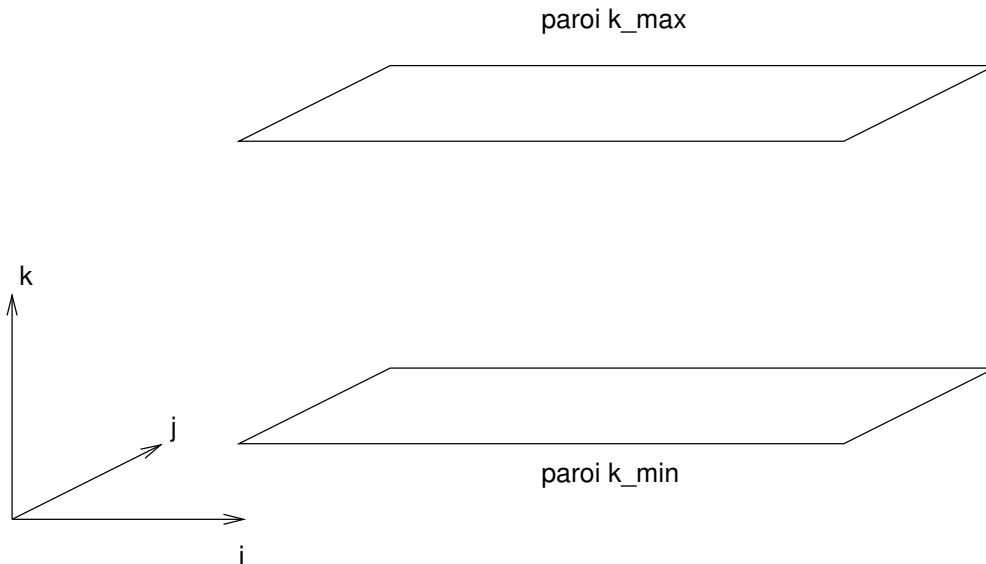


# Documentation maquette IJK + Front-tracking

Benoit Mathieu et Guillaume Bois

7 juin 2013

## 1 Conventions



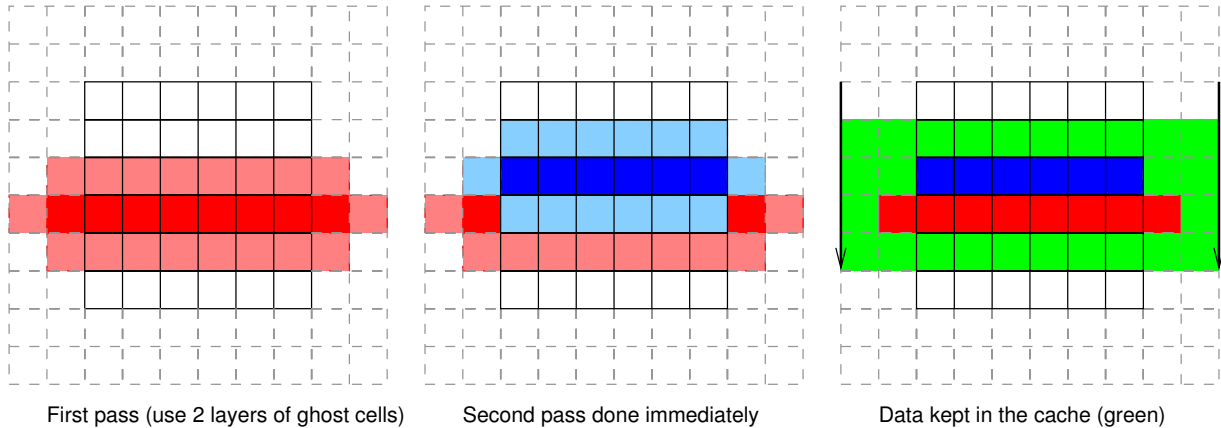
## 2 A propos du solveur multigrille

- Le seuil du solveur grossier doit être inférieur au seuil du solveur global (par exemple seuil gcp à  $10^{-10}$  pour un seuil global à  $10^{-9}$ )

```
multigrid_solver {
  solver_precision mixed
  # Liste des operateurs de deraffinement, le nombre d'opérateurs
  # donne le nombre de niveaux.
  4 operateurs => niveaux 0 (origine), 1, 2, 3, 4 (niveau grossier)
  Ici, niveau grossier aura  $2^4$  fois moins de mailles par direction que
  le niveau 0 (16 fois) #
  coarsen_operators 4
    Coarsen_Operator_Uniform { }
    Coarsen_Operator_Uniform { }
    Coarsen_Operator_Uniform { }
    Coarsen_Operator_Uniform { }
  # parametre d'optimisation "temporal blocking" #
  ghost_size 1
  pre_smooth_steps 1 7
  smooth_steps 1 7
  relax_jacobi 1 0.7
  # solveur utilise pour le maillage le plus grossier (niveau 4) avec seuil #
  solveur_grossier GCP { impr seuil 1e-10 precondition ssor { omega 1.5 } }
  # seuil du solveur multigrille global: #
  seuil 1e-8
  nb_full_mg_steps 2 20 1
  impr
}
```

## 2.1 temporal blocking

Le temporal blocking est une optimisation de l'algorithme Jacobi qui fait plusieurs itérations de lissage de Jacobi en parcourant une seule fois les champs d'entrée et de sortie. Pour cela on a besoin de plusieurs couches de mailles fantomes (ghost\_size), autant que de passe simultanées. Sur les processeurs récents, l'optimum est ghost\_size=4 à 6 pour des maillages de taille 32 à 64 de côté en i et j, par processeur.



Voir présentation pici2 du 25/03/2011.

Attention : le parametre ghost\_size ne peut pas etre plus grand que la taille du maillage le plus grossier sur un processeur.

## 3 Quelques infos sur le Front-tracking

### 3.1 Classe Descripteur\_FT

Classe envoyant la liste des sommets/facettes aux autres processeurs.

Il y a 2 Desc\_Structure\_FT : un pour les sommets et un pour les facettes : desc\_sommets\_et desc\_facettes\_

Chacun possède un espace virtuel et un espace distant. Le virtuel est celui qu'il reçoit d'autres processeurs ; le distant est celui qu'il envoie aux autres.

Il y a un objet maillage par proc.

PROC1:

```
doit envoyer le sommet 0 au processeur 2
doit envoyer le sommet 4 au processeur 2
```

PROC2:

```
doit recevoir, dans l'ordre, du processeur 1:
    le premier sommet reçu dans le sommet 3,
    le deuxieme sommet reçu dans le sommet 4
```

PROC1:

```
maillage.desc_sommets_.espace_distant_.elements_[2] =
    { 0, 4 }
```

PROC2:

```
maillage.desc_sommets_.espace_virtuel_.elements_[1] =
    { 3, 4 }
```

sur le processeur 1:

```
appeler maillage.creer_sommets_virtuels(
    liste_sommets = { 0, 4 }
    liste_pe = { 2, 2 }
)
```

c'est une methode parallele, le proc 1 va envoyer des donnees  
et le proc 2 va en attendre en reception,  
il faut donc appeler la methode sur le proc2 aussi, en meme temps:

PROC2:

```

maillage.creer_sommets_virtuels(
    liste_sommets= {}
    liste_pe = {}
)

```

## 3.2 Liste des classes Front-tracking utilisées et/ou modifiées

- Intersections\_Elem\_Facettes et Intersections\_Elem\_Facettes\_Data : base de données contenant pour chaque maille volumique la liste des facettes qui la traverse, et pour chaque facette la liste des mailles volumiques traversées. Pour chaque couple, on connaît le barycentre de la surface d'intersection et la surface de l'intersection (fraction de la surface totale de la facette).

## 4 Journal

### 4.1 13/05/2013

- Répertoire "test" : un jdd de test pour voir.
- Source : Création de la classe IJK\_FT.

### 4.2 15/05/2013

- Création d'un maillage d'une sphère avec gmsh ConvectionSphere/Sphere1.geo. Mise au point d'un convertisseur vers le lata :  
`./msh_to_lata.sh Sphere1.msh test.lata`
- Sources – Création de méthodes :  
`Maillage_FT_IJK::deplacer_sommets_ijk`. Il faut finir de coder le transport des points en parallele.  
`Maillage_FT_IJK::lire_maillage_ft_dans_lata` : pour la relecture d'un maillage.

### 4.3 17/05/2013

- Sources – Création de méthodes :  
`IJK_Interfaces::transporter_maillage`. Transport des interfaces par un champ vectoriel ;  
`IJK_Interfaces::sauvegarder_interface`. Sauvegarde des interfaces. La reprise se fait naturellement ;
- Mise au point d'un premier test.  $\mathcal{C}$

### 4.4 06/06/2013 et 07/06/2013

- Réflexion parallélisation & périodicité ;
- Notion de proc. virtuels ; Besoin d'un proc virtuel face à chaque proc. Un seul à gauche par exemple ne suffit pas.
- Volonté de réutiliser les drapeaux des éléments virtuels pour leur rajouter l'indication de la périodicité ( $\pm x_{et}/ou \pm y_{et}/ou \pm z$ ). En ajoutant 0, +1 ou -1 devant l'entier (en binaire), on n'aura pas besoin de changer le prototype des routines du FT

## 5 Mémo

### 5.1 gdb

```

(gdb) p displacement.operator(0,0)
$2 = (double &) @0x53b1ce0: 0.5
(gdb) p displacement.operator(0,1)
$3 = (double &) @0x53b1ce8: 0.20000000298023224
(gdb) p displacement.operator(0,2)
$4 = (double &) @0x53b1cf0: 2
(gdb) p displacement.data_[0]@3
$5 = {0.5, 0.20000000298023224, 2}
(gdb) p displacement.data_[0]@9
$6 = {0.5, 0.20000000298023224, 2, 1.5, 0.20000000298023224, 2, 0.5, 1.2000000476837158,
(gdb) p (&displacement.operator(0,0))@9

```

Only values in memory can be extended with '@'.  
(gdb) p &(deplacement.operator(0,0))@9  
Only values in memory can be extended with '@'.

## 5.2 hg