

# **ANNEXE**

## A. Solvers ANSYS FLUENT

### A.1. Introduction :

Comme tout logiciel de CFD, il est composé de trois éléments : le préprocesseur, le solveur ainsi que le postprocesseur.

- La définition de problème à résoudre s'effectue à l'aide du **préprocesseur** ANSYS Design Modeler et Meshing. Il permet de représenter la géométrie, discrétiser le domaine par plusieurs algorithmes de maillage et nommer les différents composants et/ou matériaux (fluide ou solide).
- Le **solveur** permet de définir numériquement les conditions opératoires dans lesquelles sera effectué la simulation, ainsi que la spécification des conditions aux limites. Enfin, il permet de choisir le processus itératif en proposant divers schémas numériques pour la discrétisation spatio-temporelle et pour le couplage vitesse/pression.
- Le **postprocesseur** est l'élément qui permet d'afficher les résultats obtenus. Il rend possible la visualisation des champs de vecteur vitesse, les champs de pression, de turbulence ainsi que toutes les autres grandeurs sur un segment, une section ou tout le domaine. Il permet aussi de tracer des courbes [23].

ANSYS FLUENT nous permet de choisir l'une des deux méthodes numériques :

- Solveur basé sur la pression
- Solveur basé sur la masse volumique

Historiquement, l'approche basée sur la pression a été développée pour les écoulements incompressibles à basse vitesse, tandis que l'approche basée sur la masse volumique a été principalement utilisée pour les écoulements compressibles à grande vitesse. Cependant, les deux méthodes ont récemment été étendues et reformulées pour résoudre et fonctionner pour un large éventail de conditions de flux allant au-delà de leur intention traditionnelle ou originale. Dans les deux méthodes, le champ de vitesse est obtenu à partir des équations de quantité de mouvement. Dans l'approche basée sur la masse volumique, l'équation de continuité est utilisée pour obtenir le champ de masse volumique tandis que le champ de pression est déterminé à partir de l'équation d'état.

D'autre part, dans l'approche basée sur la pression, le champ de pression est extrait en résolvant une équation de correction de pression ou de pression obtenue en manipulant les équations de continuité et de quantité de mouvement. En utilisant l'une ou l'autre des méthodes, ANSYS FLUENT résoudra les équations intégrales gouvernantes pour la conservation de la masse et de la quantité de mouvement, et (le cas échéant comme le nôtre) pour l'énergie et d'autres scalaires tels que la turbulence et les espèces chimiques. Dans les deux cas, une technique basée sur le volume de contrôle est utilisée. Elle consiste à :

- Division du domaine en volumes de contrôle discrets à l'aide d'une grille de calcul.
- Intégration des équations régissant les volumes de contrôle individuels pour construire des équations algébriques pour les variables dépendantes discrètes ("inconnues") telles que les vitesses, la pression, la température et les scalaires conservés.
- Linéarisation des équations discrétisées et solution du système d'équations linéaires résultant pour obtenir des valeurs mises à jour des variables dépendantes.

Les deux méthodes numériques utilisent un processus de discrétisation similaire (volume fini), mais l'approche utilisée pour linéariser et résoudre les équations discrétisées est différente [24].

## A.2. Solveur basé sur la pression (Pressure-Based Solver)

Deux algorithmes de résolution basés sur la pression sont disponibles dans ANSYS FLUENT. Un algorithme séparé et un algorithme couplé. Ces deux approches sont discutées dans les sections ci-dessous.

### A.2.1. Algorithme Séparé Basé Sur La Pression

Le solveur basé sur la pression utilise un algorithme de résolution dans lequel les équations qui régissent sont résolues de manière séquentielle (c'est-à-dire, séparées les unes des autres). Les équations de base étant non linéaires et couplées, la boucle de solution doit être réalisée de manière itérative afin d'obtenir une solution numérique convergente. Dans l'algorithme séparé, les équations individuelles régissant les variables de solution (e.g.,  $u, v, w, p, T, k, \epsilon$ , ect.) sont résolues les uns après les autres. Chaque équation directrice, tout en

étant résolue, est "découplée" ou "séparée" des autres équations, d'où son nom. L'algorithme séparé utilise efficacement la mémoire, car les équations discrétisées n'ont besoin d'être stockées dans la mémoire qu'une à la fois. Cependant, la convergence des solutions est relativement lente, dans la mesure où les équations sont résolues de manière découplée [24].

Avec l'algorithme séparé, chaque itération comprend les étapes illustrées dans la figure (A.1) et décrites ci-dessous.

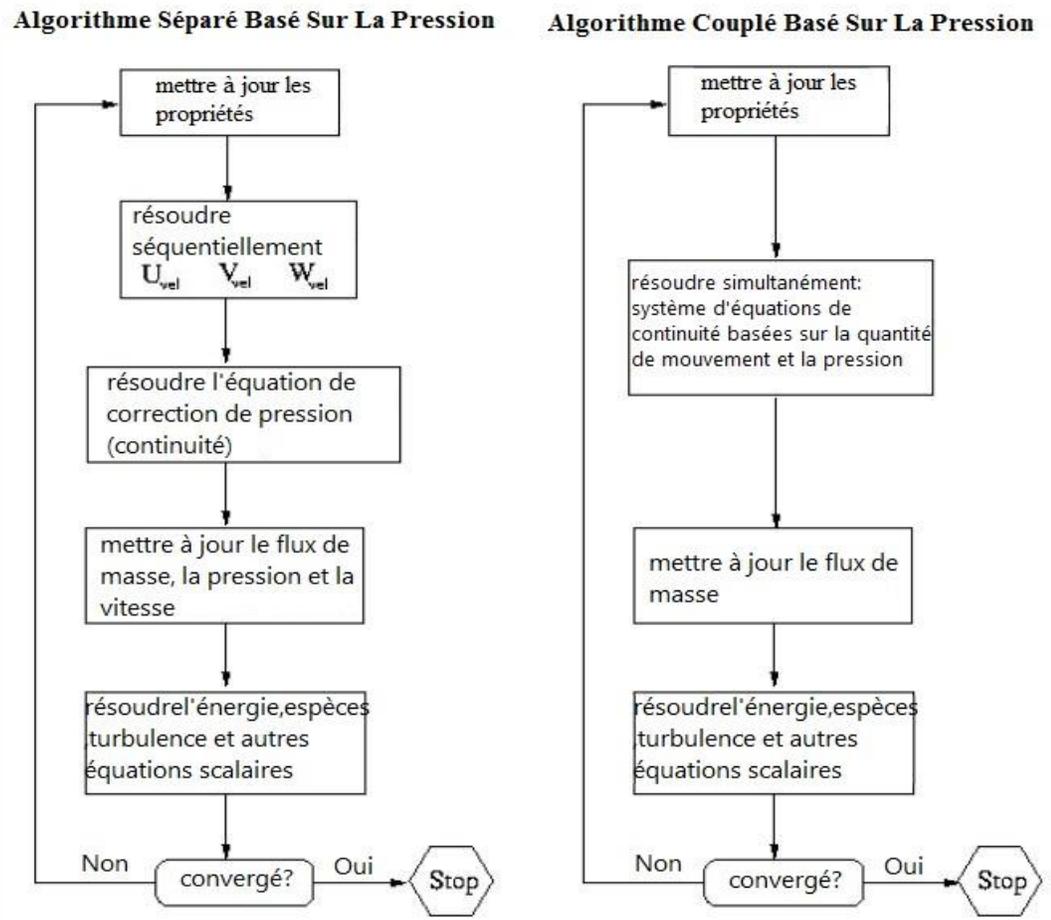


Figure (A.1) : Présentation des méthodes de solutions basées sur la pression [24].

### A.2.2. Algorithme Couplé Basé Sur La Pression

Contrairement à l'algorithme séparé décrit ci-dessus, l'algorithme couplé basé sur la pression résout un système d'équations couplées comprenant les équations de quantité de mouvement et l'équation de continuité basée sur la pression. Ainsi, dans l'algorithme couplé, les étapes 2 et 3 de l'algorithme de solution séparée sont remplacées par une étape unique dans

laquelle le système d'équations couplé est résolu. Les équations restantes sont résolues de manière découplée comme dans l'algorithme séparé.

Puisque les équations de moment et de continuité sont résolues de manière étroitement couplée, le taux de convergence de la solution s'améliore considérablement par rapport à l'algorithme séparé. Toutefois, les besoins en mémoire augmentent de 1,5 à 2 fois ceux de l'algorithme séparé, car le système discret de toutes les équations de continuité basées sur la quantité de mouvement et la pression doit être stocké dans la mémoire lors de la résolution des champs de vitesse et de pression (plutôt qu'un seul l'équation, comme c'est le cas avec l'algorithme séparé).

### A.3. Solveur basé sur la masse volumique (Density-Based Solver)

Le solveur basé sur la densité résout les équations qui régissent la continuité, la quantité de mouvement et (le cas échéant) le transport d'énergie et d'espèces simultanément (c'est-à-dire couplés ensemble). Les équations qui régissent des scalaires supplémentaires seront résolues ensuite et séquentiellement (c'est-à-dire, séparées les unes des autres et de l'ensemble couplé). Les équations de base étant non linéaires (et couplées), plusieurs itérations de la boucle de solution doivent être effectuées avant d'obtenir une solution convergée [24]. Chaque itération comprend les étapes illustrées à la figure (A.2) et décrites ci-dessous :

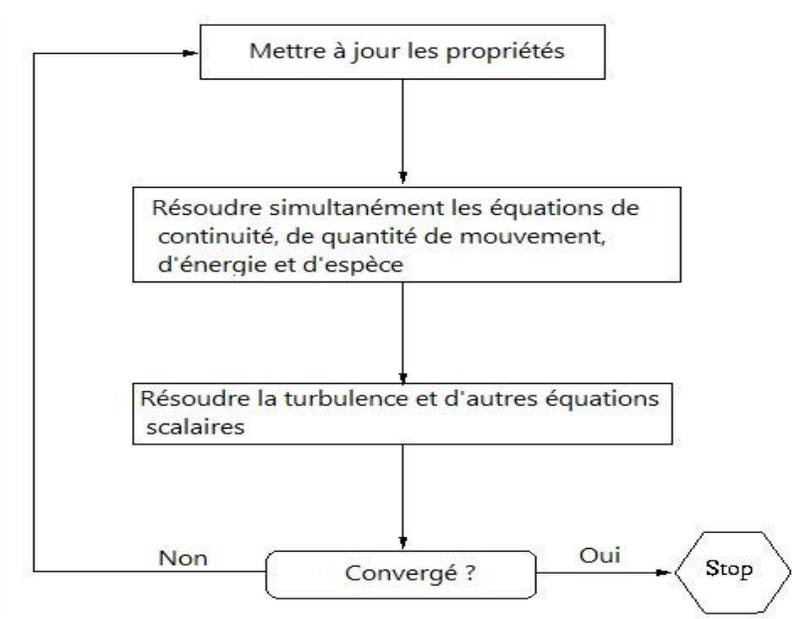


Figure (A.2) : Aperçu de la méthode de solution basée sur la densité [24].

## A.4. Algorithmes ANSYS Fluent

### A.4.1. Algorithme SIMPLE :

En dynamique des fluides numérique (CFD), l'algorithme SIMPLE est une procédure numérique largement utilisée pour résoudre les équations de Navier-Stokes. SIMPLE est l'acronyme de Semi-Implicit Method for Pressure Linked Equations (Méthode semi-implicite pour les équations liées à la pression).

L'algorithme SIMPLE a été développé par le professeur Brian Spalding et son étudiant Suhas Patankar à l'Imperial College de Londres au début des années 1970.

L'algorithme est itératif. Les étapes de base de la mise à jour de la solution sont les suivantes :

- Définir les conditions aux limites.
  - Calcul des gradients de vitesse et de pression.
  - Résoudre l'équation de moment discrétisée pour calculer le champ de vitesse intermédiaire.
  - Calcul des flux de masse non corrigés sur les faces.
  - Résoudre l'équation de correction de pression pour produire les valeurs de cellule de la correction de pression.
- Mettre à jour le champ de pression :  $p^{k+1} = p^k + urf \times p'$  où urf est le facteur de sous-relaxation pour la pression.
- Mettre à jour les corrections de pression limite  $P_b'$
- Corriger les flux de masse faciale  $\dot{m}_f^{k+1} = \dot{m}_f^* + \dot{m}_f'$
- Corrigez les vitesses des cellules  $\vec{v}^{k+1} = \vec{v}^* - \frac{Vol \nabla p'}{\vec{a}_p^v}$  : où  $\nabla p'$  est le gradient des corrections de pression,  $\vec{a}_p^v$  est le vecteur des coefficients centraux pour le système linéaire discrétisé représentant l'équation de vitesse et Vol est le volume de la cellule [26].

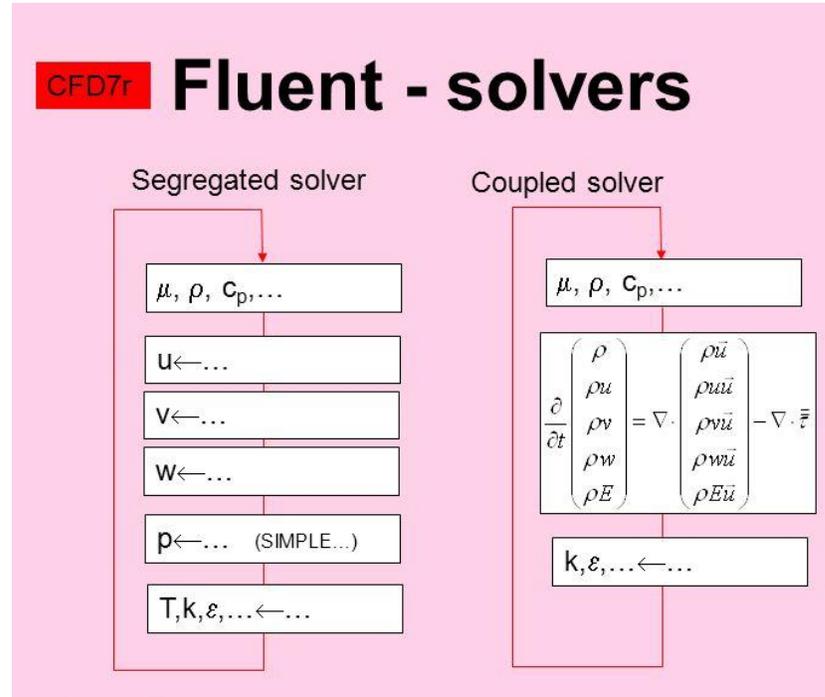


Figure (A.3) : Solvers Fluent [25].

#### A.4.2. Algorithme PISO :

L'algorithme PISO (de l'anglais Pressure-Implicit with Splitting of Operators, pression-implicite avec fractionnement des opérateurs) a été proposé par Issa en 1986, sans itérations, avec des pas de temps importants et un effort de calcul moindre. C'est une extension de l'algorithme SIMPLE utilisé dans la dynamique des fluides numérique pour résoudre les équations de Navier-Stokes. PISO est une procédure de calcul de pression / vitesse pour les équations de Navier-Stokes développées à l'origine pour le calcul non itératif d'un écoulement compressible instable, mais elle a été adaptée avec succès à des problèmes d'état stable. PISO implique une étape de prédicteur et deux étapes de correcteur et est conçu pour satisfaire la conservation de masse à l'aide d'étapes de correcteur de prédicteur [28].

L'algorithme peut être résumé comme suit :

1. Définir les conditions aux limites.
2. Résoudre l'équation de moment discrétisée pour calculer un champ de vitesse intermédiaire.
3. Calcul des flux de masse aux faces des cellules.

4. Résoudre l'équation de pression.
5. Corriger les flux de masse sur les faces des cellules.
6. Corriger les vitesses en fonction du nouveau champ de pression.
7. Mettre à jour les conditions aux limites.
8. Répéter à partir de 3 pour le nombre de fois prescrit.
9. Augmenter le pas de temps et répéter à partir de 1

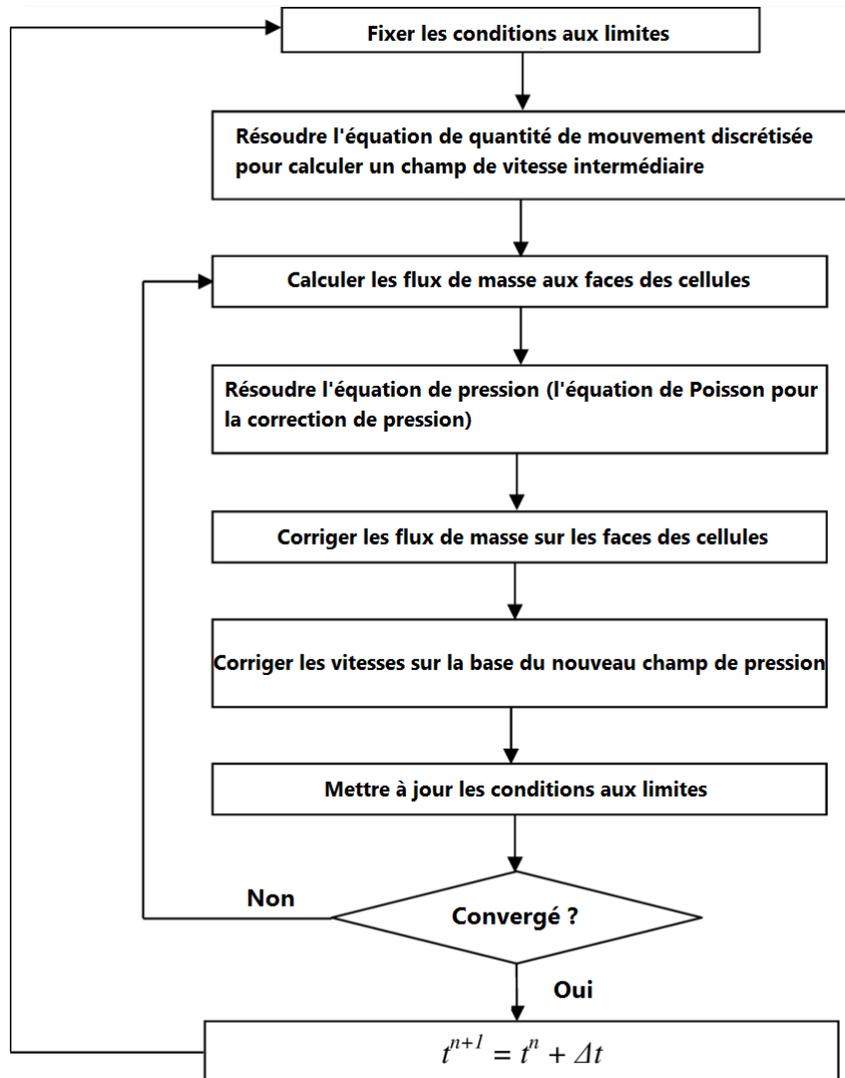


Figure (A.4) : Algorithme PISO [27].