -578,7 +587,8 @@
       IF( before ) THEN
          zrhoy = Agrif_Rhoy()
          DO jk = k1, k2
-            tabres(i1:i2,j1:j2,jk,1) = zrhoy * e2u(i1:i2,j1:j2) * e3u_n(i1:i2,j1:j2,jk) * un(i1:i2,j1:j2,jk)
+            tabres(i1:i2,j1:j2,jk,1) = zrhoy * e2u(i1:i2,j1:j2) * un(i1:i2,j1:j2,jk)
          END DO
       ELSE
          DO jk=k1,k2
@@ -587,14 +597,20 @@
                   tabres(ji,jj,jk,1) = tabres(ji,jj,jk,1) * r1_e2u(ji,jj)
                   !
                   IF (.NOT.(lk_agrif_fstep.AND.(neuler==0))) THEN ! Add asselin part
-                     zub  = ub(ji,jj,jk) * e3u_b(ji,jj,jk)  ! fse3t_b prior update should be used
-                     zuno = un(ji,jj,jk) * e3u_a(ji,jj,jk)
+                     zub  = ub(ji,jj,jk)  ! fse3t_b prior update should be used
+                     zuno = un(ji,jj,jk)
                      zunu = tabres(ji,jj,jk,1)
                      ub(ji,jj,jk) = ( zub + atfp * ( zunu - zuno) ) &
-                                  & * umask(ji,jj,jk) / e3u_b(ji,jj,jk)
+                                  & * umask(ji,jj,jk)
                   ENDIF
                   !
-                  un(ji,jj,jk) = tabres(ji,jj,jk,1) * umask(ji,jj,jk) / e3u_n(ji,jj,jk)
+                  un(ji,jj,jk) = tabres(ji,jj,jk,1) * umask(ji,jj,jk)
                END DO
             END DO
          END DO
@@ -766,7 +782,8 @@
          DO jk=k1,k2
            DO jj=j1,j2
              DO ji=i1,i2
-               tabres(ji,jj,jk,1) = zrhox * e1v(ji,jj) * e3v_n(ji,jj,jk) * vn(ji,jj,jk)
+               tabres(ji,jj,jk,1) = zrhox * e1v(ji,jj) * vn(ji,jj,jk)
              END DO
            END DO
          END DO
@@ -777,14 +794,20 @@
                   tabres(ji,jj,jk,1) = tabres(ji,jj,jk,1) * r1_e1v(ji,jj)
                   !
                   IF (.NOT.(lk_agrif_fstep.AND.(neuler==0))) THEN ! Add asselin part
-                     zvb  = vb(ji,jj,jk) * e3v_b(ji,jj,jk) ! fse3t_b prior update should be used
-                     zvno = vn(ji,jj,jk) * e3v_a(ji,jj,jk)
+                     zvb  = vb(ji,jj,jk)  ! fse3t_b prior update should be used
+                     zvno = vn(ji,jj,jk)
                      zvnu = tabres(ji,jj,jk,1)
```

```
                     vb(ji,jj,jk) = ( zvb + atfp * ( zvnu - zvno) ) &
-                                   & * vmask(ji,jj,jk) / e3v_b(ji,jj,jk)
+                                   & * vmask(ji,jj,jk)
                   ENDIF
                   !
-                  vn(ji,jj,jk) = tabres(ji,jj,jk,1) * vmask(ji,jj,jk) / e3v_n(ji,jj,jk)
+                  vn(ji,jj,jk) = tabres(ji,jj,jk,1) * vmask(ji,jj,jk)
               END DO
            END DO
         END DO

==================================================================
Index: NST/agrif_top_update.F90
==================================================================
--- NST/agrif_top_update.F90 (revision 13658)
+++ NST/agrif_top_update.F90 (working copy)
@@ -181,10 +181,11 @@
             DO jk=k1,k2
                DO jj=j1,j2
                   DO ji=i1,i2
-                     tabres(ji,jj,jk,jn) = trn(ji,jj,jk,jn)  * e3t_n(ji,jj,jk) / e3t_0(ji,jj,jk)
+                     tabres(ji,jj,jk,jn) = trn(ji,jj,jk,jn)
                   END DO
                END DO
             END DO
@@ -192,7 +193,9 @@
       ELSE
 !> jc tmp
         DO jn = n1,n2
-           tabres(i1:i2,j1:j2,k1:k2,jn) =  tabres(i1:i2,j1:j2,k1:k2,jn) * e3t_0(i1:i2,j1:j2,k1:k2) &
+           tabres(i1:i2,j1:j2,k1:k2,jn) =  tabres(i1:i2,j1:j2,k1:k2,jn)  &
+                                        & * tmask(i1:i2,j1:j2,k1:k2)
         ENDDO
 !< jc tmp
@@ -203,11 +206,15 @@
                   DO jj=j1,j2
                      DO ji=i1,i2
                         IF( tabres(ji,jj,jk,jn) .NE. 0. ) THEN
-                           ztb  = trb(ji,jj,jk,jn) * e3t_b(ji,jj,jk) ! fse3t_b prior update should be used
+                           ztb  = trb(ji,jj,jk,jn)
+                           ztnu = tabres(ji,jj,jk,jn)
-                           ztno = trn(ji,jj,jk,jn) * e3t_a(ji,jj,jk)
+                           ztno = trn(ji,jj,jk,jn)
+                           trb(ji,jj,jk,jn) = ( ztb + atfp * ( ztnu - ztno) )  &
-                                            &        * tmask(ji,jj,jk) / e3t_b(ji,jj,jk)
+                                            &        * tmask(ji,jj,jk)
                        ENDIF
                     ENDDO
                  ENDDO
@@ -219,7 +226,8 @@
                DO jj=j1,j2
                   DO ji=i1,i2
                      IF( tabres(ji,jj,jk,jn) .NE. 0. ) THEN
-                        trn(ji,jj,jk,jn) = tabres(ji,jj,jk,jn) / e3t_n(ji,jj,jk)
+                        trn(ji,jj,jk,jn) = tabres(ji,jj,jk,jn)
                      END IF
                   END DO
                END DO
```

```
===================================================================
Index: NST/agrif_user.F90
===================================================================
--- NST/agrif_user.F90 (revision 13658)
+++ NST/agrif_user.F90 (working copy)
@@ -657,8 +657,10 @@
    CALL agrif_declare_variable((/2,2,0,0/),(/ind3,ind3,0,0/),(/'x','y','N','N'/),(/1,1,1,1/),(/nlci,nlcj,jpk,jptra+1/),trn_id)
    CALL agrif_declare_variable((/2,2,0,0/),(/ind3,ind3,0,0/),(/'x','y','N','N'/),(/1,1,1,1/),(/nlci,nlcj,jpk,jptra+1/),trn_sponge_id)
 # else
-   CALL agrif_declare_variable((/2,2,0,0/),(/3,3,0,0/),(/'x','y','N','N'/),(/1,1,1,1/),(/nlci,nlcj,jpk,jptra/),trn_id)
-   CALL agrif_declare_variable((/2,2,0,0/),(/3,3,0,0/),(/'x','y','N','N'/),(/1,1,1,1/),(/nlci,nlcj,jpk,jptra/),trn_sponge_id)
+   CALL agrif_declare_variable((/2,2,0,0/),(/ind3,ind3,0,0/),(/'x','y','N','N'/),(/1,1,1,1/),(/nlci,nlcj,jpk,jptra/),trn_id)
+   CALL agrif_declare_variable((/2,2,0,0/),(/ind3,ind3,0,0/),(/'x','y','N','N'/),(/1,1,1,1/),(/nlci,nlcj,jpk,jptra/),trn_sponge_id)
 # endif

    ! 2. Type of interpolation
```