

TP2 : Comparaison de modèles de turbulence pour un écoulement turbulent décollé derrière une marche descendante

Objectif :

Ce sujet consiste à modéliser sous le logiciel libre **OpenFoam** un écoulement turbulent décollé. Nous nous inspirons encore une fois d'un des cas-tests de la base de validation de modèles de turbulence de la NASA.

Ce premier TP nous permettra de

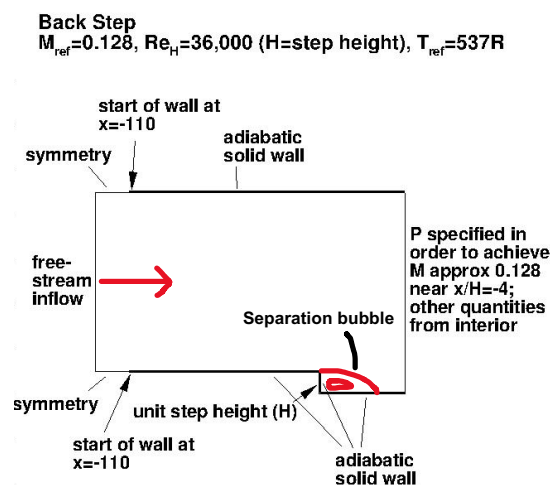
1. Comparer quelques modèles de turbulence parmi les plus utilisés pour un cas avec gradient de pression est décollement, pour lequel la turbulence est loin d'un état d'équilibre.
2. Apprendre à mettre en données plusieurs modèles de turbulence sous OpenFoam, dont le modèle aux tensions de Reynolds de Launder, Reece et Rodi (LRR)
3. Etudier l'influence de la discrétisation numérique des équations de transport sur la robustesse des simulations RANS
4. Comparer les modèles avec les données issues d'expériences en soufflerie

1. Description du problème

Nous étudions le cas d'un écoulement dans un canal avec un élargissement brusque de section, de type « marche descendante ». Le cas fait partie de la base de validation de la NASA : https://turbmodels.larc.nasa.gov/backstep_val.html

Sur ce site l'on trouve la description de la géométrie et des conditions de l'écoulement. On peut également y trouver les données expérimentales de Driver et Seegmiller (AIAA Journal, 1985).

L'écoulement est schématisé sur la figure 1. L'écoulement incident est caractérisé par un nombre de Mach (basé sur la vitesse du son et les conditions thermodynamiques à l'infini) $Ma=0.2$. Le nombre de Reynolds par unité de longueur de plaque (semi-infinie), basé sur la viscosité cinématique de l'écoulement incident est $Re=5$ millions. D'autres détails sont fournis sur la figure 1.



L'écoulement, initialement aligné avec l'axe x, procède de la gauche vers la droite. Le nombre de Mach étant faible, les effets de la compressibilité peuvent être négligés. L'écoulement est donc régi par les équations RANS incompressibles complétées par un modèle de turbulence. L'écoulement décolle au coin de la marche et les expériences montrent qu'il recolte à une distance de $6.26H$ en aval de la marche.

1.1 Algorithme de résolution

Nous recherchons une solution stationnaire des équations RANS. Pour cela nous utilisons de nouveau l'algorithme SIMPLE, disponible dans le solveur `simpleFoam`.

1.2 Modèles de turbulence

Comme la dernière fois, nous allons utiliser les modèles `kOmegaSST` et `kEpsilon` standard. De plus, nous allons considérer la version RNG du modèle k-epsilon (Yakhot et Orszag), qui est censée avoir une meilleure performance pour des écoulements décollés grâce à l'introduction d'une fonction d'amortissement nonlinéaire dans la formulation de la viscosité tourbillonnaire. Par ailleurs, compte-tenu des conditions hors équilibre de l'écoulement, nous allons étudier un modèle aux tensions de Reynolds, et plus précisément le modèle LRR.

2. Travail à effectuer

Après avoir démarré le docker avec OpenFoam, recopier dans votre repertoire de travail le tutoriel de la marche descendante :

```
cp -r $FOAM_TUTORIALS/incompressible/simpleFoam/backwardFacingStep2D .
```

Le répertoire `backwardFacingStep2D` contient deux scripts permettant d'automatiser certaines opérations.

- `Allclean` : permet d'effacer les données des calculs
- `Allrun` : permet de lancer le calcul préconfiguré dans le tutoriel et de tracer quelques résultats : le profil du coefficient de frottement $C_f = \frac{\tau_{wall}}{\frac{1}{2}\rho U_{ref}^2}$ le long de la paroi inférieure et les profils de vitesse longitudinale u/U_{ref} (adimensionnés par rapport à la vitesse de référence) à $x=H$, $x=4H$, $x=6H$ et $x=10H$ derrière la marche. U_{ref} est la vitesse de l'écoulement à l'entrée du domaine (uniforme)

Le maillage et les schémas de discrétisation, ainsi que les méthodes de résolution, ont déjà été préconfigurées dans le dossier `system`. Par ailleurs le fichier `constants/turbulenceProperties` et le dossier `0` ont été préconfigurés pour le modèle `kOmegaSST`.

Activité 1 :

1.1 A l'aide de la documentation en ligne de OpenFoam, étudier et commenter brèvement la structure des dictionnaires suivants :

- `0` : `nut`, `k`, `omega`, `epsilon`
- `constant` : `turbulenceProperties`

- `system:controlDict`, `fvSchemes`, `fvSolution`

1.2 A l'aide la documentation en ligne de OpenFoam, étudier et commenter brièvement la structure du script `Allrun`.

Activité 2 : Travail sur le modèle komegaSST

2.1 Lancer `Allrun` et calculer la solution pour le modèle `kOmegaSST`. Le script génère automatiquement une figure avec la distribution du coefficient de frottement et une autre avec les profils de vitesse aux stations indiquées plus haut.

Vérifier la sensibilité de la solution au nombre d'itérations utilisées et s'assurer que le temps de simulation préconfiguré dans `controlDict` est suffisant pour obtenir une solution stationnaire convergée.

Modifier `controlDict` afin d'insérer l'extraction de la quantité y_{plus} le long de la paroi inférieure. Vous pouvez vous inspirer du script `controlDict` utilisé pour le cas de la plaque plane dans le TP1.

2.2 En s'inspirant du fichier `Allrun`, créer un nouveau script de post-traitement `plotAll`. Dans ce script, intégrer les lignes nécessaires à :

- Tracer la distribution de y_{plus} le long de la paroi inférieure
- Superposer les données de Driver&Seegmiller pour le C_f à la courbe issue de la simulation.
- Créer une figure séparée pour chacun des profils de vitesse longitudinale, où l'on superposera les données de Driver&Seegmiller.

Nota1 : les données disponibles sur le site NASA sont au format Tecplot. Vous trouverez les données adaptées à la lecture avec `gnuplot` sur Moodle

Nota2 : vous être libre de remplacer `gnuplot` avec un autre logiciel de visualisation si cela est plus simple pour vous.

Activité 3 :

On va maintenant mettre en données le calcul pour `kEpsilon` standard. Pour cela il faut modifier le fichier `constants/turbulenceProperties`. Dans le dossier 0 il y a déjà un script pour epsilon. Vérifier dans `system/fvSchemes` et `system/fvSolution` que tout est bien configuré pour la variable epsilon.

3.1 Lancer le calcul avec `./Allrun>out&`, puis visualiser le déroulé du calcul avec la commande `tail -f log.simpleFoam` (utiliser CTRL C pour sortir). Qu'observez-vous au niveau de la convergence de la solution?

3.2 On modifie le schéma de résolution pour les termes convectifs des équations de transport turbulentes. Dans `fvSchemes`, on remplace le schéma «`bounded Gauss limitedLinear 1`» (schéma avec reconstruction linéaire, donc d'ordre 2) avec le schéma «`bounded Gauss upwind`». Qu'observe-t-on au niveau de la convergence ?

3.3 On retient désormais la discrétisation «`bounded Gauss upwind`». Lancer le calcul et s'assurer que le nombre d'itérations effectué est suffisant pour assurer une bonne convergence vers l'état stationnaire.

3.4 Générer les mêmes figures qu'au point 2.2.

3.5 Enfin, on considère la variante RNG de kepsilon. Pour cela il faut utiliser la clé «`RNGkEpsilon`» dans `turbulenceProperties`. Lancer le calcul. S'assurer comme

d'habitude de la bonne convergence et changer les paramètres si nécessaire. Tracer les distributions de C_f , y^+ et les profils de vitesse comme au point précédent.

Que peut-on conclure de la précision de `kOmega`, `kEpsilon` et `RNGkEpsilon` ?

Activité 4 : On s'intéresse désormais au modèle aux tensions de Reynolds LRR. Pour cela, il faut tout d'abord utiliser la clé « LRR » dans `turbulenceProperties`. Par ailleurs, il est nécessaire de créer le fichier d'initialisation et conditions aux limites pour le tenseur de Reynolds, noté R dans OpenFoam. Au lieu de l'écrire à la main, nous allons nous servir d'un outil de post-traitement qui génère ce fichier de façon automatique à partir des fichiers disponibles pour `kEpsilon`.

Après avoir recopié le dossier dans lequel vous avez fait tourner `kEpsilon` (par exemple dans un nouveau dossier nommé LRR), lancez la commande :

```
simpleFoam -postProcess -func R
```

Cela génère dans le dossier 0 un fichier nommé `turbulenceProperties :R`. Renommez-le en `R`. Il contient les informations nécessaires à l'initialisation de R calculées à partir du R de `kEpsilon`.

Dans `system/fvSchemes` spécifier les schémas de discrétisation des équations de transport pour R . Rester sur « bounded Gauss upwind » pour les termes convectifs et « Gauss linear » pour les termes diffusifs.

De même, choisir un schéma de résolution pour la variable R dans `system/fvSolution`.

4.1 Lancer le calcul en partant du temps 0. Que se passe-t-il ?

4.2 Lancer le calcul comme une reprise du calcul `kEpsilon`. Que se passe-t-il ?

4.3 Régler les paramètres dans `fvSolution` au mieux pour assurer une convergence satisfaisante de la solution. Que peut-on dire du coût de calcul et la robustesse du modèle ?

4.4 Post-traiter les résultats et les comparer aux cas précédents. Que peut-on conclure sur la précision de LRR ?