

Guide d'utilisation du code SUNFLUIDH_EDU

Yann FRAIGNEAU

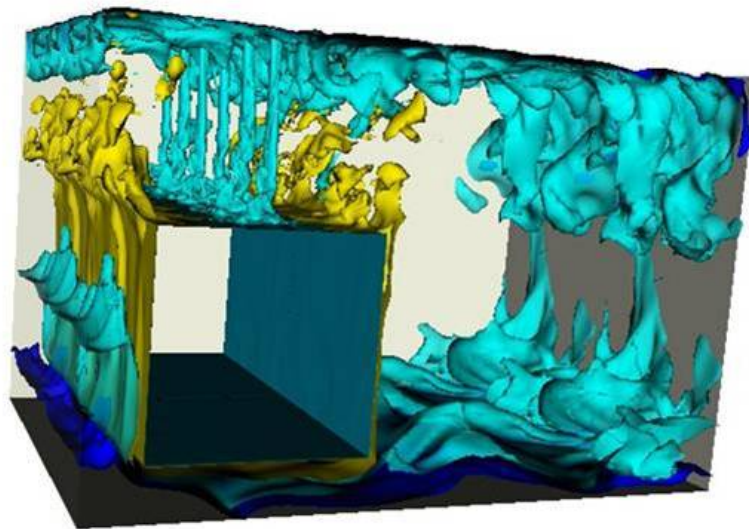


Table des matières

1	Description générale des fonctionnalités	2
1.1	Généralités	2
1.2	Maillage	2
1.3	Méthodes numériques	2
1.4	Conditions aux limites	4
1.5	Milieux à géométrie complexe	4
1.6	Résumé des fonctionnalités du codes	4
2	La compilation	5
3	Les fichiers d'entrée	6
3.1	Fichiers du maillage	6
3.2	Fichiers de reprise de simulation	6
3.3	Fichier de données principal : input3d.dat	7
4	Les fichiers de sortie	7
4.1	Fichiers de contrôle	7
4.2	Fichiers de séries temporelles	8
4.3	Fichiers des champs de données	8
4.4	fichiers de sauvegarde	9
A	Annexe : Forme générale des équations de conservation dans le contexte incompressible	10

1 Description générale des fonctionnalités

1.1 Généralités

Le développement du code SUNFLUIDH a débuté en 2011 au LIMSI pour la simulation numérique d'écoulements monophasiques instationnaires, incompressibles ou dilatables, à faible nombre de Mach. Les configurations d'étude peuvent être 2D ou 3D. Une version allégée, SUNFLUIDH_EDU, a été développée pour l'enseignement et n'intègre que la simulation d'écoulements incompressibles instationnaires.

La simulation numérique repose sur l'intégration des équations de conservation suivant une approche eulérienne formulée en vitesse–pression pour le traitement de la dynamique. La quantité de mouvement est gouvernée par les équations de Navier-Stokes, les transferts thermiques par l'équation de conservation de l'énergie. Lorsque le fluide est incompressible, la contrainte à divergence de vitesse nulle est satisfaite au moyen de la méthode de prédiction–projection incrémentale qui permet également de déterminer le champ de pression dynamique.

1.2 Maillage

La discrétisation des différentes grandeurs est réalisée sur des maillages cartésiens décalés de type MAC, ce qui permet d'assurer la stabilité numérique de la méthode de prédiction–projection. Les nœuds de discrétisation sont différents pour chacune des composantes de vitesse ainsi que pour les grandeurs scalaires (cf. figure 1). La gestion des conditions aux limites en bordure du domaine est facilitée par l'utilisation de mailles fictives placées en dehors du domaine. Le maillage de référence est associé aux grandeurs scalaires et les composantes de vitesse sont définies sur ses interfaces ($U(i + \frac{1}{2}, j, k)$, $V(i, j + \frac{1}{2}, k)$, $W(i, j, k + \frac{1}{2})$) (cf. figure 2).

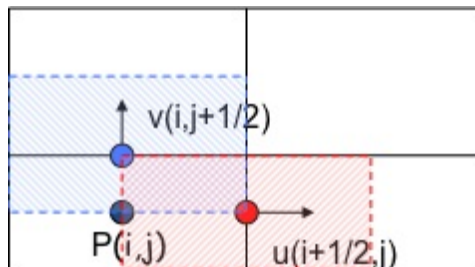


FIGURE 1 – Représentation 2D d'une configuration de grille en maillages décalés associée à la discrétisation spatiale de la pression (en noir) et des deux composantes de vitesse (en rouge et en bleu).

1.3 Méthodes numériques

Les méthodes de discrétisations des équations sont basées sur une approche volumes finis d'ordre deux en temps et en espace. La discrétisation spatiale des équations est réalisée en se basant sur les schémas numériques centrés. Plusieurs schémas de discrétisation temporelle sont proposés :

- le schéma Euler retardé d'ordre 2 (ou Backward Differentiation Formula – BDF2). Une version à l'ordre 1 est également disponible.

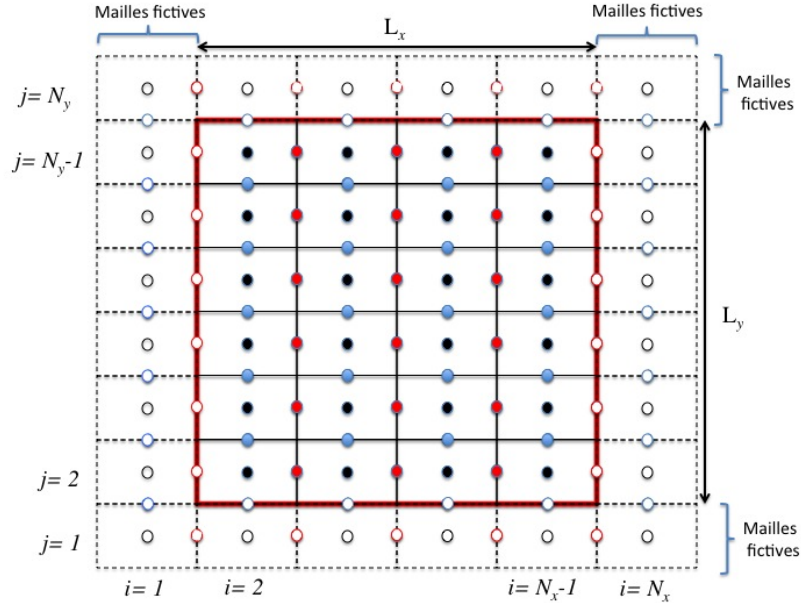


FIGURE 2 – Représentation de la discrétisation d’un domaine 2D avec les maillages décalés. Les frontières du domaine sont représentées en rouge. Nœuds associés : (i) à la composante longitudinale de vitesse U en rouge, (ii) à la composante normale de vitesse V en bleu, (iii) à la pression en noir. Cercles pleins : nœuds du domaine de calcul. Cercles vides nœuds associés aux conditions aux limites.

– le schéma de Crank –Nicolson (ordre 2).

Afin de renforcer la stabilité numérique en fonction du pas de temps, ces schémas temporels sont écrits suivant une formulation semi-implicite, où seuls les termes visqueux des équations de conservation sont définis à l’instant t^{n+1} . Les termes convectifs sont formulés de manière explicite (en fonction des champs connus aux instant t_n, t_{n-1}, \dots) et estimés à l’instant t_{n+1} à l’aide d’une extrapolation linéaire en adéquation avec la discrétisation temporelle adoptée.

La conservation de la masse (qui se réduit à la contrainte divergence de vitesse nulle) est assurée par une méthode de prédiction-projection. L’étape de prédiction fournit une estimation du champ de vitesse en intégrant les équations de Navier-Stokes. L’étape de projection consiste à calculer une fonction potentielle par résolution d’une équation de Poisson qui permet de mettre à jour le champ de pression et le champ de vitesse de telle sorte que la conservation de la masse soit satisfaite.

La résolution de l’équation de Poisson admet différentes variantes. Dans le code, on utilise une version incrémentale, c’est à dire que la solution de l’équation de Poisson correspond à l’incrément temporel de pression (Méthode de Goda). Cela vient du fait que le gradient de pression est explicitement pris en compte dans l’intégration des équations de Navier-Stokes au cours de l’étape de prédiction. L’équation de Poisson est résolue par la méthode S.O.R. (Successive Over Relaxation) couplée à une méthode multigrille de type nV-cycles afin d’accélérer la convergence de la méthode itérative.

Les détails sur l’ensemble des méthodes numériques utilisées sont présentés dans le document *"Principes de base des méthodes numériques utilisées dans le code SUNFLUIDH pour la simulation des écoulements incompressibles"*.

1.4 Conditions aux limites

Un jeu assez complet de conditions aux limites classiques permet de traiter un large spectre de configuration de simulation. Ce jeu concerne toutes les grandeurs gouvernées par les équations de conservation, sauf la pression dynamique. Les conditions aux limites disponibles sont :

- conditions de périodicité
- plans de symétrie
- conditions d'entrée : les conditions aux limites se rapportant aux composantes de vitesse sont à valeur imposée (condition de Dirichlet), ce qui permet notamment d'imposer le débit de l'écoulement. Pour la température, la masse volumique et les fractions massiques d'espèces les conditions sont également de type Dirichlet. Pour l'instant, la forme d'une entrée se cantonne à une géométrie rectangulaire.
- conditions de sortie : Elles sont de type gradient nul (Neumann) pour l'ensemble des grandeurs, sauf pour la composante normale de vitesse où une correction est apportée de manière à satisfaire la conservation du débit de l'écoulement (conditions de type Orlansky).
- conditions de paroi : Elles sont de type Dirichlet pour les composantes de vitesse (condition d'imperméabilité, d'adhérence ou de glissement). Pour les transferts thermiques, on peut définir des conditions à température imposée et à flux de chaleur imposé. On peut également définir le corps solide auquel appartient la paroi comme étant thermiquement conducteur. Dans ce cas, le transfert de chaleur dans le solide par conduction thermique est gouverné par l'équation de l'énergie, dont les termes de l'équation propres à la zone fluide sont nuls. Les conditions aux limites de parois sont considérées par défaut comme uniformes et constante sur une paroi. Toutefois, l'utilisateur peut définir des conditions aux limites variables, en déclarant ces propres modèles de fonctions dans des routines spécifiques.

Pour l'instant, la pression calculée à partir d'une équation de Poisson admet des conditions aux limites de type Neumann en toutes circonstances (la dérivée première normale au plan limite du domaine est nulle), sauf dans le cadre de conditions périodiques.

1.5 Milieux à géométrie complexe

Il est possible de définir des corps immergés dans l'écoulement. Ces corps sont des volumes solides de géométrie élémentaire (polyèdres, cylindres) sur les parois desquelles on peut placer les conditions aux limites de paroi citées ci-dessus. Pour l'instant, le code n'admet que des mailles pleinement fluide ou solide. De ce fait, la paroi d'un corps immergé ne peut être qu'alignée avec les interfaces des mailles et en aucun cas traverser les cellules. Si on définit une paroi initialement oblique par rapport au maillage, le code procède à un alignement local automatique de la paroi par rapport au maillage, ce qui engendre des structures de type "escalier". L'assemblage de plusieurs corps immergés peut être réalisé pour construire des milieux à géométrie complexe.

1.6 Résumé des fonctionnalités du codes

- code d'ordre 2 en temps et en espace, basé sur les méthodes de prédiction–projection
- discrétisation sur maillages décalées à topologie cartésienne
- géométries traitées : Configurations cartésienne et cylindrique.

- simulation 2D/3D instationnaire d'écoulements incompressibles.
- prise en compte des transferts thermiques de nature convective et conductive (dans les matériaux solide)
- traitement d'écoulements de convection forcée, naturelle et mixte en simulation numérique directe (pas de modèle de turbulence).
- propriétés physiques du fluide constante ou variable (dépendance en température de la viscosité et de la conductivité thermique estimée par la loi de Sutherland). Le fluide est supposé à γ constant, la capacité calorifique du fluide est indépendante de la température.
- conditions aux limites
 - conditions d'entrée classique de type Dirichlet (Pour les composantes de vitesse, la température, les espèces)
 - conditions de sortie de type Neumann (Pour les composantes de vitesse, la température, les espèces)
 - conditions périodiques
 - plan de symétrie
 - conditions de Parois : Conditions d'adhérence et d'imperméabilité pour la vitesse, Température imposé, flux de chaleur imposé ou matériau conducteur pour les transferts thermiques, conditions de gradient nul pour les espèces. Possibilité de définir des conditions aux limites variables à l'aide de fonctions créées par l'utilisateur.
- possibilité de définir des corps immergés pour concevoir une configuration géométrique complexe du milieu. Limitation : alignement des parois avec les interfaces du maillage.

2 La compilation

L'ensemble des fichiers sources du code (*.f90) se trouve dans un répertoire de stockage unique que nous appellerons "SOURCES". Ce répertoire contient également le makefile (fichier makefile) utilisé pour la compilation du code. En tapant simplement la commande "make", le makefile génère les modules objets (modulename.mod) et les fichiers objets (filename.o) associés aux modules sources (modulename.f90) et aux fichiers sources (filename.f90). Puis, il génère par l'édition de lien l'exécutable qui, par défaut, porte le nom "sunfluidh.x". Il est bien sûr possible de changer ce nom dans le makefile. Une fois l'exécutable créé, il est possible de supprimer les fichiers objets à l'aide de la commande "make clean". Le début du makefile est composé d'une série de variables pré-définie :

- CF : contient la commande du compilateur qu'on souhaite utiliser (par exemple, ifort, pgf95, gfortran, ...).
- FFLAGS : permet de définir les options de compilation usuelles (options de sécurité, de débogage, d'optimisation, ...) propres à chaque compilateur.
- LDFLAGS : définit l'accès aux bibliothèques scientifiques dont le code a besoin : BLAS, PBLAS, LAPACK, SCALAPACK. Elles sont utilisées pour le calcul des éléments propres des matrices tridiagonales dans le cadre de la méthode de diagonalisation partielle de l'opérateur laplacien pour la résolution de l'équation de Poisson. Le chemin d'accès à ces bibliothèques dépend de l'environnement de la machine de calcul.
- OPT : permet de définir des options de compilation propres au code : le format libre des fichiers sources si nécessaire, l'écriture des résultats binaires en format big endian, et la compilation sélective. Cette dernière option est particulièrement importante, elle permet de compiler le code dans une configuration donnée répondant aux besoins de

la simulation envisagée (configuration 2D ou 3D). Cette configuration est définie par le choix des étiquettes proposées ci-dessous :

- DIM2D : activation de la configuration géométrique en 2D
- DIM3D : activation de la configuration géométrique en 3D)

Ces étiquettes sont reconnues par le compilateur en les accolant à l'argument de compilation -D (ex : -DDIM2D pour une configuration géométrique bidimensionnelle). Pour que la compilation sélective ait lieu, il est nécessaire que l'option de compilation adéquate soit déclarée avant les étiquettes (-cpp pour ifort, -Mpreprocess pour pgi, par exemple). Si on ajoute ou supprime une étiquette dans le makefile, il est conseillé de recompiler entièrement le code pour s'assurer que l'ensemble des fichiers sources soit compilé de manière compatible (effectuer un "make clean" pour effacer les fichiers.o et .mod, suivi de "make" pour recompiler le code).

3 Les fichiers d'entrée

Ce sont les fichiers lus par le code qui contiennent toutes les données nécessaires pour réaliser une simulation numérique. Suivant le cas de figure, certains sont optionnels. Deux fichiers sont indispensables en toutes circonstances : "input3d.dat" et "num_sav_XXXX.d". L'ensemble des fichiers d'entrée est décrit dans les sections suivantes.

3.1 Fichiers du maillage

Lorsque le maillage est irrégulier (tailles de maille variables), il est construit à l'aide d'un logiciel particulier, le générateur de maillage (meshgen), dont nous reparlerons ultérieurement. Ce logiciel génère trois fichiers distincts :

- maillx.d ou maillx_XXXX.d
- mailly.d ou mailly_XXXX.d
- maillz.d ou maillz_XXXX.d

Ils sont associés aux trois directions de l'espace (dans le cas d'une simulation 2D, seuls les deux premiers fichiers sont utilisés). Quand une série de cinq chiffres XXXXX apparaît dans le nom du fichier, elle est associée au numéro de sous-domaine correspondant dans le cadre d'une simulation parallèle réalisée en décomposition de domaine. Chaque fichier contient la coordonnée correspondante associée à la position de l'interface supérieure de la maille, à savoir $x(i + \frac{1}{2})$, $y(j + \frac{1}{2})$ et $z(k + \frac{1}{2})$, respectivement. Par défaut, les premières coordonnées correspondant aux indices $i = 1, j = 1$ et $k = 1$ valent zéro, elles définissent la position des frontières inférieures du domaine de calcul (cf. figure 2). Elles ne sont pas reportées dans les fichiers dont les valeurs des coordonnées commencent aux indices $i = 2, j = 2$ et $k = 2$ et finissent aux indices $i = N_x - 1, j = N_y - 1$ et $k = N_z - 1$. Dans le cas d'un maillage régulier, ces fichiers ne sont pas pris en compte car le code génère lui-même le maillage.

3.2 Fichiers de reprise de simulation

Il est possible de segmenter une simulation particulièrement longue en plusieurs parties. Pour cela, le code doit pouvoir être exécuté à partir d'un fichier de sauvegarde généré au cours d'une précédente simulation. L'information nécessaire pour que le code puisse effectuer une reprise de calcul est contenu dans différents fichiers :

- save_fld_XXXXX_m.d : Ce fichier contient l'ensemble des champs instantanés sauvegardé par le code à un instant t_n donné pour effectuer une éventuelle reprise de la simulation à t_n . La variable 'm' est un entier pouvant prendre la valeur 1 ou 2 si le fichier est créé en cours de simulation. Elle prend la valeur 3 si le fichier est créé durant le dernier pas de temps de la simulation en cours. Ce fichier est en format binaire.
- save_XXXXX_var_m.d : Ce fichier contient diverses informations propres à l'instant de création du fichier de sauvegarde (pas de temps, temps physique de la simulation, numéro de l'itération temporelle, ...). La variable 'm' est définie comme précédemment. Ce fichier est en format binaire.
- num_sav_XXXXX.d : Ce fichier contient uniquement une valeur entière : le contenu de la variable 'n' définie ci-dessus ou zéro si aucun fichier de sauvegarde n'a été créé au cours de la simulation. IL EST IMPORTANT QUE CE FICHIER SOIT TOUJOURS PRESENT AU MOMENT DE L'EXECUTION DU PROGRAMME, même lors de la toute première simulation (dans ce cas, la valeur contenue dans ce fichier n'a aucune importance).

Comme précédemment, la série de cinq chiffres incluse dans le nom du fichier définit le numéro de sous-domaine dans le cadre d'une simulation parallèle réalisée en décomposition de domaine.

3.3 Fichier de données principal : input3d.dat

Une description complète du fichier de données est fournies sur le wiki : <https://sun-fluidh.limsi.fr>

4 Les fichiers de sortie

Définissons quelques conventions communes aux fichiers de sorties.

- "*name*" est un nom générique qui définit une série de caractères servant à distinguer les fichiers.
- "*XXXX*" est un entier codé sur cinq caractères. Il représente le numéro de sous-domaine (défini par le numéro de processus MPI) lorsque la simulation est réalisée suivant l'approche de décomposition de domaine. Dans le cadre d'un calcul mono–processeur, cette valeur vaut "00000".
- "*yyyyyyy*" est un entier codé sur sept caractères. Il représente le numéro de création du fichier.

Pour les utilisateurs confirmés, il peut être utile de savoir que les procédures d'écriture de tous les résultats sont regroupées dans la routine "Getting_Results" qui se trouve dans le fichier source "main.f90". Il est conseillé de définir toute nouvelle procédure d'écriture dans cette routine.

4.1 Fichiers de contrôle

Ils sont au nombre de trois :

- "checkdata_XXXX.d" : Ce fichier, créé en début de simulation, récapitule toutes les données d'initialisation définissant la simulation numérique.

- "check_namelist_data.d" : Ce fichier, également créé en début de simulation, répertorie l'ensemble des namelists et fournit les valeurs numériques des jeux de données initialisés par défaut ou modifiés lors de la lecture du fichier de données "input3d.dat".
- "checkcalc_XXXX.d" : Ce fichier permet de contrôler le bon déroulement du calcul en cours d'exécution. Il contient des données représentatives de la simulation (bilans de flux de chaleur, de masse, d'énergie, valeurs minimum et maximum des grandeurs physiques calculées, leurs moyennes spatiales, leurs normes L_2 (sur le domaine de calcul),...).

4.2 Fichiers de séries temporelles

Pour chacune des grandeurs sélectionnées dans le fichier de données "input3d.dat", un fichier contenant les séries temporelles associées aux sondes est créé. Son nom est défini sous la forme "name_ins_XXXX.d", avec *name* prenant les noms "u", "v" et "w" pour les composantes de vitesse associées aux directions d'indice i, j et k , "t" pour la température, "r" pour la masse volumique et "p" pour la pression dynamique.

Le format du fichier est en ASCII. Les données sont ordonnées en colonnes. La première correspond au temps et les autres se réfèrent aux différentes sondes positionnées dans l'écoulement, suivant leur ordre de déclaration dans le fichier de données "input3d.dat".

Le fichier ASCII nommé "resid_L2_Li.d" contient l'évolution temporelle de la norme L_2 et de la valeur maximum des résidus de convergence d'une solution stationnaire (cf. *Condition d'arrêt pour l'écoulement stationnaire* dans le fichier de données "input3d.dat"). Les données sont définies sur trois colonnes : le temps, la norme L_2 du résidu, sa valeur maximum. Il existe également un ensemble de fichiers similaires concernant les bilans de flux au niveau des conditions aux limites :

1. "check_massflow_balance.d" fournit l'évolution temporelle des débits massiques de l'écoulement au niveau des conditions aux limites d'entrée et de sortie, ainsi qu'un bilan de ces débits.
2. "check_volflow_balance.d" est similaire au fichier précédent pour les débits volumiques.
3. "check_heatflux_balance.d" fournit l'évolution temporelle des flux de chaleur au niveau des entrées/sortie et des parois, ainsi qu'un bilan de ces flux.

4.3 Fichiers des champs de données

Les fichiers contenant la chaîne de caractères *name*="res" se réfèrent aux champs instantanés tandis que ceux qualifiés par *name*="rst" sont associés aux champs statistiques enregistrés à l'itération temporelle *yyyyyy*.

Les champs sont une représentation spatiale des grandeurs physiques sur l'ensemble de la grille de calcul. Les données associées à ces champs sont réparties dans deux types de fichiers distincts : "*name_XXXX_yyyyyyy.d*" contient l'ensemble des champs associés aux grandeurs physiques définies dans le fichier de données "input3d.dat" ainsi que nombre d'informations caractérisant ces champs (taille de grille, nombre de champs, nom des champs, nombre de Reynolds ou nombre de Rayleigh, temps de simulation, ...). Ce fichier est en format binaire en "stream access". La routine qui gère l'écriture des champs se nomme "Recording_Fields", placée dans le fichier "define_recording_fields.f90".

La visualisation des champs contenus dans ces fichiers ainsi que le post-traitement des données

peuvent être directement réalisés avec Matlab à l'aide du programme "read_sunfluidh_data.m". Les fichiers peuvent également être traités par VisField (convertisseur de formats) pour ensuite être visualisés avec le logiciel Paraview.

4.4 fichiers de sauvegarde

cf. section 3.2

A Annexe : Forme générale des équations de conservation dans le contexte incompressible

Le système d'équations présenté ci-dessous permet de décrire l'évolution d'un écoulement incompressible. Il est composé de l'équation de continuité, des équations de conservation de quantité de mouvement et d'énergie. Le fluide est supposé newtonien et de composition chimique homogène (le fluide est composé d'une seule espèce). La force volumique présente dans l'équation de quantité de mouvement représente la résultante des forces de gravité et de flottabilité.

$$\begin{cases} \nabla \cdot (\vec{V}) = 0 \\ \rho_0 \left(\frac{\partial \vec{V}}{\partial t} + (\vec{V} \cdot \nabla) \vec{V} \right) = -\nabla P + \nabla \cdot \tau + (\rho - \rho_0) g \\ \rho_0 C_p \left(\frac{\partial T}{\partial t} + (\vec{V} \cdot \nabla) T \right) = \nabla \cdot \lambda \nabla T \end{cases}$$

ρ_0 représente la masse volumique (uniforme et constante), \vec{V} le vecteur vitesse, T la température, g la constante gravitationnelle, P la pression (elle est calculée dans la méthode de prédiction–projection de manière à satisfaire la conservation de la masse cf. le document "Principes de base des méthodes numériques utilisées dans le code SUNFLUIDH pour la simulation des écoulements incompressibles et à faible nombre de Mach"), λ la conductivité thermique et τ le tenseur des contraintes tel que :

$$\tau_{ij} = \mu \left(\left(\frac{\partial V_i}{\partial x_j} + \frac{\partial V_j}{\partial x_i} \right) \right)$$

μ étant la viscosité dynamique du fluide.

Notons que l'équation de l'énergie se ramène simplement à une équation d'advection–diffusion de la température. La dissipation visqueuse et la puissance des forces volumiques sont supposées négligeables et aucun phénomène de transfert thermique autre que la conduction et l'advection n'est considéré.

Dans le code SUNFLUIDH_EDU, les équations contiennent des coefficients paramétriques qui permettent à l'utilisateur de les définir à sa convenance sous forme dimensionnelle ou adimensionnelle. Dans le cadre d'un écoulement incompressible à propriétés physiques uniformes et constantes, elles peuvent s'écrire sous la forme :

$$\begin{cases} \nabla \cdot (\vec{V}) = 0 \\ \frac{\partial \vec{V}}{\partial t} + (\vec{V} \cdot \nabla) \vec{V} = -C_1 \nabla P + C_2 \Delta \vec{V} + C_3 (T - T_0) \\ \frac{\partial T}{\partial t} + \vec{V} \cdot \nabla T = C_4 \Delta T \end{cases}$$

La résultante des forces de gravité et de flottabilité s'exprime en faisant appel à l'hypothèse de Boussinesq, où l'on considère que l'écoulement incompressible implique que ρ est uniforme et constant, sauf pour les effets de flottabilité. On suppose qu'une perturbation de ρ est directement liée à une variation de température. Les phénomènes de flottabilité peuvent alors

directement s'exprimer en fonction de la température sous la forme :

$$\rho - \rho_0 = \beta (T - T_0)$$

Où T_0 est une température de référence associée à ρ_0 et β est le coefficient de dilatation.

Dans le cas des équations dimensionnelles, nous avons donc :

$$\left\{ \begin{array}{l} C_1 = \frac{1}{\rho_0} \\ C_2 = \frac{\mu}{\rho_0} \text{ la viscosité cinématique} \\ C_3 = \beta g \\ C_4 = \frac{\lambda}{\rho_0 C_p} \text{ la diffusivité thermique} \end{array} \right.$$