Interpolations à partir de fichiers de poids et d'adresses avec OASIS4

Développé par J. Latour et S. Valcke

CLUC du 20/10/2011

Laure Coquart

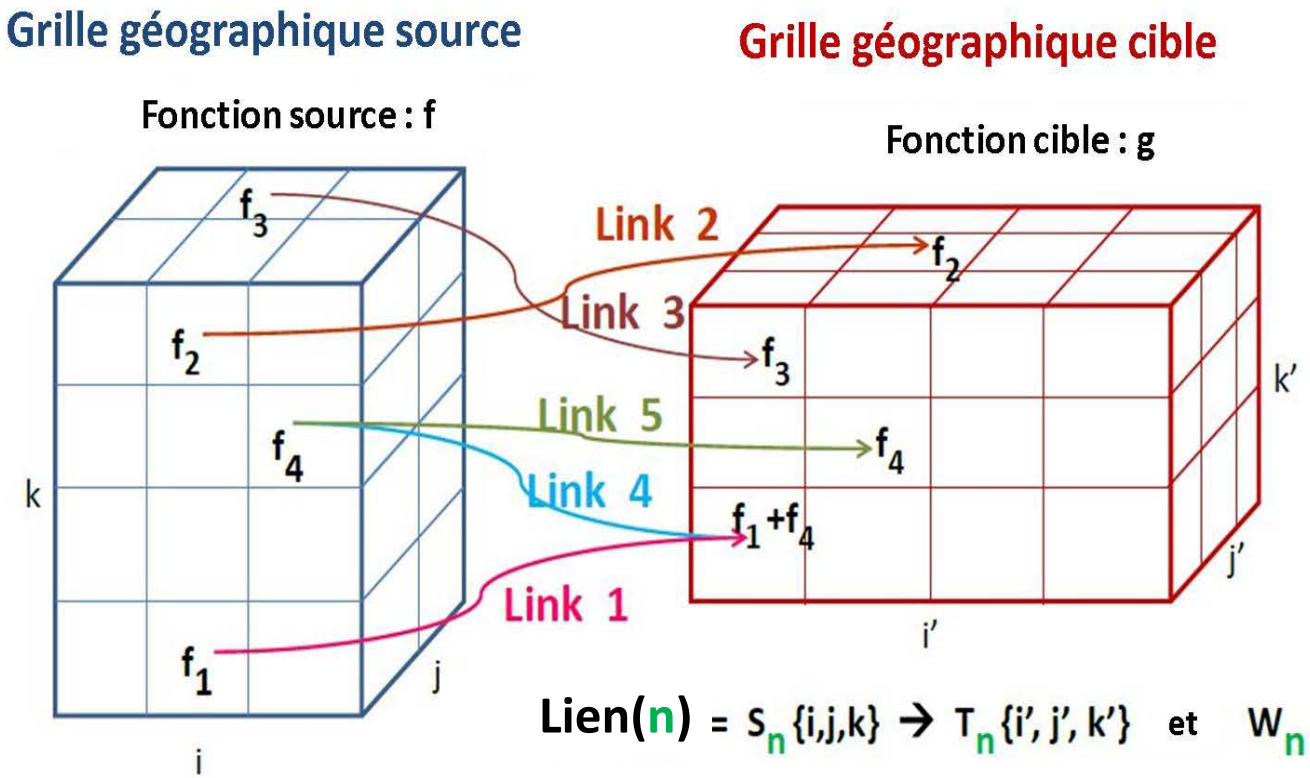
- Fonctionnement

- Tests de validation

$g(i',j',k') = \Sigma\{ Wn * f(i,j,k)\}$

La librairie de recherche globale parallèle d'Oasis4 pour le calcul « en ligne » des poids **Wn** n'est pas stable : encore des différences mono-para

=> Utilisation d'un fichier de poids et adresses pré-calculé contenant les poids **Wn** et les adresses (i,j,k) et (i',j'k') (« **USERDEF** »)



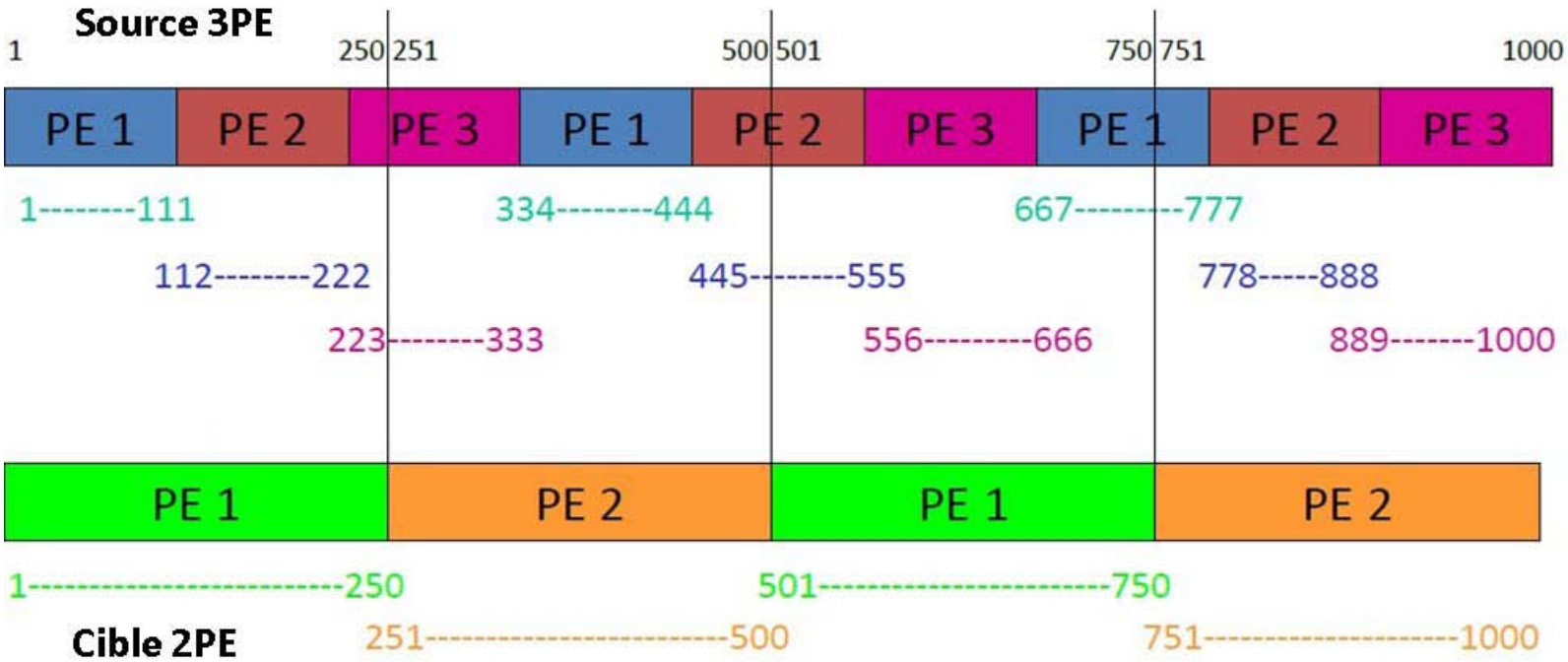
Fichier de poids et d'adresses

n	i,j,k	i',j',k'	W
1	2,1,1	1,1,1	0.8
2	2,1,4	2,2,3	1.0
3	2,2,4	1,1,3	1.0
4	3,1,3	1,1,1	0.2
5	3,1,3	2,1,2	0.8

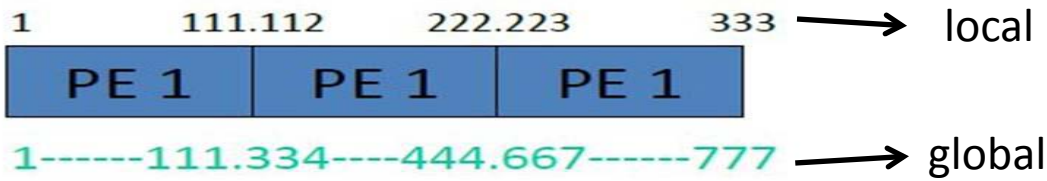
Fonctionnement du « Userdef »

➢Lecture du fichier (1 par champ de couplage) par la source et la cible: transformation des indices 3D en indice 1D sur une grille « gridless » multiblocs (ensembles discontinus de liens continus)

=> autant de points que de liens



Gridless sur le PE1 de la source

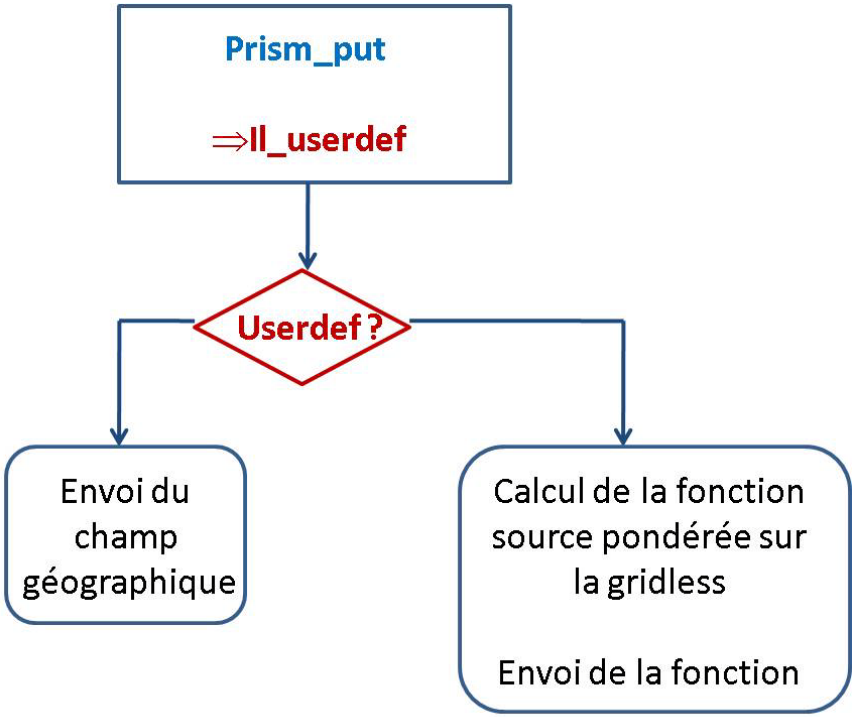


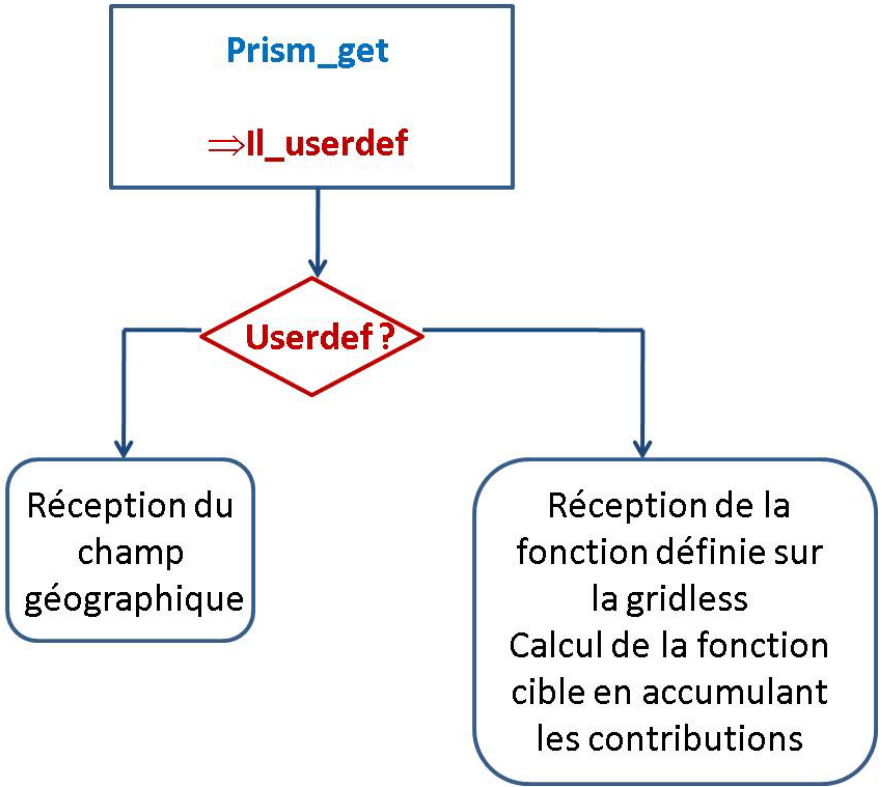
Utilisation des gridless pour échanger les données

Fonctionnement du « Userdef »

- Pas de modifications du smioc: interpolation définie comme les autres, avec le nom du fichier de poids et adresses

  ⇒Tout est fait **automatiquement** dans le driver (oasis4) et le psmile (librairie de communication) : définition des « gridless », des nouveaux champs de couplage définis sur ces « gridless » -> remplaceront les champs et grilles « géographiques »
- Recherche des intersections entre les blocs de la gridless des procs source et cible
- Chaque proc source envoie les points sur chacun des procs cibles auxquels il contribue
- **Prism_put** / **Prism_get** : envoi et réception du champ physique





Validation 1

Nombreux tests en parallèle sur **CORAIL, CURIE, JADE** depuis août 2011:

- CORAIL (CERFACS): HP PROLIANT - 2304 cœurs sur 96 noeuds, 32 Go/noeud, 211 Gflops/nœud
- JADE (CINES): SGI Altix ICE 8200 - 23040 cœurs sur 2880 nœuds, 30 G0/nœud, 267 Tflop/s
- CURIE (CCRT): Bullsx S6010 - 11520 cœurs sur 360 nœuds,128 Go/nœud,105 Tflop/s + Bullx B510 -> petaflops

- **Jouet**: deux modèles couplés échangeant une fonction analytique sur un pas de temps.
- **Grilles**: T359 (181724 points) et Orca025T (1442x1021 points).
- T359 partitionnée selon J par bandes de latitudes entières.
- Orca025T partitionnée en I et J en carré.
- Fichiers de poids obtenus à partir des fichiers de poids OASIS3 (en mono)
- Comparaison avec les résultats obtenus avec OASIS3 en monoprocesseur (web)

Problèmes rencontrés: grand nombre de blocs, mémoire globale augmente quand le nombre de processeurs augmente, variables d'environnement MPI mal positionnées

⇒Debogage

⇒ Jouet : lecture de la grille locale seule et non globale

⇒ Psmile : lecture du fichier de poids et adresses par tranches de 5000 liens

Le jouet a tourné sur **201 procs** sur JADE (100 procs par modèle + 1 procs OASIS4)

⇒Objectif 500 puis 1000 processeurs par modèle sur JADE

⇒ Augmentation du nombre de champs, du nombre de pas de temps

Validation 2

Support à Mike Hobson sur l'interpolation des champs de couplage avec un fichier de poids et adresses dans le modèle du Met Office (Unified Model - NEMO):

- **Grilles** : Orca1(T,U,V) (360 x 292 points) pour l'océan
  reglonlat (192x145 points) pour l'atmosphère
- **38 champs de couplage** : 20 de l'atmosphère vers l'océan
  18 de l'océan vers l'atmosphère

Débogage : lié au nombre élevé de champs

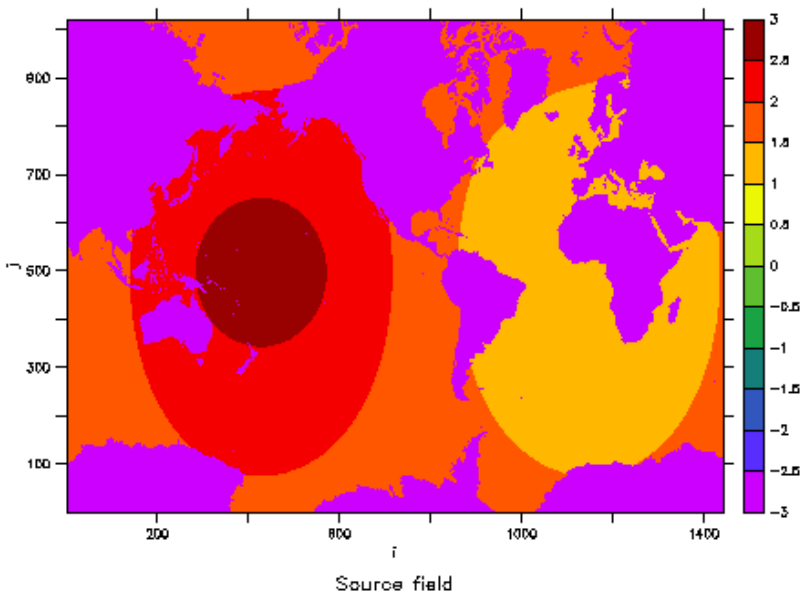
- Jouet simple (10 liens) échangeant 4 puis 15 champs de couplage dans un sens
  ⇒Bug dans la mémoire allouée dans un bsend entre le driver et les modèles
  ⇒Tableau pointant vers un champ désalloué dans une routine, donc non défini
- Jouet représentant le vrai modèle couplé avec échanges dans les 2 sens
  ⇒Pointeur non nullifié lorsqu'on augmentait le nombre de champs

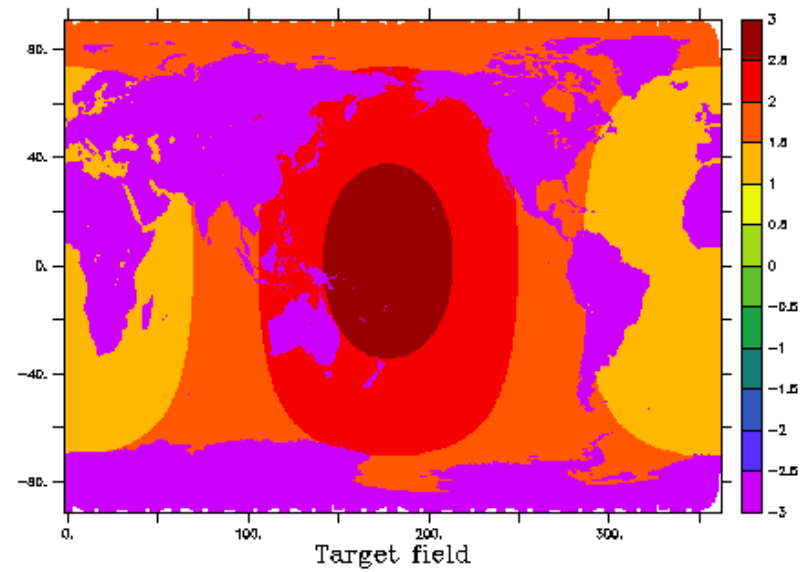
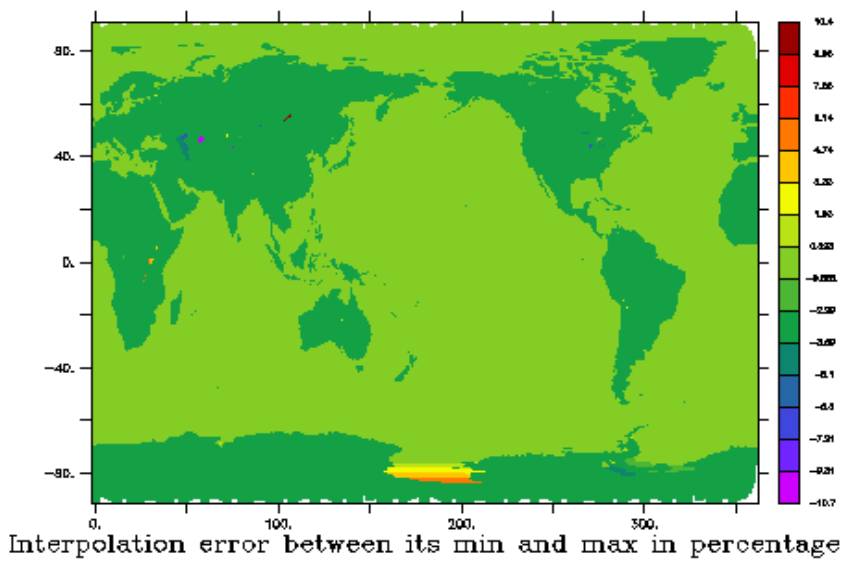
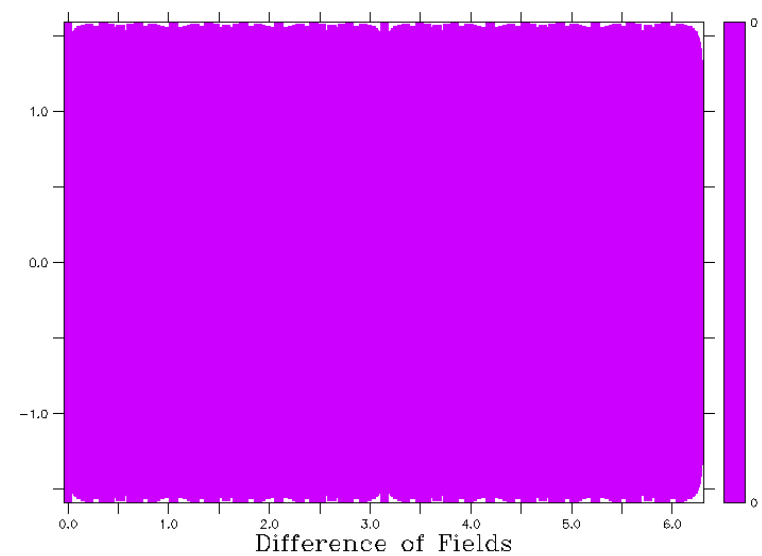
Actuellement, le modèle tourne en **monoprocesseur** sur une IBM Power 6. Problème quand le nombre de processeurs augmente => la mémoire globale augmente (lié au fait qu'il y a un fichier de poids par champ pour l'instant ?) …

Merci de votre attention

Questions ?

Interpolation bilinéaire Orca025T (29x2 procs) vers T359 (58 procs)





max 10%
sinon 1%