

# Réalisation d'un modèle couplé océan-atmosphère avec le coupleur dynamique de codes parallèles OpenPALM

P. Trespeuch

7 décembre 2011

- 1 Introduction
- 2 Les modèles ARPEGE et NEMO
- 3 Le couplage avec OpenPALM
- 4 Mes développements
  - Maillage et partitionnement de la grille gaussienne réduite
  - Domaines de complétion
  - Gestion des points isolés
- 5 Contraintes d'utilisation de CWIPI
- 6 Tests
  - Qualité de l'interpolation
  - Tests de performances
- 7 Conclusion

- **Reproduire** avec OpenPALM **les couplages** de modèles jouets déjà couplés avec OASIS, en particulier reproduire le **modèle jouet ARPEGE-NEMO**.
- **Identifier les fonctionnalités manquantes** de CWIPI.
- **Comparer la qualité** d'interpolation obtenue avec OASIS et avec OpenPALM.
- **Comparer les temps de calculs** des deux méthodes de couplage.
- Montrer qu'il est possible d'utiliser des **grilles icosaédriques**.

- 1 Introduction
- 2 Les modèles ARPEGE et NEMO
- 3 Le couplage avec OpenPALM
- 4 Mes développements
  - Maillage et partitionnement de la grille gaussienne réduite
  - Domaines de complétion
  - Gestion des points isolés
- 5 Contraintes d'utilisation de CWIPI
- 6 Tests
  - Qualité de l'interpolation
  - Tests de performances
- 7 Conclusion

# Le modèle NEMO

Nucleus for European Modelling of the Ocean.

Le modèle **NEMO** traite la **partie océanique** du système couplé. Il est basé sur une discrétisation en différences finies 3D de l'océan.

La grille utilisée par le modèle NEMO est la grille **ORCA**. Cette grille est "logiquement-rectangulaire", les latitudes et longitudes des points peuvent être exprimés dans deux tableaux distincts à deux dimensions.

# Le modèle ARPEGE

Action de Recherche Petite Échelle Grande Échelle

Le modèle numérique **ARPEGE** est un modèle de circulation générale global et spectral développé par Météo-france en collaboration avec le Centre Européen de Prévision.

Ce modèle représente la **composante atmosphérique** du modèle couplé. La grille du modèle ARPEGE est une **grille gaussienne réduite**.

- 1 Introduction
- 2 Les modèles ARPEGE et NEMO
- 3 Le couplage avec OpenPALM**
- 4 Mes développements
  - Maillage et partitionnement de la grille gaussienne réduite
  - Domaines de complétion
  - Gestion des points isolés
- 5 Contraintes d'utilisation de CWIPI
- 6 Tests
  - Qualité de l'interpolation
  - Tests de performances
- 7 Conclusion

## Orca

```
!PALM_UNIT -name orca\  
!           -functions {f90 orca}\  
!           -parallel mpi\  
!           -minproc 1\  
!           -maxproc 100000\  
!           -object_files {Write_Result.o orca_domain.o orca.o}\  
!           -comment {modele ocean}  
!  
!PALM_CWIPY_COUPLING -name ocean\  
!  
!PALM_CWIPY_OBJECT -name surface_downward_water_flux\  
!                  -coupling ocean\  
!                  -intent OUT  
!  
!PALM_CWIPY_OBJECT -name sea_surface_temperature\  
!                  -coupling ocean\  
!                  -intent IN
```

FIG.: En-tête du fichier implémentant le modèle jouet NEMO/ORCA



# L'interface de OpenPALM

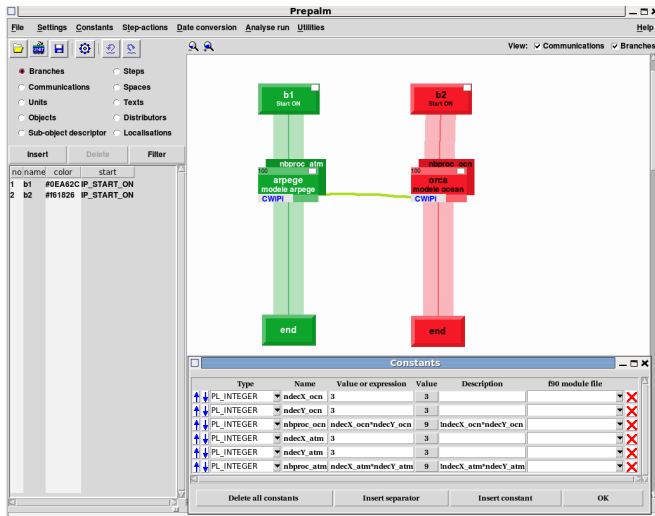


FIG.: Interface du coupleur OpenPALM

## Fonctions principales pour réaliser un couplage :

- *PCW\_Init*
- *PCW\_Create\_coupling*
- *PCW\_Define\_mesh*
- *PCW\_Send / PCW\_Recv*
- *PCW\_Finalize*

# Utilisation de la bibliothèque d'interpolation CWIPI

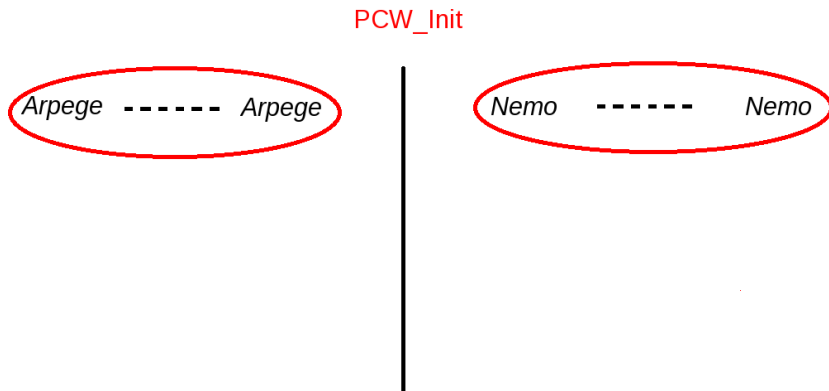


FIG.: Création des communicateurs "locaux"

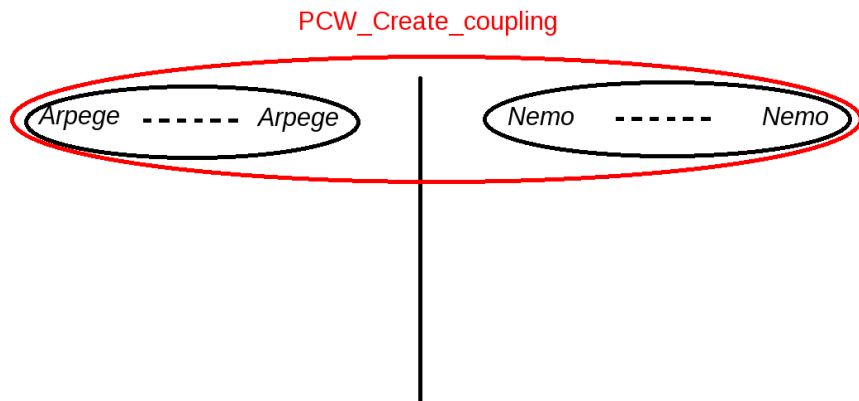


FIG.: Création de l'intercommunicateur

# Utilisation de la bibliothèque d'interpolation CWIPI

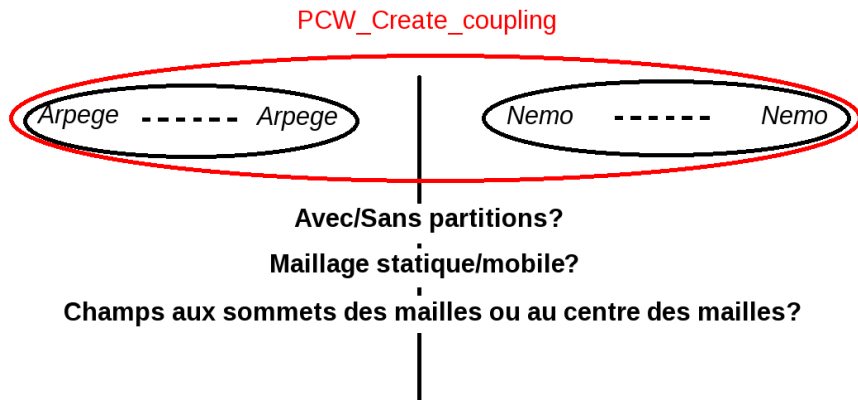


FIG.: Choix des options de couplage

# Utilisation de la bibliothèque d'interpolation CWIPI

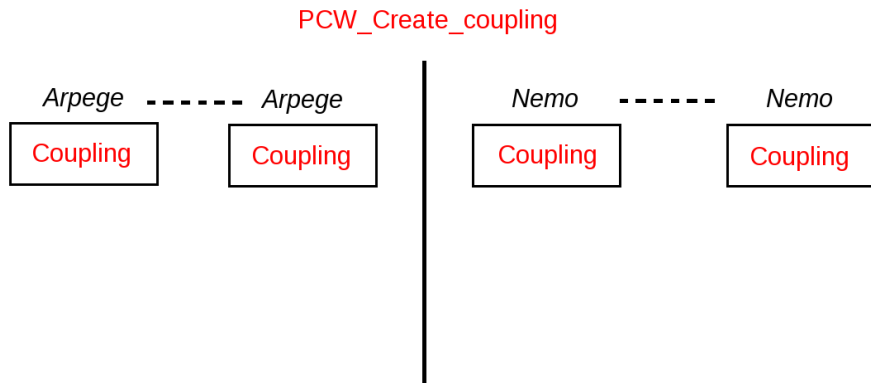


FIG.: Création des objets "couplage"

# Utilisation de la bibliothèque d'interpolation CWIPI

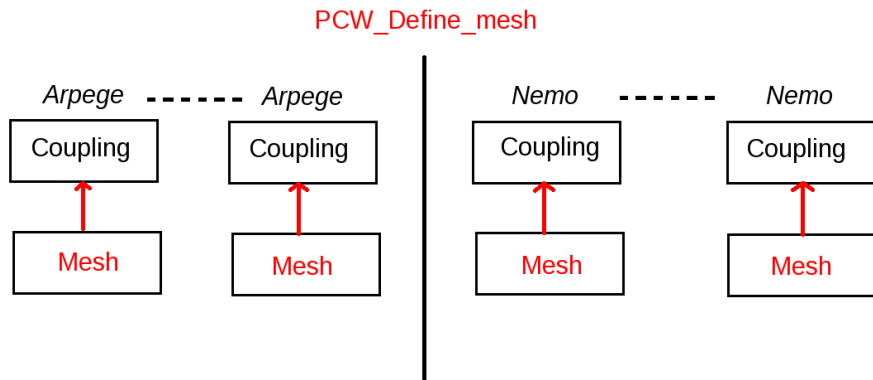


FIG.: Définition du maillage

- Un tableau contenant les coordonnées cartésiennes de chaque point du maillage.
- Un tableau contenant les indices des points composant chaque maille.
- Un tableau contenant pour chaque maille son nombre de sommets.
- Le nombre de points.
- Le nombre de mailles.



# Utilisation de la bibliothèque d'interpolation CWIPI

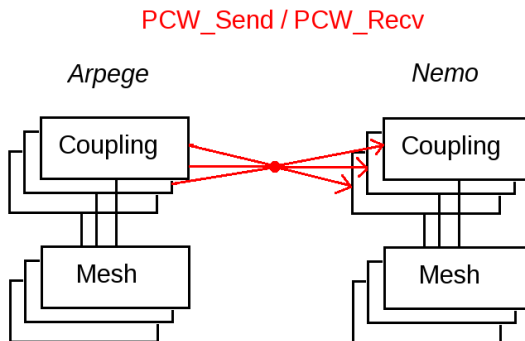


FIG.: Échanges de champs

# Utilisation de la bibliothèque d'interpolation CWIPI

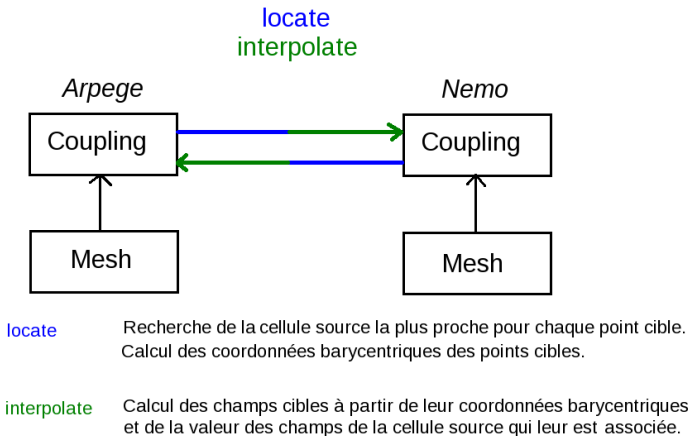


FIG.: Échanges de champs

- 1 Introduction
- 2 Les modèles ARPEGE et NEMO
- 3 Le couplage avec OpenPALM
- 4 **Mes développements**
  - Maillage et partitionnement de la grille gaussienne réduite
  - Domaines de complétion
  - Gestion des points isolés
- 5 Contraintes d'utilisation de CWIPI
- 6 Tests
  - Qualité de l'interpolation
  - Tests de performances
- 7 Conclusion

# Algorithme de maillage de la grille gaussienne réduite

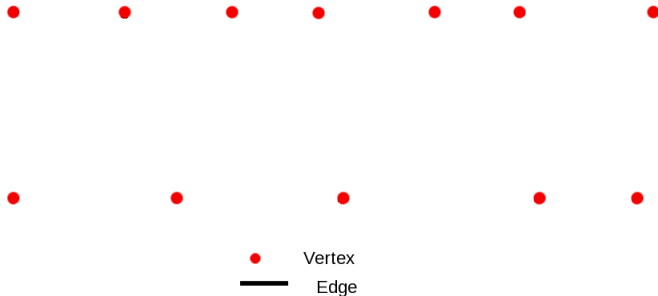


FIG.: Points.

# Algorithme de maillage de la grille gaussienne réduite

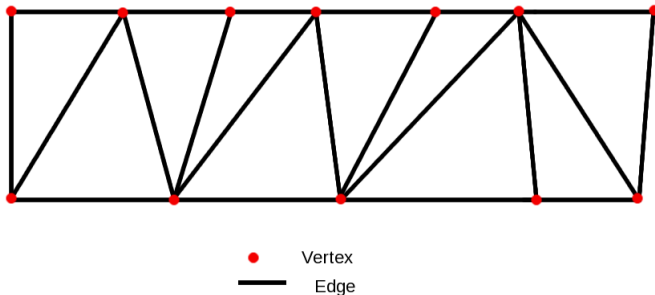


FIG.: Mailles triangulaires.

# Partitionnement de la grille gaussienne réduite

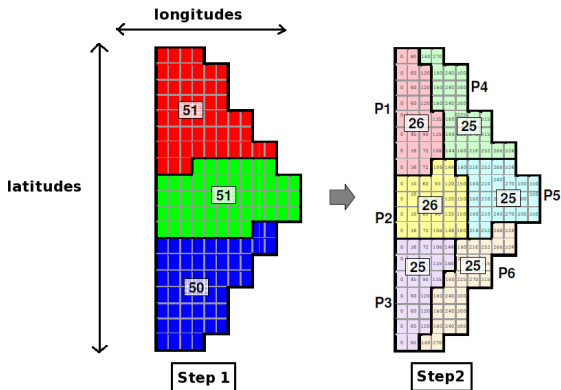
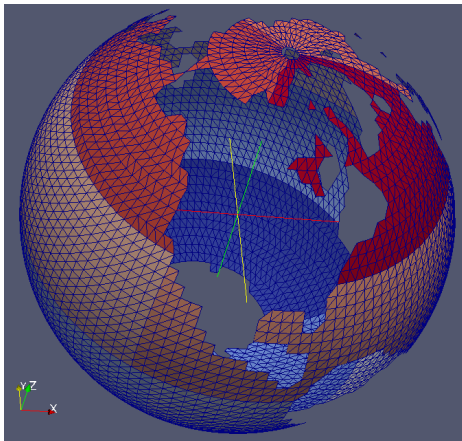


FIG.: Algorithme de partitionnement.

# Partitionnement de la grille gaussienne réduite



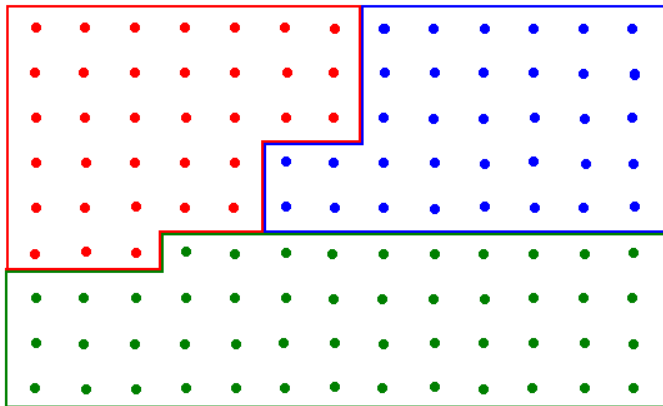
Maillage obtenu sur une grille gaussienne réduite (6232 points)

**FIG.:** Maillage obtenu sur la grille gaussienne réduite basse résolution pour 3 décompositions est-ouest et 4 décompositions nord-sud (soit un total de 12 partitions).

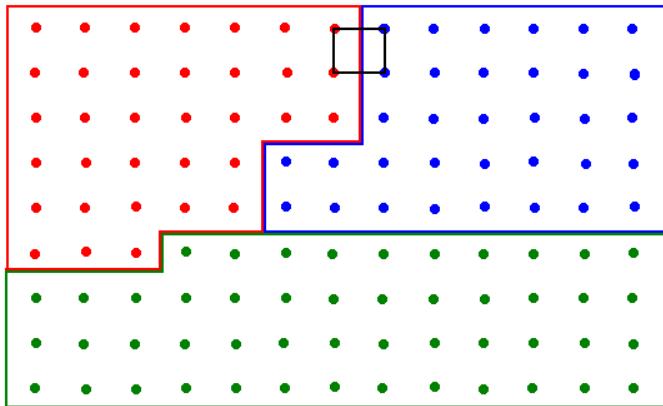
- 1 Introduction
- 2 Les modèles ARPEGE et NEMO
- 3 Le couplage avec OpenPALM
- 4 **Mes développements**
  - Maillage et partitionnement de la grille gaussienne réduite
  - **Domaines de complétion**
  - Gestion des points isolés
- 5 Contraintes d'utilisation de CWIPI
- 6 Tests
  - Qualité de l'interpolation
  - Tests de performances
- 7 Conclusion



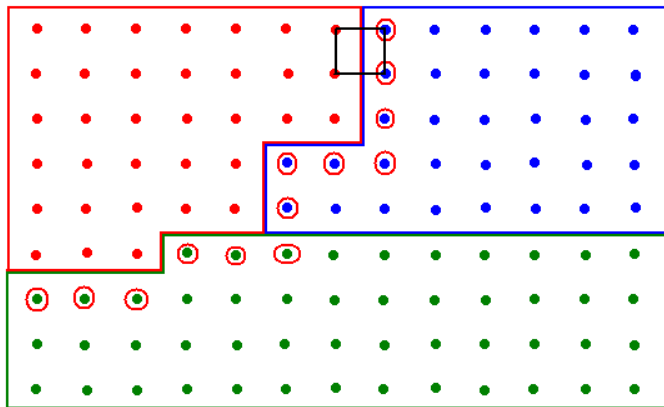
# Domaine de complétion



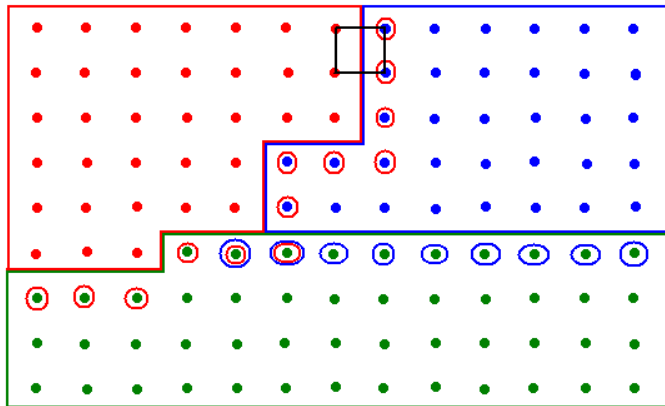
# Domaine de complétion



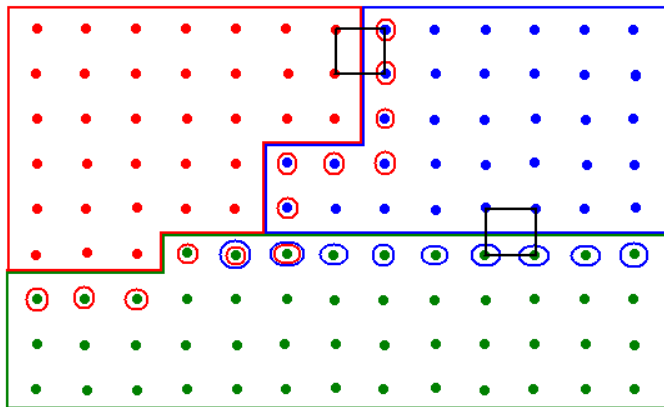
# Domaine de complétion



## Domaine de complétion



# Domaine de complétion



# Domaine de complétion

Les points sont tagués :

- **0 : Domaine de complétion :**  
La valeur du point doit être rapatriée depuis un processus distant.
- **1 : Domaine de calcul :**  
La valeur du point est calculée localement.

Une méthode **defineHollowedMesh** ajoutée à l'objet couplage ainsi qu'une surcharge du constructeur de l'objet **Mesh** permettent de transmettre les tags.

Ces tags seront contenus dans un vecteur attribut de l'objet **Mesh**.

Une méthode **completeField** est ajoutée et appelée lors de l'envoi des champs par un des modèle.

# Utilisation de la bibliothèque d'interpolation CWIPI

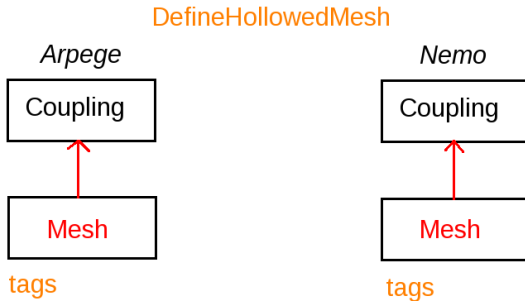


FIG.: Définition des tags

# Utilisation de la bibliothèque d'interpolation CWIPI

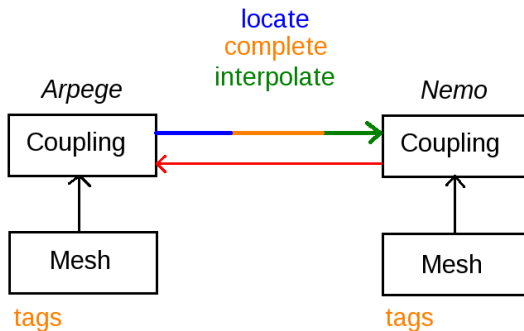


FIG.: Complétion



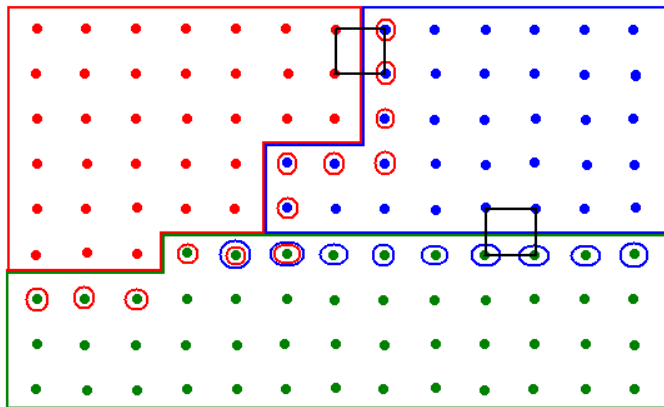
# Algorithme de complétion : localisation

Chaque processus **envoie les coordonnées des points du domaine de complétion** à tous les autres processus (MPI\_BCAST).

Chaque processus **reçoit les coordonnées des points “demandés”** par les autres et **envoie les coordonnées des points faisant partie du domaine de calcul local** (MPI\_BCAST puis MPI\_SEND).

**Deux vecteurs** sont créés contenant pour chaque processus les indices des points à recevoir et à envoyer.

# Illustration de la localisation

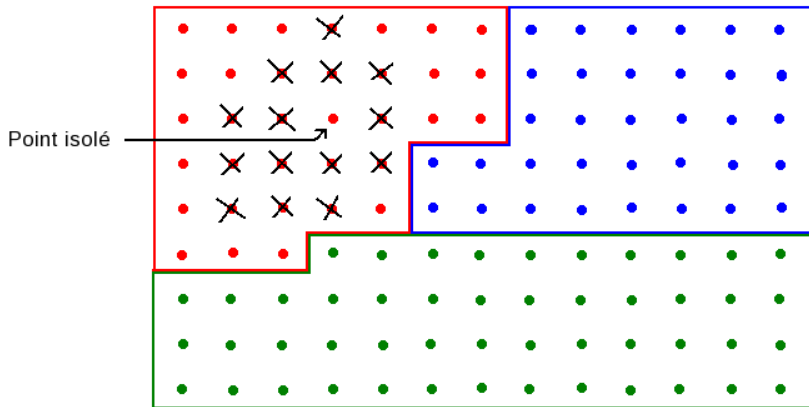


# Algorithme de complétion : échange

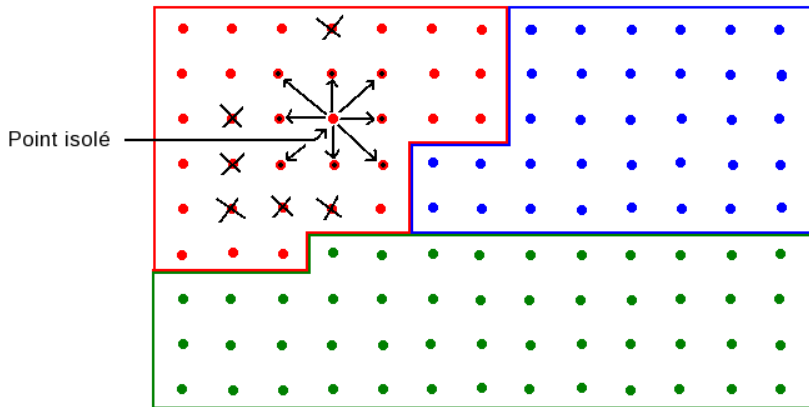
```
Pour chaque processus k
  Si mon rang est k alors
    Pour chaque processus distant p
      Réception des valeurs manquantes que possède
      le processus p (MPI_Recv)
    Fin Pour
  Sinon
    Envoi des valeurs manquantes au processus k (MPI_Send).
  Fin Si
Fin Pour
```

- 1 Introduction
- 2 Les modèles ARPEGE et NEMO
- 3 Le couplage avec OpenPALM
- 4 Mes développements**
  - Maillage et partitionnement de la grille gaussienne réduite
  - Domaines de complétion
  - **Gestion des points isolés**
- 5 Contraintes d'utilisation de CWIPI
- 6 Tests
  - Qualité de l'interpolation
  - Tests de performances
- 7 Conclusion

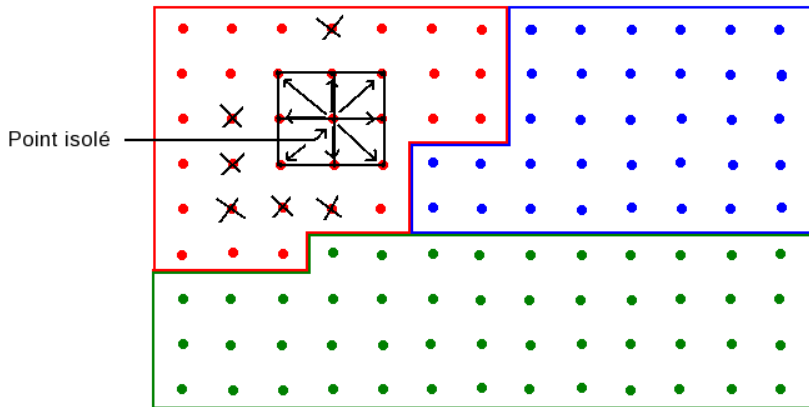
# Points isolés



# Points isolés : solution



# Points isolés : solution



Les points sont tagués :

- **0 : Domaine de complétion**

La valeur du point doit être rapatriée depuis un processus distant.

- **1 : Domaine de calcul**

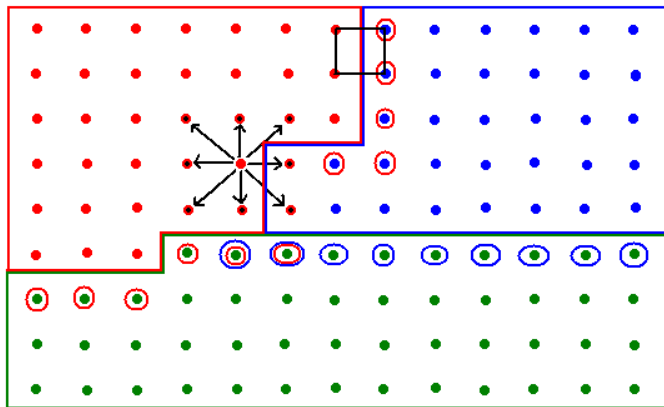
La valeur du point est calculée localement.

- **N :**

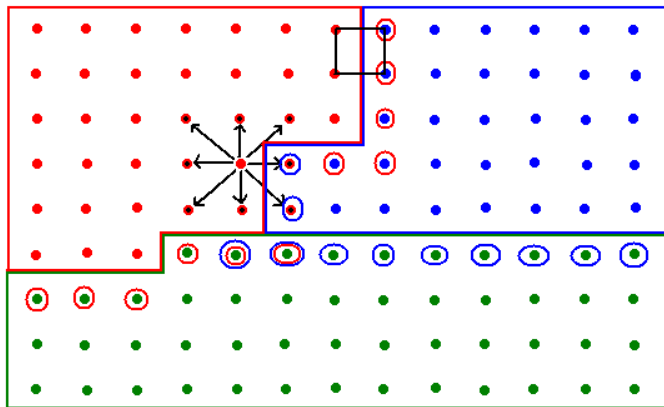
La valeur du point doit être copiée depuis le point d'indice N-1.



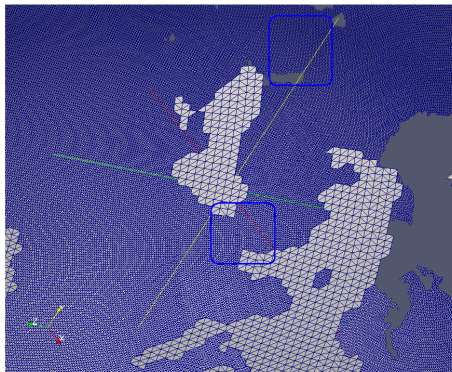
# Points isolés en bordure de partition



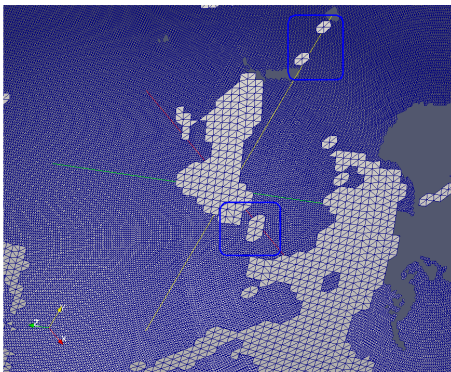
# Points isolés en bordure de partition



# Points isolés : Résultat



Sans traitement des points isolés

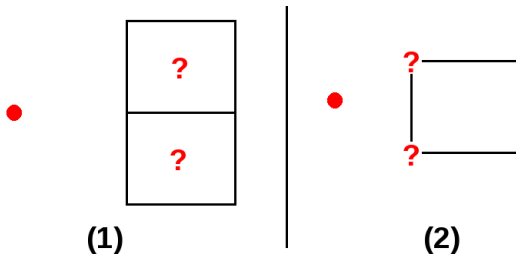


Avec traitement des points isolés

FIG.: Maillage de la grille ARPEGE.

- 1 Introduction
- 2 Les modèles ARPEGE et NEMO
- 3 Le couplage avec OpenPALM
- 4 Mes développements
  - Maillage et partitionnement de la grille gaussienne réduite
  - Domaines de complétion
  - Gestion des points isolés
- 5 Contraintes d'utilisation de CWIPI**
- 6 Tests
  - Qualité de l'interpolation
  - Tests de performances
- 7 Conclusion

# Contraintes d'utilisation



Pour éviter les différences Mono/Para lorsque tous les points cibles ne "tombent" pas dans le maillage source :

- (1) Définir les partitions dans le même ordre que les éléments.**
- (2) Les éléments doivent être définis dans le même sens et le même ordre quel que soit le partitionnement.**

FVM transforme les quadrangles en triangles pour localiser les points cibles sur la grille source. La triangularisation de FVM ne donnerait pas toujours le même résultat.

## **Définition de triangles plutôt que de quadrangles.**

Dans FVM les comparaisons de distances aux cellules sont en simple précision entre processus et en double précision à l'intérieur d'un même processus. Différences de résultats selon le partitionnement.

## **Passage aux doubles dans les deux cas.**

- 1 Introduction
- 2 Les modèles ARPEGE et NEMO
- 3 Le couplage avec OpenPALM
- 4 Mes développements
  - Maillage et partitionnement de la grille gaussienne réduite
  - Domaines de complétion
  - Gestion des points isolés
- 5 Contraintes d'utilisation de CWIPI
- 6 Tests
  - Qualité de l'interpolation
  - Tests de performances
- 7 Conclusion

# La qualité de l'interpolation

La fonction analytique suivante a été utilisée pour les tests :

$$f(x, y) = 2 - \cos\left(\pi\left(\frac{\arccos(\cos(y)\cos(x))}{1.2*\pi}\right)\right)$$

L'erreur est calculée comme suit :

$$\text{Erreur} = \frac{\text{champ\_recu} - \text{fonction\_analytique}}{\text{fonction\_analytique}} * 100$$



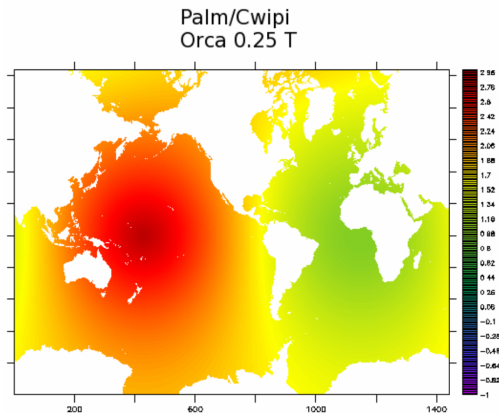


FIG.: Champs reçus sur la grille ORCA.

# La qualité de l'interpolation : grille ORCA

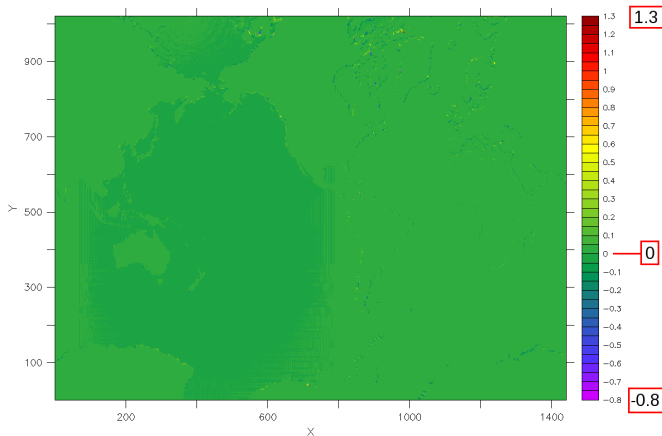


FIG.: Erreurs d'interpolation sur la grille ORCA.

# La qualité de l'interpolation : grille ORCA

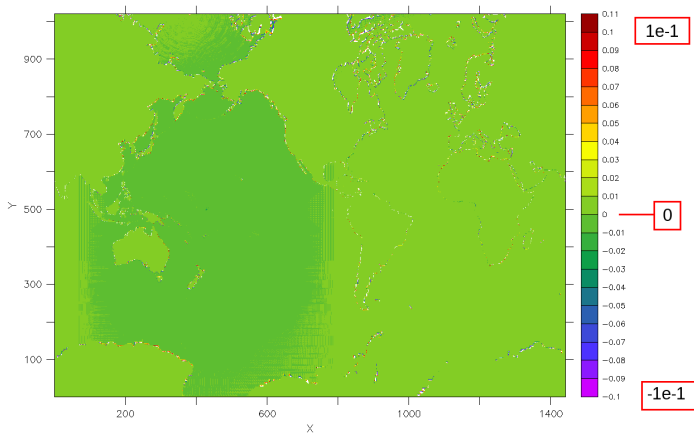


FIG.: Erreurs d'interpolation sur la grille ORCA.

# La qualité de l'interpolation : grille ORCA

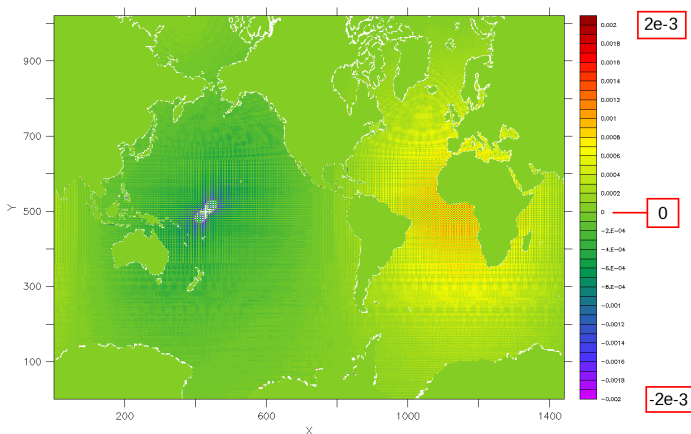
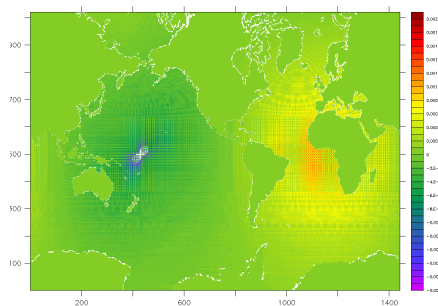


FIG.: Erreurs d'interpolation sur la grille ORCA.

# La qualité de l'interpolation : grille ORCA

OpenPALM



OASIS 3

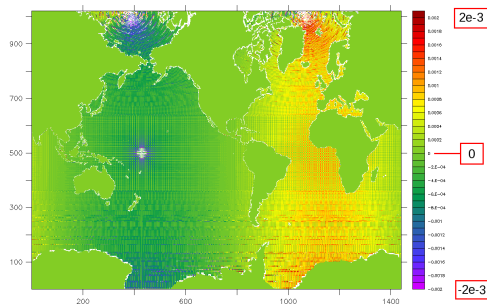


FIG.: Erreurs d'interpolation sur la grille ORCA.

# La qualité de l'interpolation : grille ARPEGE

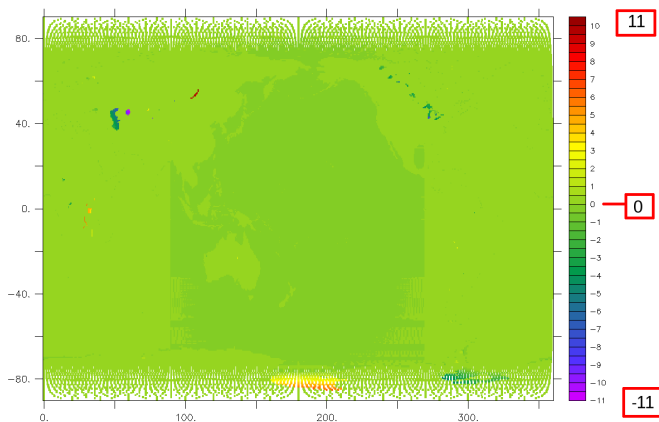


FIG.: Erreurs d'interpolation sur la grille ARPEGE.

# La qualité de l'interpolation : grille ARPEGE



FIG.: Erreurs d'interpolation sur la grille ARPEGE.

# La qualité de l'interpolation : grille ARPEGE

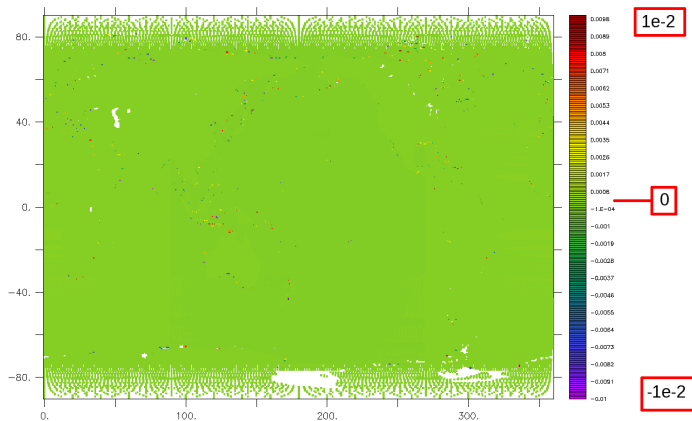


FIG.: Erreurs d'interpolation sur la grille ARPEGE.



# La qualité de l'interpolation : grille ARPEGE



FIG.: Erreurs d'interpolation sur la grille ARPEGE.

# La qualité de l'interpolation : grille ARPEGE

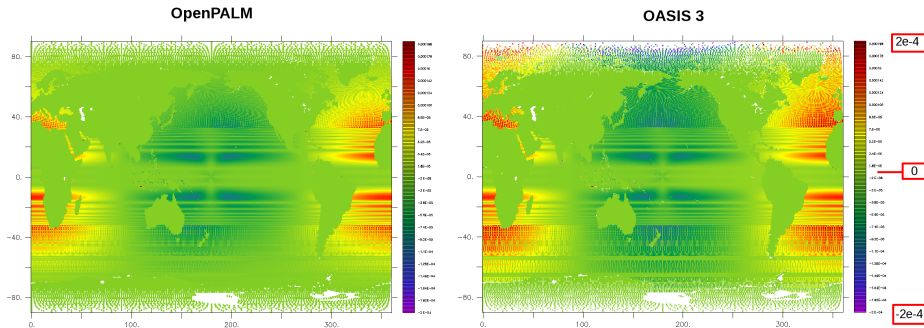


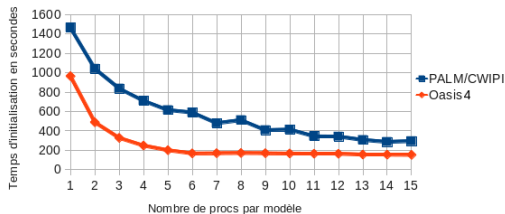
FIG.: Erreurs d'interpolation sur la grille ARPEGE.

- 1 Introduction
- 2 Les modèles ARPEGE et NEMO
- 3 Le couplage avec OpenPALM
- 4 Mes développements
  - Maillage et partitionnement de la grille gaussienne réduite
  - Domaines de complétion
  - Gestion des points isolés
- 5 Contraintes d'utilisation de CWIPI
- 6 Tests
  - Qualité de l'interpolation
  - Tests de performances
- 7 Conclusion

# Tests de performance : Initialisation

Evolution du temps d'initialisation en fonction du nombre de processus par modèle

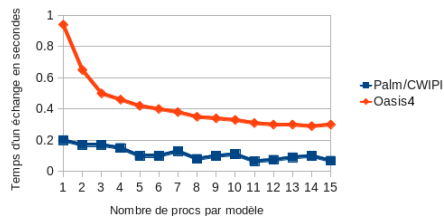
nombre de proc par modèle	PALM/CWIP1	Oasis4
1	1466	965
2	1042	489
3	838	328
4	711	249
5	615	201
6	590	168
7	480	170
8	512	172
9	408	170
10	414	167
11	345	165
12	342	164
13	307	156
14	287	155
15	295	153



# Tests de performance : Échanges

Evolution du temps d'échange en fonction du nombre de processus par modèle

nombre de proc par modèle	Palm/CWIP	Oasis4
1	0.2	0.94
2	0.17	0.65
3	0.17	0.5
4	0.15	0.46
5	0.1	0.42
6	0.1	0.4
7	0.13	0.38
8	0.08	0.35
9	0.1	0.34
10	0.11	0.33
11	0.066	0.31
12	0.074	0.3
13	0.09	0.3
14	0.1	0.29
15	0.068	0.3



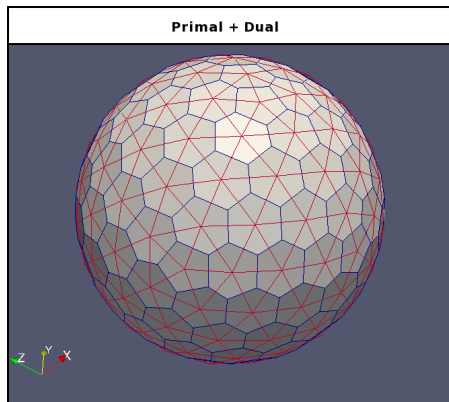


FIG.: Grille icosaèdre

- CWIPI/FVM sont des bibliothèques bien construites qui permettent d'ajouter facilement des fonctionnalités et de comprendre aisément les fonctionnalités existantes.
- Ajouter des nouveaux algorithmes d'interpolation ne poserait pas de problèmes.
- Un peu de travail nécessaire pour utiliser les grilles dont la connectivité n'est pas définie.
- Très simple d'utilisation pour les grilles non structurées.

Merci de votre attention.