

Notice pour forcer en flux l'océan seul (sans la glace) avec NEMO3.6

Yona Silvy - Janvier 2020

Objectif : forcer l'océan à partir des flux à la surface de l'océan liquide (sortant du même modèle d'océan). Réaliser une simulation de contrôle (CTL) en appliquant exactement ces flux, dont le but est de répliquer la simulation de référence qui sort les flux. Réaliser également des simulations rajoutant une perturbation anthropique à ces flux.

Simulations

Force the ocean with outputs from:

- IPSLCM6 pre-industrial control, 251 years, 3hourly fluxes at the ocean surface Q, F, τ
→ has to be re-run because we don't have the 3h fluxes
- IPSLCM6 historical+ssp ensemble mean, 1850-2100, monthly fluxes perturbation Q', F', τ'

Monthly flux perturbation from hist+ssp ensemble mean
3hourly flux from piControl
All runs are 251 years long, to simulate 1850-2100

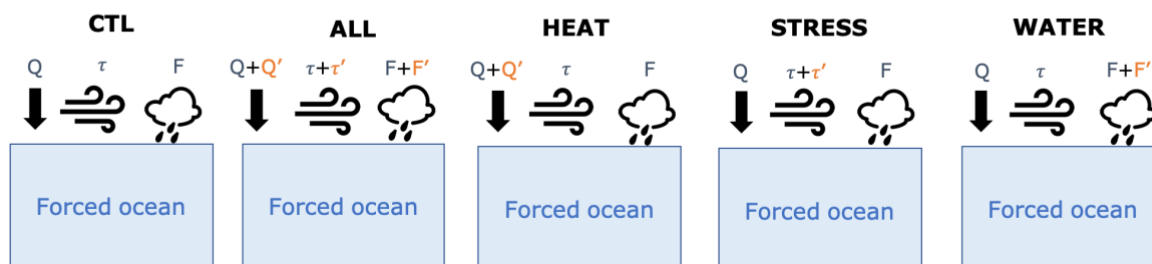


Schéma très simplifié de mes simulations forcées

Les flux utiles

Tous les flux sortent du modèle d'océan donc sur la grille océanique, au bord de l'océan liquide (=sous la glace dans les zones englacées).

Pour plus de précisions sur les budgets dans NEMO et les variables associées, voir le doc de Clément « budgets-nemo.pdf ».

Le schéma et le tableau ci-dessous listent les variables utilisées pour forcer le modèle, ainsi que les routines associées.

Nom variable dans le code	Nom variable en sortie	Unité	Description	Routine dans laquelle elle est lue/utilisée, ou commentaire
Chaleur				
qt (=qsr+qns)	qt	W/m2	Net Downward Heat Flux	sbcflx.F90
qsr	qsr	W/m2	surface_net_downward_shortwave_flux	sbcflx.F90
qns	qns	W/m2	surface_net_downward_non_solar_flux	Pas lue mais déduite de qt-qsr dans sbcflx.F90
qisf (latent) + contribution sensible	hflx_isf(_cea)	W/m2	heat_flux_into_sea_water_due_to_iceshelf	Reconstruit dans sbcisf.F90 avec nn_isf=3 ou lu avec nn_isf=5 (on a opté pour nn_isf=3). Latent + sensible
/	hflx_rnf(_cea)	W/m2	temperature_flux_due_to_runoff_expressed_as_heat_flux_into_sea_water	sbc_rnf.F90; flux de chaleur sensible
rau0_rcp * qgh_trd0(:,,:)	hfgeou	W/m2	upward_geothermal_heat_flux_at_sea_floor_heat_flux	Aucun changement (trabbc.F90), juste pour info car important dans budget de chaleur
Flux de chaleur entrant total = qt + hflx_isf + hflx_rnf + hfgeou (voir doc budgets Clem)				
Eau douce				
emp	emp	kg/m2/s		sbcflx.F90
rnf	runoffs	kg/m2/s	water_flux_into_sea_water_from_runoffs (rivers+icebergs)	sbc_rnf.F90
empmr	wfo (empmr)	kg/m2/s	water_flux_into_sea_water (l'opposé en fait : upward)	empmr = emp - runoffs
friver	friver	kg/m2/s	water_flux_into_sea_water_from_rivers	runoffs = friver + ficeberg
ficeberg	iceberg(_cea)	kg/m2/s	water_flux_into_sea_water_from_icebergs	On peut aussi lire ficeberg séparément pour pouvoir le sortir. Sinon juste besoin de runoffs
fwfisf	-iceshelf(_cea)	kg/m2/s	water_flux_into_sea_water_from_iceshelf	sbcisf.F90; attention signe opposé entre variable code et variable de sortie
Flux total d'eau douce = empmr - iceshelf				
Sel				
sfx	sosflxdo (saltflx)	g/m2/s	Downward salt flux	sbcflx.F90 (attention il y a une erreur d'unité dans field_def_nemo-opa.xml, ce ne sont pas des kg/m2/s mais des g/m2/s)
Vent				
utau	tauuo	N/m2	surface_downward_x_stress	sbcflx.F90
vtau	tauvo	N/m2	surface_downward_y_stress	sbcflx.F90
taum	taum	N/m2	wind stress module (magnitude_of_surface_downward_stress)	Eventuellement à lire dans sbcflx.F90, sinon il est reconstruit à partir de utau et vtau
Glace				
fr_i	siconc (LIM3)		sea ice fraction	Lue dans sbcflx.F90 et utilisée dans zdf_tke.F90
hcpl_i	sithic (LIM3)	m	sea ice thickness	Lue dans sbcflx.F90 et utilisée dans zdf_tke.F90
vcpl_i	sivolu (LIM3)	m	sea ice volume per area	Lue dans sbcflx.F90 et utilisée dans zdf_tke.F90 (en fait besoin que de hcpl_i ou vcpl_i au choix, en plus de fr_i)
Chloro				
chl	nchl + dchl (PISCES)	mg/m3	Total chlorophyl Concentration	traqsr.F90 (si PISCES désactivé)

Liste des variables pour forcer l'océan (ou reliées)

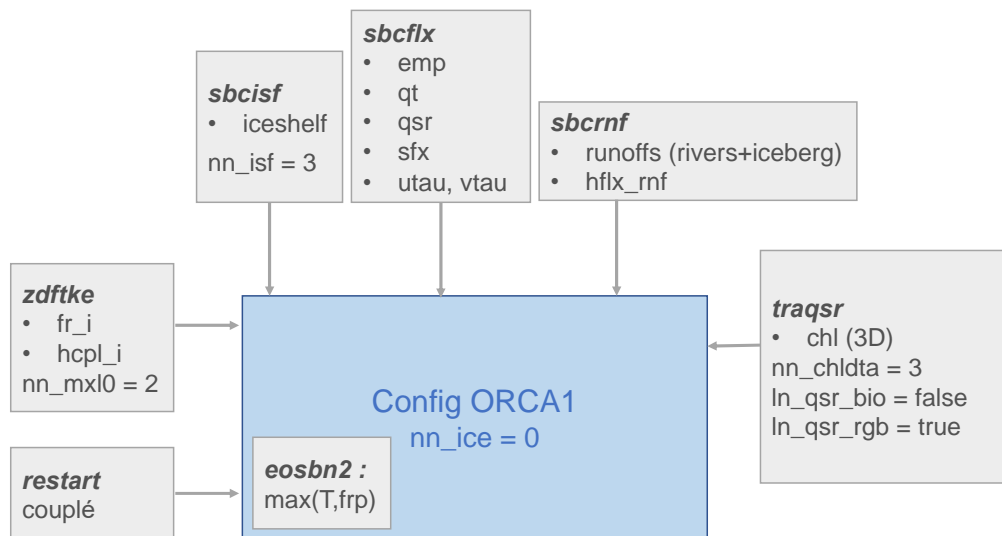


Schéma de forçage (sans PISCES sans LIM3)

Les anomalies

Dans le cas où l'on veut imposer des anomalies de flux par-dessus les flux précédents (par exemple venant des historiques + scenario CMIP6, c'est notre cas avec IPSLCM6), on les met dans des fichiers séparés à la fréquence que l'on veut, et on a rajouté une routine `sbcflx_ano.F90` pour les lire, routine qui est ensuite appelée par `sbcflx.F90`, `sbcrrf.F90` et `sbcisf.F90` pour sommer les anomalies aux flux de base.

Correspondance nomenclature format standard NEMO – format CMIP6

Toutes les informations sont vérifiables dans le `ping_nemo.xml` (donne la façon dont les sorties CMIP6 sont construites à partir des formats standard).

Pour les simulations avec les anomalies, on lit les mêmes variables de glace que pour la simulation CTL et non des variables qui viendraient de CMIP6 car cela n'a pas de sens pour les simulations de sensibilité (ex : en ne mettant que les anomalies de vent, cela n'est pas physique de lire la concentration de glace qui provient de simulations historiques qui ont par ailleurs aussi vu un changement des flux de chaleur et d'eau douce). Les variables de glace dans la simulation CTL ne sont utilisées que pour paramétrer le mélange sous la glace (dans `zdfcke.F90`). On garde donc la même paramétrisation dans les simulations avec anomalies que dans celle de contrôle (sans anomalies).

Nom sortie NEMO	Nom CMIP6	Unité	Commentaire
Chaleur			
qt	hfds	W/m2	Attention, ici questionnement car dans Griffies et al. 2016 il est écrit que hfrunoffds est compris dans hfds, alors que ce ne serait pas le cas pour qt hfds = hfrainds + hfevapds + hfrunoffds + rlntds + hfls + hfss + rsntds + hfsifrazil
qsr	rsntds	W/m2	
qns	nshfls	W/m2	
hflx_isf	Ne sort pas dans CMIP6	W/m2	Reconstructible = -iceshelf_cea * (rLfus(isf) - Tfreeze * cp)
hflx_rnf	hfrunoffds	W/m2	
Eau douce			
wfo=empmr	wfo	kg/m2/s	En entrée du modèle il faut séparer emp des runoffs. Attention wfo s'appelle "water flux into sea water" mais est en fait égal à empmr i.e. out of sea water
runoffs	Ne sort pas dans CMIP6	kg/m2/s	Reconstruit simplement en sommant friver et ficeberg
friver	friver	kg/m2/s	
iceberg	ficeberg	kg/m2/s	
iceshelf	flandice	kg/m2/s	Attention à chercher dans Emon (et non Omon)
Sel			
sosflxdo	sfdsi/convSpsu	g/m2/s	Attention a la conversion salinité absolue (NEMO) / salinité pratique (CMIP6). Pratique = absolue * convSpsu. D'autre part, il y a surement une erreur d'unité dans les sorties IPSLCM6 comme dans les sorties standard (les sorties indiquent des kg/m2/s mais a priori ce sont bien des g/m2/s qui sortent)
Vent			
tauuo	tauuo	N/m2	
tauvo	tauvo	N/m2	

Correspondance sorties NEMO – sorties CMIP6 pour les variables utilisées

Liste des routines à modifier

Toutes les modifications (sauf oubli) sont annotées d'un commentaire !Yona.

sbcflx.F90

C'est la routine principale qui lit les flux.

Sont lus : qt, qsr, emp, sfx, utau, vtau, sithic, sivol, siconc

A la fin de la routine, si les anomalies sont activées, on appelle sbc_flux_ano et on additionne les anomalies aux flux ci-dessus.

Taum est reconstruit à part de utau et vtau, cependant on peut aussi choisir de le lire directement (voir sbcflx.F90_readtaum).

Pour la version où on lit taum, les lignes modifiées sont annotées d'un commentaire !Yona taum

[sbcflx_ano.F90 \(routine créée\)](#)

C'est la routine qui lit les anomalies. Dans mon cas, les anomalies sont toujours lues car elles servent à forcer mes traceurs passifs. Cependant elles sont additionnées aux flux lus dans sbcflx seulement si les booléens ln_heat_ano et/ou ln_fwf_ano et/ou ln_stress_ano sont à true (voir explications dans la namelist).

Dans cette routine, sont lues les anomalies de : qt, qsr, emp, sfx, utau, vtau, hflx_rnf (hfrnf), iceshelf (fwisf), runoffs (fwrnf).

Le flux de chaleur latent venant de l'iceshelf est reconstruit $hfisf_ano(:, :) = fwisf_ano(:, :) * lfusisf$, puis la partie sensible est ajoutée dans sbcisf.F90.

On avait créé un booléen ln_dm2dc_ano pour appliquer un cycle diurne à l'anomalie de qsr séparément des flux de base, cependant on ne peut appliquer le cycle diurne (routine sbc_dcy) seulement sur des fichiers journaliers donc ça ne s'applique pas à notre cas. J'ai laissé le code avec cet ajout mais il ne sert à rien.

[sbcrrnf.F90](#)

C'est la routine qui traite les runoffs.

On y lit les runoffs (le flux d'eau douce) ainsi que le flux de chaleur associé (au lieu de la température ce qui est initialement prévu par le code).

Si les anomalies sont activées, elles sont additionnées (eau douce et chaleur).

[sbcisf.F90](#)

C'est la routine qui traite les conditions de surface sous l'iceshelf.

On a choisi l'option nn_isf=3, donc la routine lit le flux d'eau douce (fwfisf) et calcul le flux de chaleur $qisf = fwfisf * lfusisf$ (latent) puis en rajoutant la partie sensible flux de chaleur total = $qisf(:, :) - rdivisf * fwfisf(:, :) * zt_frz * rcp = fwfisf (lfusisf - rdivisf * zt_frz * rcp)$

Si les anomalies sont activées, elles sont additionnées (eau douce et chaleur).

Originellement on avait ajouté un cas nn_isf=5 mais on ne l'utilise pas.

[eosbn2.F90](#)

Gère l'équation d'état de l'eau de mer.

On a décidé de faire comme la méthode des simulations ocean-only FAFMIP (Todd et al. 2020), c'est-à-dire que dans les simulations forcées (et en particulier avec les anomalies), il se peut que la température sous la glace de mer descende en-dessous du point de congélation (~2°C). Si c'est le cas, on laisse la température à la valeur qu'elle atteint (même si c'est < freezing point) pour conserver la chaleur, cependant pour calculer la densité on utilise le freezing point (c'est-à-dire on prend $\max(T, T_{freeze})$, afin de garder une densité physique et ne pas créer trop de convection par exemple.

Toutes les modifications dans cette routine sont marquées de commentaires !!clem

[zdftke.F90](#)

Gère la physique verticale. En particulier le mélange sous la glace dépend de la concentration/épaisseur de glace (on a nn_mx10=2). Comme on n'active pas le modèle de glace dans notre cas, on lit ces valeurs en sortie de notre simulation de référence qui sert à extraire tous les flux (modèle couplé pour moi, mais ça peut très bien être NEMO avec la glace), afin

d'avoir un mélange vertical cohérent avec la simulation de référence. Les variables de glace (fraction, épaisseur, volume par surface) sont lues dans sbcflx.F90 et utilisées ici.

Les modifications dans cette routine sont marquées de commentaires !!clem (dans la subroutine tke_avn).

On a besoin d'avoir lu dans sbcflx fr_i (siconc, la concentration de glace), et vcpl_i (sivol, le volume/area) ou bien hcpl_i (sithic, l'épaisseur de glace). J'ai laissé les 3 mais 2 suffisent.

[sbcmod.F90](#)

De simples ajouts de paramètres de namelist concernant les anomalies (voir les deux lignes commentées !Yona). Il me semble que ce sont les seuls ajouts.

[sbc_oce.F90](#)

Idem + déclaration du volume et épaisseur de la glace.

[traqsr.F90](#)

C'est la routine qui gère la pénétration du flux solaire dans les premiers niveaux de l'océan. Quand PISCES est activé il n'y a rien à modifier (ln_qsr_bio=true & ln_qsr_rgb=false). Dans mon cas PISCES est désactivé (on a ln_qsr_bio=false & ln_qsr_rgb=true) et on lit un champ de chlorophylle. On a remarqué que pour être le plus proche possible de la simulation de référence, le choix du champ de chlorophylle utilisé était très important. On se rapproche donc le plus possible de la simulation de référence en lisant le champ 3D de chlorophylle moyenné sur 5 jours qui sort de ma simulation couplée.

Pour cela on a du rajouter un cas de figure (nn_chldta=3) dans traqsr.F90 afin de lire un fichier 3D.

On a également remarqué que dans la version d'océan utilisée dans mon run couplé de référence, on lit une climatologie « par_fraction ». On la lit donc ici pour utiliser le coefficient fr_par dans le calcul de la pénétration du flux solaire (voir modifs lignes 241→255 + 309→318). On a également modifié un seuil pour qu'il corresponde avec celui utilisé dans le modèle de bio (0.05 au lieu de 0.03, ligne 263, 299).

Toutes les modifications sont a priori commentées.

[Namelist_cfg et spécifications de forçage](#)

Ma namelist cfg s'appelle « namelist_flx_ia_cfg ».

Les paramètres modifiés/importants pour le forçage en flux se trouvent essentiellement dans les sections namsbc, namsbc_flx, namsbc_flx_ano, namtra_qsr, namsbc_rnf, namsbc_isf.

Les seuls paramètres à changer pour les différentes simulations sont ceux concernant l'activation des anomalies (ln_heat_ano, ln_fwf_ano et ln_stress_ano, voir namsbc_flx_ano).

[namsbc_flx](#)

C'est là où on indique les fichiers et variables des flux utau, vtau, qt, qsr, emp et sfx ainsi que les variables de glace (lues dans sbcflx.F90). Pour chacune des variables, il faut indiquer le nom du fichier, sachant que tous les fichiers de forçages doivent être séparés par année et finir par yearXX.nc, par exemple mes fichiers s'appellent flux_1850.nc, flux_1851.nc etc.. donc sous la colonne « file name » je renseigne simplement flux (le modèle se débrouille pour le suffixe). Il faut ensuite indiquer la fréquence de chaque variable (ici 3h pour mes flux donc simplement 3),

le nom de la variable dans le fichier lu, si on veut que le modèle interpole temporellement à son pas de temps (pour les flux à 3h on ne veut pas d'interpolation car c'est exactement ce qui est reçu de l'atmosphère dans le modèle couplé), s'il s'agit d'une climatologie (ici non), si les fichiers sont séparés par mois ou par an (ici 'yearly').

[namsbc_flx_ano](#)

C'est là qu'on indique les fichiers de forçages pour toutes les anomalies (lues dans sbcflx_ano.F90). Ici on lit des moyennes mensuelles, donc on active l'interpolation temporelle. On définit également les paramètres qui permettent d'activer les anomalies :

- `ln_heat_ano` si on veut ajouter les anomalies de flux de chaleur
- `ln_fwf_ano` si on veut ajouter celles de flux d'eau douce
- `ln_stress_ano` si on veut ajouter celles de tension du vent

Ainsi, dans CTL ils sont tous à false, dans ALL tous à true, et dans les simulations de sensibilité on décide d'en activer un à la fois (mais on pourrait aussi en mettre 2).

[namtra_qsr](#)

C'est là où on traite les informations relativement à la pénétration du flux solaire (lues dans traqsr.F90).

On y lit la chlorophylle (dans le cas où l'on désactive PISCES) ainsi que `par_fraction`.

Les paramètres sont les suivants :

- `ln_qsr_rgb` = true
- `ln_qsr_bio` = false
- `nn_chldta` = 3 (rajout d'un cas 3 afin de lire le champ de chlorophylle 3D)
- `ln_qsr_ice` = false

[namsbc_rnf](#)

Informations concernant les runoffs (lues dans sbcrnf.F90).

On vient lire les runoffs dans `sn_rnf` et le flux de chaleur associé (`hflx_rnf`) dans `sn_t_rnf`. La profondeur à laquelle sont répartis les runoffs est également lue dans un fichier mais ce n'est pas nécessaire de le modifier.

Tous les paramètres sont également à vérifier.

[namsbc_isf](#)

Informations concernant les conditions de surface sous l'iceshelf, lues dans sbcisf.F90.

On lit le flux d'eau douce « iceshelf » dans `sn_rnfisf` pour le cas `nn_isf=3` (`sn_qisf` et `sn_fwfisf` sont appelés dans sbcisf.F90 mais ne servent pas dans le cas `nn_isf=3`).

Traceurs passifs

Pour plus tard cette section

Divers

Bien faire attention au fait que la salinité et température dans NEMO3.6 sont la salinité absolue et la température conservative (TEOS-10 ; `nn_eos=-1`). Les variables de sorties s'appellent `so` et `thetao`.

Cependant, dans les sorties CMIP6 la variable 'thetao' est la température potentielle, et la variable 'bigthetao' la température conservative.

Pour la salinité, 'so' dans CMIP6 est la salinité pratique, et a été convertie depuis la salinité absolue du modèle en divisant par le facteur $\text{convSpsu} = 0.99530670233846$ (voir le ping_nemo.xml).