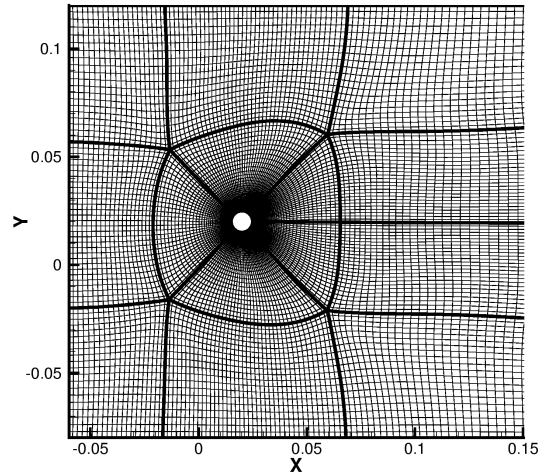


Partitionnement optimal de grilles structurées multi-blocs

Patrice Castonguay¹

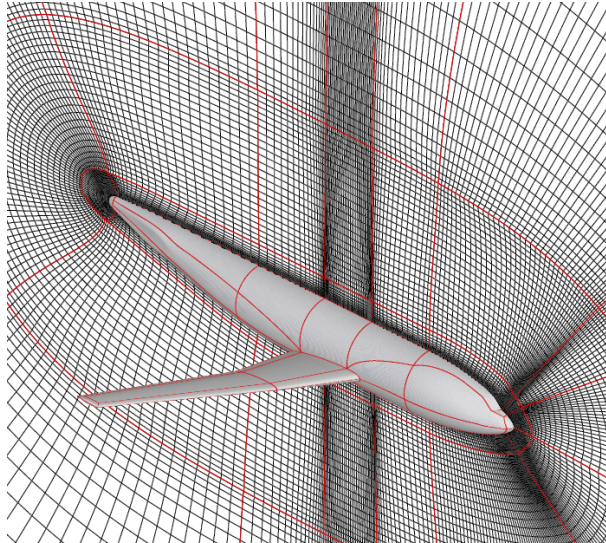
Bombardier Aéronautique, Montréal

Chez Bombardier nous utilisons souvent le logiciel de DFC appelé Fansc, qui permet de résoudre les équations de Navier-Stokes sur des grilles structurées multi-blocs. Une grille structurée multi-blocs contient un arrangement non structuré de blocs hexaédriques, où chaque bloc contient lui-même une grille structurée. Voici un exemple de telle grille (en deux dimensions).

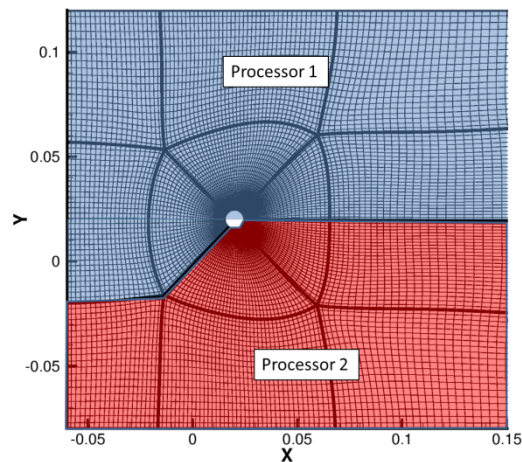


On trouvera ci-dessous un autre exemple pour une géométrie d'aéronef (les frontières des blocs sont représentées en rouge). En général les grilles que nous utilisons contiennent un nombre de blocs compris entre 40 et 100 et environ 20 millions de cellules.

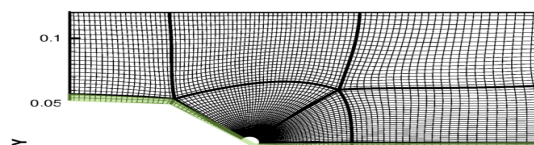
¹ Département d'aérodynamique avancée, tél.: 514-855-5001 x.12345,
patrice.castonguay@aero.bombardier.com

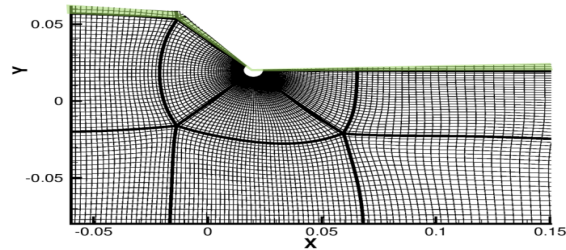


Pour accélérer les calculs sur de telles grilles, de multiples unités de traitement (ou cœurs) sont utilisées, où chaque unité de traitement a pour tâche de calculer la solution sur une portion de la grille. Dans le logiciel Fansc, ce sont les blocs qui sont affectés à des unités de traitement. En voici un exemple, où l'unité 1 doit calculer la solution sur les 8 blocs bleus et l'unité 2 doit la calculer sur les 6 blocs rouges.

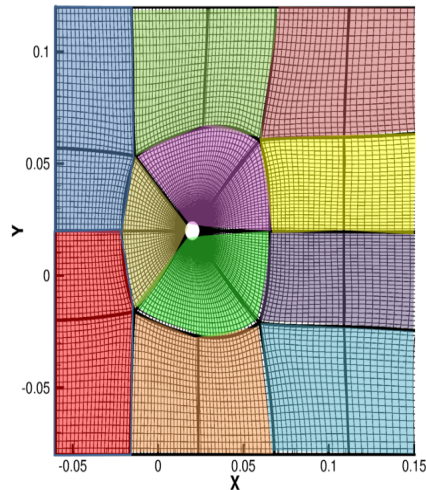


Parce que les unités de traitement travaillent en parallèle, il faut qu'elles communiquent entre elles fréquemment pendant le calcul de la solution, afin d'échanger des informations à la frontière de leurs domaines respectifs. Dans le logiciel Fansc, les informations contenues dans deux couches de cellules (appelées les cellules du halo) sont échangées à intervalles réguliers pendant le calcul de la solution. Les cellules du halo sont représentées en vert dans la grille ci-dessous.





Pour que la distribution des tâches entre les processeurs permette des gains de temps appréciables, les nombres de cellules affectées à deux processeurs distincts doivent être semblables (cette propriété s'appelle « un bon équilibrage de charge »). De plus les échanges d'informations entre unités de traitement doivent être minimisés. Généralement le nombre de blocs dans la grille est indépendant du nombre d'unités de traitement et le nombre de cellules peut varier d'un bloc à l'autre. Par conséquent nous nous donnons la possibilité de scinder des blocs afin d'obtenir un bon équilibrage de charge et de minimiser les échanges d'informations. Dans l'exemple ci-dessous, certains blocs ont été scindés afin d'obtenir un bon équilibrage de charge pour 11 unités de traitement.



Lorsque le nombre de blocs est très grand, il est difficile de trouver la manière optimale de scinder les blocs et les affecter aux unités de traitement. De plus nous voulons faire en sorte que les blocs soient aussi grands que possible parce que c'est plus efficace d'un point de vue algorithmique.

Des heuristiques simples pour scinder les blocs et équilibrer les charges ont été présentées en [1]. Bien que ces algorithmes améliorent l'efficacité du calcul parallèle de la solution (au moins pour les tailles de grilles et les nombres de processeurs utilisés jusqu'ici), ils n'essaient pas de minimiser les échanges d'informations ou les coûts liés aux cellules de halo.

Nous sommes donc à la recherche d'un algorithme qui scinde les blocs (et peut-être les fusionne) et les affecte à des unités de traitement. L'algorithme doit avoir les qualités suivantes.

- 1) Il doit produire un bon équilibrage de charge.
- 2) Il doit minimiser les échanges d'informations entre unités de traitement.
- 3) Il doit produire des blocs aussi grands que possible.

Référence

- [1] K. Sermeus, E. Laurendeau, F. Parpia, « Parallelization and performance optimization of Bombardier multiblock structured Navier-Stokes solver on IBM eserver cluster 1600 », AIAA Paper 2007-1109.