

# STAGE DE FIN D'ÉTUDE

École d'ingénieurs SUP Galilée

Université Paris nord XIII



Effectué au

Commissariat à l'Énergie Atomique et aux Énergies Alternatives

CEA Saclay - DEN/DANS/DM2S/STMF/LMEC

Du 01 avril 2015 au 30 septembre 2015

**Discipline : Mathématiques appliquées**

**Option : Calcul scientifique**

---

## Environnement de simulation pour les écoulements bouillants dans des canaux parallèles chauffés différentiellement

---

PAR : Tassadit SID ABDELKADER

### RESPONSABLES:

**Tuteur CEA :** Michaël NDJINGA, Ingénieur chercheur CEA

**Tuteur Universitaire :** Olivier LAFITTE, Professeur des Universités, Paris Nord 13

**Date de soutenance :** 25 septembre 2015



# Résumé

---

Dans le cadre du diplôme d'ingénieur en mathématiques appliquées et calcul scientifique de l'école d'ingénieurs Sup'Galilée à l'université Paris Nord, j'ai effectué mon stage de fin d'études sur *l'Environnement de simulation pour les écoulements bouillants dans des canaux parallèles chauffés différentiellement* au Cea-Saclay, du 01 avril 2015 au 30 septembre 2015.

J'ai tout d'abord effectué un projet de fin d'études (de décembre 2014 à février 2015), de recherche et étude bibliographique pour le maquettage du code industriel **FLICA 4** en 1D, afin d'améliorer la robustesse. Une des conclusions de ce premier travail est qu'un traitement spécifique est capital pour les termes sources raides propres à la thermohydraulique des coeurs de réacteurs nucléaires (schéma équilibre).

L'objectif du présent stage de fin d'études est de confirmer ces conclusions dans le cas 2D, 3D en particulier, dans les configurations présentant une recirculation dans des canaux parallèles. Nous souhaitons aussi comparer les schémas équilibres proposés lors du projet de fin d'études aux schémas précis à faible nombre de Mach (correction de pression, ou schéma au maillage décalé).

Dans le cadre de ce stage, j'ai commencé par débbugger, corriger et valider la maquette développée dans le PFE, ce qui m'a permis de monter en compétence dans la librairie Petsc et l'IDE Eclipse. Le deuxième volet du stage consistait à développer l'application CoreFlows qui permet de traiter les géométries quelconques (1D, 2D et 3D), et d'imposer les conditions limites y compris au sein de l'écoulement (cloison interne). Pour cela, j'ai du prendre en main plusieurs librairies Open Source, de gestion de maillage, calcul scientifique et de visualisation (Salomé, CDMATH, PETSc) et outils de développement logiciel. Et finalement, après une importante étape d'étude et d'implantation, j'ai pu valider le bon fonctionnement de la maquette CoreFlows, et comparer les méthodes numériques proposées (schéma équilibre et schéma précis à bas Mach).

Ce rapport est composé de cinq parties principales, la première décrit le contexte du stage et la structure d'accueil (CEA). Dans la deuxième, des notions de modélisation mathématiques des écoulements diphasiques sont brièvement rappelées avec un focus sur les équations de bilan. La troisième partie comporte quelques méthodes et schémas numériques que nous avons étudiés. La quatrième partie décrit l'application CoreFlows en détaillant tous les outils du génie logiciel utilisés. Les différents résultats permettant de valider le travail effectué sont ensuite présentés dans la cinquième partie. Et pour finir, quelques conclusions et perspectives sont proposées.

**Mots Clés :** Écoulements diphasiques - Mécanique des fluides - Modèle de dérive - Simulation numérique - Schéma de Roe - Schéma équilibre - Schéma bas Mach - Maillage décalé - Logiciels Open Source - Programmation orientée objet

# Remerciements

---

Je tiens à remercier mon tuteur, Michael NDJINGA, qui, par sa compétence, son expérience, et ses qualités humaines exceptionnelles, a guidé mon activité tout en préservant mon esprit d'initiative.

J'adresse également mes remerciements au professeur Olivier LAFITTE pour sa disponibilité et son accompagnement pédagogique.

J'aimerais ensuite remercier l'ensemble des professeurs de la MACS et plus particulièrement les deux responsables, Olivier LAFITTE et Emanuel AUDUSSE, pour leurs disponibilités et la qualité des enseignements qu'ils m'ont apportée.

Je tiens à remercier tous les membres du laboratoire LMEC pour leur accueil et leurs conseils, plus particulièrement Anouar MEKKAS. J'exprime également mes remerciements à Pascal OMNES, Adrien BRUNETON, Marc TAJCHMAN et Francis KLOSS pour leur aide précieuse, ainsi que mes collègues stagiaires pour les échanges qu'on a régulièrement eus durant ce stage.

Enfin, je tiens à remercier ma famille pour sa présence, son écoute, et son soutien inconditionnel. Malgré la distance et tant d'années passées loin d'elle, elle reste ma source inépuisable de motivations et de volonté.





# Nomenclature

---

On désigne par l'indice :

- $k = l$  les grandeurs relatives à la phase liquide,
- $k = v$  les grandeurs relatives à la phase vapeur.

## Grandeurs thermodynamiques

Symbole	Signification	Unité (SI)
$\rho_k$	masse volumique	$kg.m^{-3}$
$\alpha_k$	taux d'occupation volumique	1
$e_k$	énergie interne	$J.kg^{-1}$
$E_k = e_k +  u_k ^2/2$	énergie totale	$J.kg^{-1}$
$h_k = e_k + P/\rho_k$	enthalpie spécifique	$J.kg^{-1}$
$H_k = h_k +  u_k ^2/2$	enthalpie totale spécifique	$J.kg^{-1}$
$\bar{\sigma}_k$	tenseur de viscosité	$kg.m^{-1}.s^{-2}$
$h_{k,sat}$	enthalpie spécifique à saturation	$J.kg^{-1}$

## Grandeurs relatives au mélange

Symbole	Signification	Unité
$\rho = \sum \alpha_k \rho_k$	masse volumique	$kg.m^{-3}$
$\vec{u} = \frac{\sum \alpha_k \rho_k \vec{V}_k}{\rho}$	vecteur vitesse de mélange	$m.s^{-1}$
$\vec{U}_r = \vec{u}_v - \vec{u}_l$	vitesse relative	$m.s^{-1}$
$C_v = \frac{\alpha_v \rho_v}{\rho}$	concentration massique en vapeur	1
$e = \frac{\sum \alpha_k \rho_k e_k}{\rho}$	énergie interne	$J.kg^{-1}$
$E = C_v E_v + (1 - C_v) E_l$	énergie totale	$J.kg^{-1}$
$h = \frac{\sum \alpha_k \rho_k h_k}{\rho}$	enthalpie spécifique	$J.kg^{-1}$
$P$	pression	$Pa$





# Sommaire

---

Résumé	i
Remerciements	iii
Nomenclature	v
1 Introduction générale	1
2 Le modèle mathématique	9
3 Méthode numérique	19
4 Application : CoreFlows	27
5 Résultats numériques	49
6 Conclusions	69
7 Annexe	71
Bibliographie	85
Table des figures	90



# Introduction générale

---

## 1.1 Contexte du stage

### 1.1.1 Le CEA

Le CEA (Commissariat à l'énergie atomique et aux énergies alternatives) est un établissement public français de recherche scientifique dans 4 grands domaines : les énergies décarbonées, la défense et la sécurité globale, les technologies pour l'information, et les technologies pour la santé.

Il est organisé en cinq grandes directions :

- Direction de l'énergie nucléaire (DEN)
- Direction des applications militaires (DAM)
- Direction de la recherche technologique (DRT)
- Direction des sciences du vivant (DSV)
- Direction des sciences de la matière (DSM)

Il a été créé le 18 octobre 1945 par le général de Gaulle avec à sa tête, le haut commissaire à l'énergie atomique Frédéric Joliot-Curie. Aujourd'hui, il comprend plus de 15000 salariés répartis sur les différents centres implantés sur toute la France (la figure 1.1 illustre l'emplacement des 10 centres du CEA à travers la France) et répartis en 2 types de centres, d'une part des centres d'études pour les applications militaires et d'autre part, des centres d'études civiles dont fait partie le CEA-Saclay où j'ai effectué mon stage. Il s'agit, par ailleurs, du centre comprenant le plus de personnel.

Le CEA développe de nombreux partenariats avec les autres organismes de recherche, les collectivités locales et les universités. A ce titre, le CEA est partie prenante d'alliances nationales coordonnant la recherche française dans les domaines de l'énergie (ANCRE), des sciences de la vie et de la santé (AVIESAN), des sciences et technologies du numérique (ALLISTENE) et des sciences de l'environnement (AllEnvi).

Acteur majeur de la recherche appliquée, le CEA héberge (centre de Bruyères le Chatel) un des plus puissants supercalculateurs en Europe. Avec ses 11520 processeurs, ses 92160 cœurs de calculs et sa mémoire gigantesque de 360 Tera-octets, le supercalculateur CURIE peut atteindre des vitesses de 2 Petaflop<sup>1</sup>. CURIE fait partie du réseau GENCI<sup>2</sup>

---

1. 1 Petaflop =  $10^{15}$  flops (i.e FLoating point Operation Per Second )

2. GENCI : Grand Équipement National de Calcul Intensif

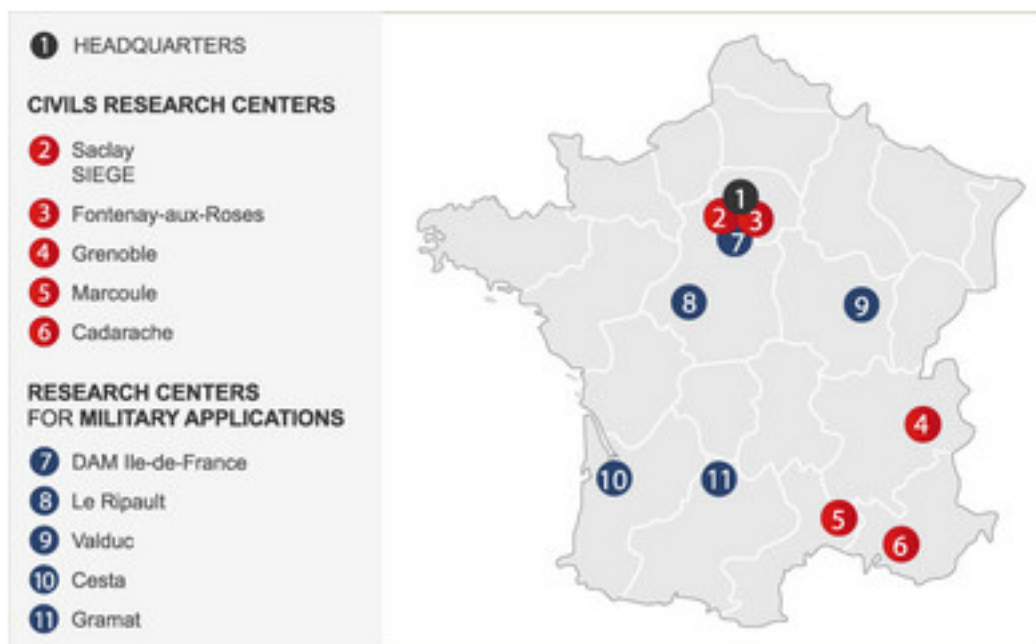


FIGURE 1.1 – Les dix centres du CEA en France

### 1.1.2 Le centre de Saclay

Le centre CEA de Saclay est l'un des plus importants centres de recherche et développement en Europe. Il se situe dans le département de l'Essonne (île de France). Le centre est d'une superficie de 220 ha, implanté sur le plateau de Saclay, dans le pôle scientifique et technologique Paris-Saclay, au sud de Paris. Il est composé d'un domaine principal qui accueille le siège administratif du CEA, et d'une extension : « l'Orme des Merisiers ». Il a été inauguré en 1952, afin de mettre en service le premier réacteur nucléaire de recherche, successeur de la pile Zoé : EL2<sup>3</sup>.

Il comprend près de 400 bâtiments. Près de 8000 personnes circulent tous les jours en son sein, 5400 salariés CEA mais aussi des salariés venant de différents instituts et entreprises.

Le Centre CEA de Saclay accueille également l'INSTN<sup>4</sup>, qui a pour mission l'enseignement et la formation professionnelle dans le domaine du nucléaire, ainsi que nombreux organismes ou entreprises :

- Agence nationale pour la gestion des déchets radioactifs (Andra),
- Institut international de l'énergie nucléaire (I2EN),
- Iba, producteur de molécules radiomarquées pour la radiothérapie,
- Institut de radioprotection et de sûreté nucléaire (IRSN),
- Areva TA (anciennement Technicatome),
- Euriso-top, producteur de solvant contenant du deutérium et de produits marqués,

3. Réacteur à Eau Lourde n°2

4. Institut national des sciences et techniques nucléaires

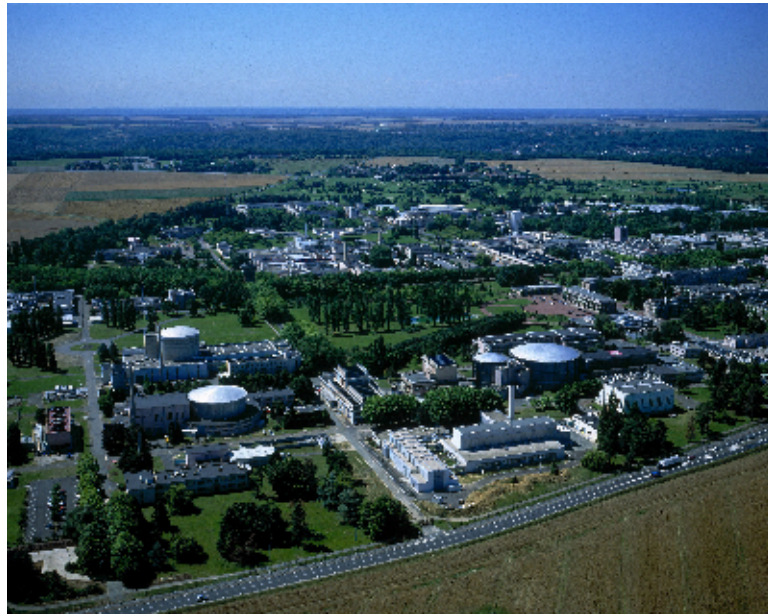


FIGURE 1.2 – Vue du ciel du centre Cea Saclay

CEA Saclay exerce ses activités dans des domaines tels que l'énergie nucléaire, les sciences du vivant, les sciences de la matière, le climat et l'environnement, la recherche technologique et l'enseignement, mais on y trouve aussi des activités de recherches dans les domaines de l'hydrogène et des bioénergies.

Les recherches sur l'énergie nucléaire concernent l'optimisation du fonctionnement des centrales nucléaires françaises actuelles et le développement de systèmes nucléaires innovants.

Ces travaux sont complétés par des recherches sur la gestion des déchets radioactifs ainsi que par le démantèlement des anciens réacteurs nucléaires de recherche.

D'autres axes de recherche sont étudiés, au CEA-Saclay, parallèlement au domaine de l'énergie :

- La recherche technologique : les systèmes informatiques embarqués, les systèmes interactifs (relations homme-machine), les capteurs et le traitement du signal.
- La recherche dans le domaine de la santé : l'effet des rayonnements sur les cellules et les molécules, l'ingénierie des protéines, les recherches en imagerie médicale et les dosages radioimmunologiques.
- Les études sur l'environnement : la modélisation du climat et l'effet de serre.

### 1.1.3 Laboratoire de modélisation à l'échelle composante

Le stage a été effectué au laboratoire LMEC, qui fait partie du "Service de Thermo-hydraulique et de Mécanique des Fluides" (STMF), qui comprend environ 130 salariés, STMF fait partie du "Département de Modélisation des Systèmes et Structures" (DM2S), qui comprend 300 salariés, qui est lui même fait partie de la "Direction déléguée aux activités Nucléaires de Saclay", au sein de la "Direction de l'Energie Nucléaire" (DEN), qui comprend environ 5000 salariés

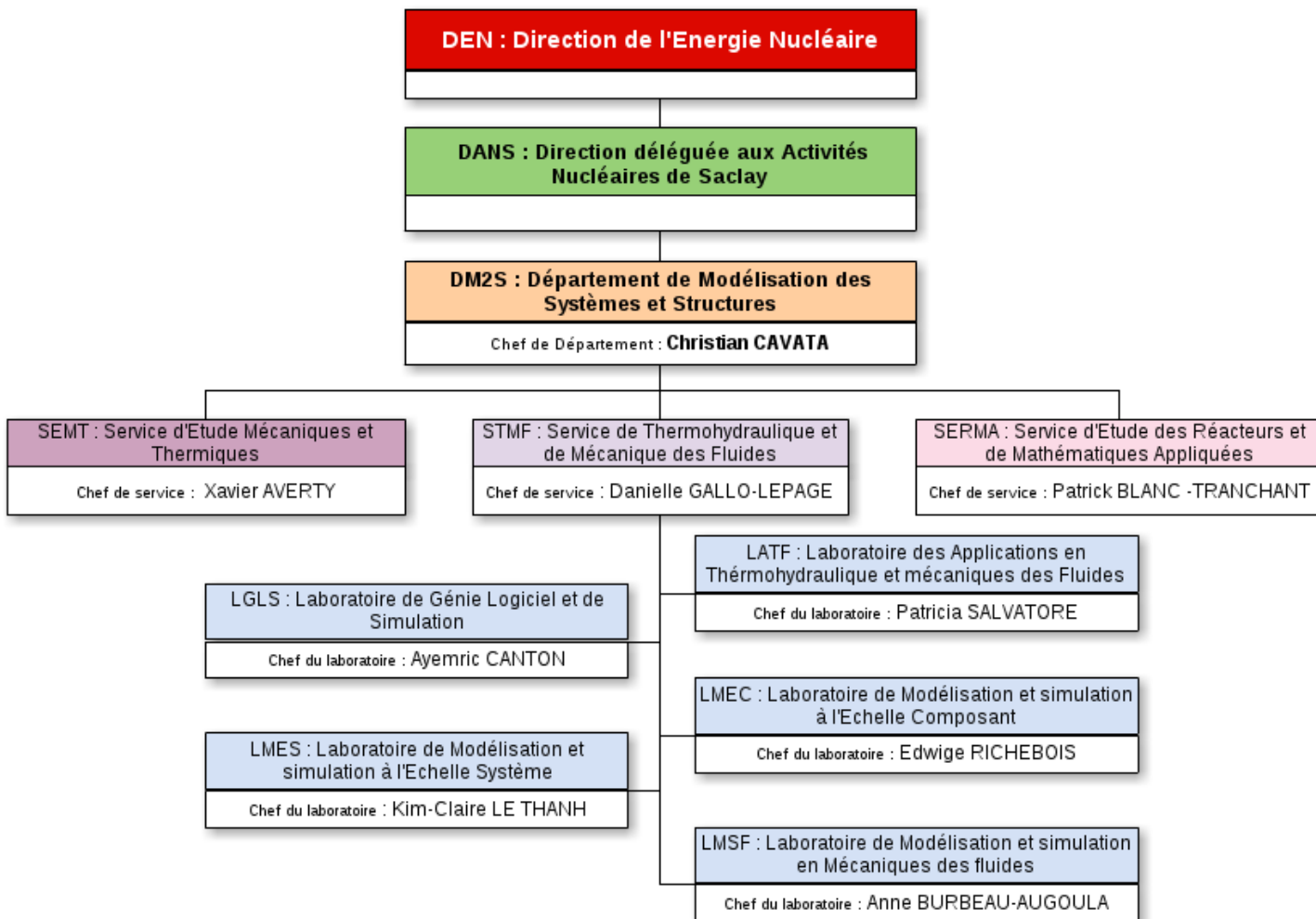


FIGURE 1.3 – Organigramme du CEA

La vingtaine de membres du laboratoire organisés en équipes travaillent sur plusieurs volets : modélisation, développement, simulation, et maintenance de codes industriels. Les compétences se situent essentiellement dans les domaines suivant :

- Modélisation physique des écoulements diphasiques,
- Analyse numérique
- Aide à la conception à l'analyse de sûreté des réacteurs,
- Génie logiciel

## 1.2 Problématique du stage

Parmi les activités du CEA, les technologies liées à la production d'énergie nucléaire sont un élément essentiel qui fait partie des missions originelles du CEA.

**Le principe de fonctionnement des centrales nucléaires REP :** La France compte 19 centrales nucléaires, dans lesquelles l'énergie est produite par des "réacteurs\* à eau pressurisée". L'expression "à eau pressurisée" vient du fait que l'eau qui transporte la chaleur, dans le circuit primaire (voir ci-dessous), est sous très forte pression : 155 bars, c'est-à-dire 155 fois la pression atmosphérique.

Et donc leur principe de fonctionnement est le suivant :

1. Une réaction nucléaire se produit dans le coeur du réacteur et dégage une chaleur très importante
2. La chaleur ainsi produite est transférée à l'eau du circuit primaire (du schéma 1.4). Cette eau est à 300°C environ.
3. A son tour, l'eau du circuit primaire transmet sa chaleur à celle du circuit secondaire ( du schéma 1.4) et la fait bouillir, autrement dit la transforme en vapeur.
4. Cette vapeur fait tourner le groupe turbo-alternateur qui produit l'électricité.
5. La vapeur est refroidie par l'eau du circuit de refroidissement ou circuit tertiaire : elle est condensée, autrement dit elle redevient liquide.

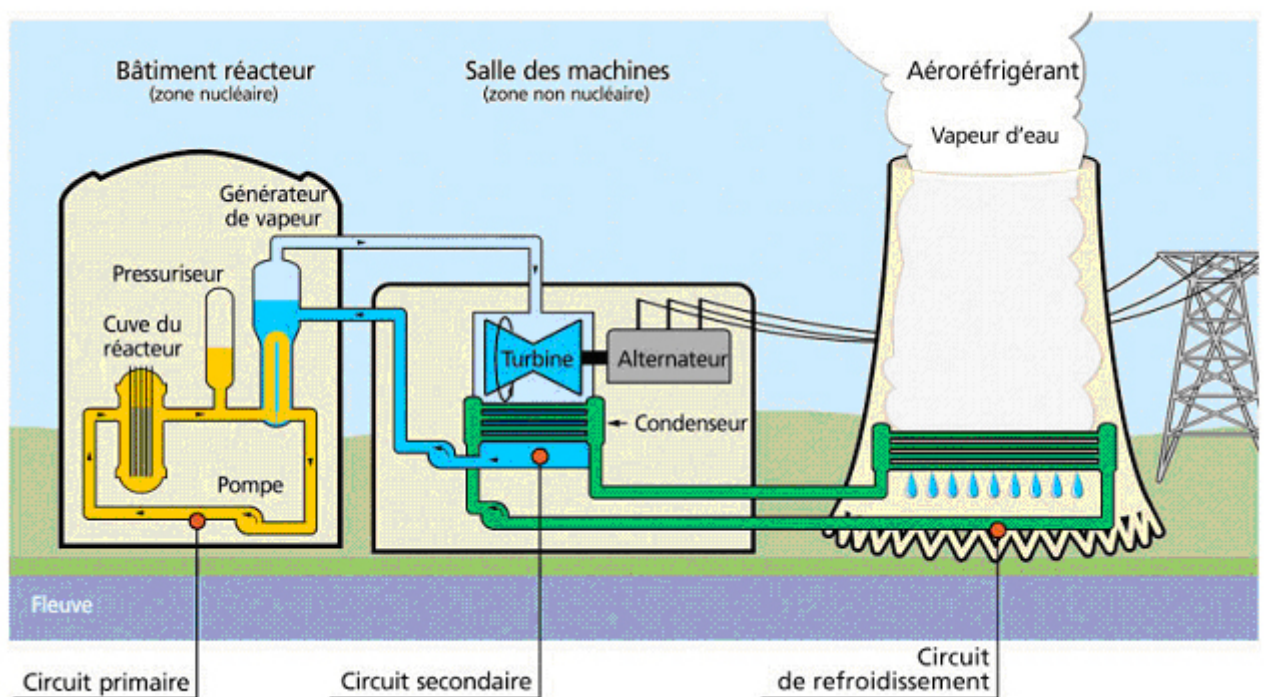


FIGURE 1.4 – Schéma de principe d'une centrale à eau pressurisée

Le risque des centrales nucléaires provient généralement du combustible utilisé, l'uranium.

L'uranium n'est pas classé parmi les éléments les plus toxiques du point de vue de sa radioactivité, mais, lors de la réaction de fission nucléaire qu'il subit dans le cœur, il se transforme en différentes substances qui elles sont très radioactives.

De plus, cette réaction doit être très soigneusement contrôlée sous peine de s'emballer, ce qui pourrait mener à une fusion catastrophique du cœur. Ainsi, il est important pour l'étude du comportement d'un réacteur en régime nominal (amélioration des performances) ou en régime accidentel (étude de sûreté) de pouvoir simuler les écoulements de fluides au sein du réacteur.

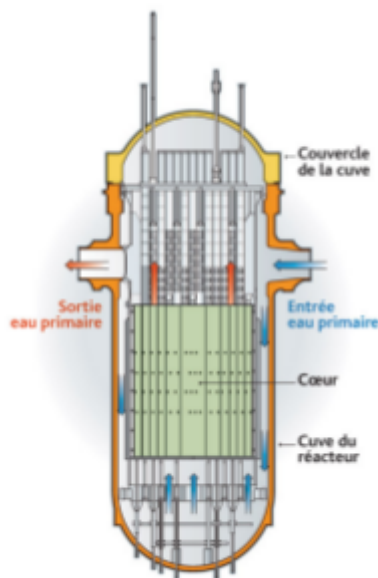


FIGURE 1.5 – Schéma d'une cuve de REP 900

Le STMF développe des modèles et méthodes numériques dédiés à l'étude des écoulements dans les circuits primaire et secondaire, l'enceinte de réacteurs nucléaires, ainsi que l'expérimentation associée.

Ces écoulements comportent en général plusieurs phases, et il importe de pouvoir comprendre et simuler leur dynamique avec précision.

Dans le cadre de la recherche en physique des réacteurs, le LMEC propose des modélisations et des codes numériques de thermohydraulique diphasique (modèles fins et schémas de calcul) afin de simuler le comportement statique et dynamique des cœurs des systèmes nucléaires qui requièrent une bonne connaissance des écoulements diphasiques<sup>6</sup>

Le code **FLICA 4** développé par le LMEC est dédié aux calculs d'écoulements diphasiques (eau liquide et vapeur) pour des régimes permanents ou transitoires dans le cœur du REP. C'est un code de thermohydraulique tridimensionnel utilisant la méthode

6. voir la définition d'un écoulement diphasique dans l'introduction du chapitre 6



des volumes finis. L'écoulement diphasique comprenant une phase liquide et une phase vapeur est modélisé par quatre équations de conservation (bilan de la masse de la phase vapeur et conservation de la masse totale, de l'énergie et de la quantité de mouvement du mélange liquide-vapeur), avec un jeu de lois de fermeture.

Dans le code de calcul **FLICA 4**, le modèle à quatre équations est résolu par la méthode numérique VFRoe [2] sur un maillage colocalisé avec un correctif pour améliorer la précision à faible nombre de Mach. Cette correction permet de mieux capturer la dynamique des écoulements quasi incompressibles mais introduit des instabilités numériques. Ces instabilités rendent certains calculs difficiles à mener, c'est le cas des écoulements bouillants dans les canaux parallèles chauffés différentiellement. C'est sur cette configuration que nous allons tester deux nouvelles approches numériques. Le premier schéma consiste à utiliser un décentrement des termes sources afin d'obtenir un schéma équilibre. Le deuxième schéma est inspiré des schémas sur maillage colocalisé et est précis à faible nombre de Mach.



# Le modèle mathématique

---

Un écoulement diphasique est un écoulement où le fluide peut se trouver dans deux phases différentes. Par exemple, on peut étudier un écoulement où de l'eau se trouve à la fois sous forme liquide et gazeuse.

Par définition, un écoulement diphasique représente un mélange de deux phases ou la topologie, la composition et les phénomènes d'échanges sont des paramètres qui peuvent constamment varier. De par leur grande diversité, on les rencontre fréquemment dans les applications industrielles : réacteur nucléaire, turbo-propulseur, injecteur diphasique,...

Il est difficile de trouver un bon modèle mathématique qui gouverne un système diphasique, ce qui rend ce type de problème physique complexe. Ces modèles mathématiques sont basés à la fois sur des lois universelles (principes de conservation de la masse, de la quantité de mouvement,...) et sur des lois de comportement qui servent essentiellement à fermer le système d'équations. Même s'il est toujours possible d'écrire des équations locales, la résolution directe semble quasiment inabordable à cause des petites échelles du problème et la modélisation des écoulements rencontrés dans l'industrie nécessite une approche de type homogénéisation [1].

Les applications des systèmes diphasiques sont nombreuses. Les écoulements diphasiques apparaissent par exemple dans un grand nombre de systèmes industriels (génie nucléaire, génie pétrolier, génie chimique, propulsion automobile, aéronautique et spatiale,...), en biologie, en chimie (lubrification, distillation,...).

Il existe deux grandes classes de méthodes de modélisation des écoulements diphasiques : les méthodes Lagrangiennes et les méthodes Eulériennes. Dans les méthodes Eulériennes choisies au STMF, la modélisation des écoulements diphasiques est basée sur un procédé d'homogénéisation [1] [3]. Ces techniques consistent à considérer les deux phases comme deux milieux continus distincts se partageant le volume disponible et à résoudre des équations qui gouvernent les variables d'état moyennes de chacune des phases. Sous leur forme la plus générale, elles aboutissent à des systèmes ouverts, caractérisés par deux vitesses et deux pressions distinctes. Ainsi en une dimension d'espace, ces modèles se composent d'au moins six équations ; deux équations de bilan de la masse, deux équations de quantité de mouvement, deux équations d'énergie auxquelles il faut rajouter des relations de fermeture.

## 2.1 Équations de bilan Euleriennes monophasiques

Dans la suite on notera  $k = v, l$  les indices respectifs de chaque phase. Tout milieu diphasique peut être assimilé à un ensemble de régions purement monophasiques, dans lesquelles les équations locales de la mécanique des fluides s'appliquent, ces régions sont séparées par des interfaces infiniment minces et supposées sans masse.

On écrit les bilans de masses  $\rho_k$ , de quantités de mouvement  $\rho_k \mathbf{u}_k$  et d'énergies totales  $\rho_k E_k = \rho_k e_k + \rho_k \frac{\mathbf{u}_k^2}{2}$  volumiques avec  $e_k(T_k)$  les énergies internes spécifiques dépendant des températures  $T_k$ .

On obtient ainsi pour chaque phase, le système classique des équations de Navier-Stokes :

$$\begin{aligned} \frac{\partial \rho_k}{\partial t} + \operatorname{div}(\rho_k \mathbf{u}_k) &= 0 \\ \frac{\partial \rho_k \mathbf{u}_k}{\partial t} + \operatorname{div}(\rho_k \mathbf{u}_k \otimes \mathbf{u}_k) + \nabla p_k + \operatorname{div}(-\bar{\tau}_k) &= \rho_k \mathbf{f}_k^v \\ \frac{\partial \rho_k E_k}{\partial t} + \operatorname{div}(\rho_k E_k + p_k) \mathbf{u}_k + \operatorname{div}(\mathbf{q}_k - \bar{\tau}_k \cdot \mathbf{u}_k) &= \rho_k \mathbf{f}_k^v \cdot \mathbf{u}_k \end{aligned} \quad (2.1)$$

où  $\bar{\tau}_k$  est le tenseur des contraintes visqueuses,  $\mathbf{f}_k^v$  l'ensemble des forces volumiques extérieures s'appliquant sur la phase  $k$  et  $\mathbf{q}_k$  le flux de chaleur.

## 2.2 Équations de bilan moyennées

Pour décrire l'évolution des variables dans les deux domaines fluides, on note  $\Omega_k(t)$  le domaine spatial occupé par la phase  $k$  dans le domaine de l'écoulement à l'instant  $t$ , et on définit  $X_k$  la fonction caractéristique de chaque phase par :

$$X_k(x, t) = \begin{cases} 1 & \text{si } x \in \omega_k(t) \\ 0 & \text{sinon} \end{cases}, \quad (2.2)$$

et nous avons  $\sum_k X_k(x, t) = 1$ . On note  $W$  la vitesse de l'interface. L'équation qui gouverne l'évolution de  $X_k$  est :

$$\frac{\partial}{\partial t} X_k = W \cdot n_k \cdot \delta, \quad (2.3)$$

où  $n_k$  est la normale unitaire à l'interface dirigée du fluide  $k$  vers le fluide  $\bar{k}^1$  et  $\delta$  est la distribution de Dirac concentrée sur l'interface.

D'autre part, nous avons également :

$$\nabla X_k = -n_k \delta \quad (2.4)$$

ce qui nous permet d'écrire 2.3 comme

$$\frac{\partial}{\partial t} X_k + W \cdot \nabla X_k = 0 \quad (2.5)$$

---

1. si  $k = g$  alors  $\bar{k} = l$  sinon  $\bar{k} = g$

Rappelons aussi que la fonction  $X_k$  vérifie au sens des distributions les identités suivantes :

$$\begin{aligned}\nabla(X_k f) &= X_k \nabla f + f \nabla X_k = X_k \nabla f - f n_k \delta \\ \operatorname{div}(X_k \mathbf{u}) &= X_k \operatorname{div} \mathbf{u} + \mathbf{u} \nabla X_k = X_k \operatorname{div} \mathbf{u} - \mathbf{u} n_k \delta \\ \operatorname{div}(X_k \tau) &= X_k \operatorname{div} \tau + \tau \nabla X_k = X_k \operatorname{div} \tau - \tau n_k \delta\end{aligned}$$

où  $f$  est une fonction scalaire,  $\mathbf{u}$  est un vecteur,  $\tau$  un tenseur du 2<sup>ème</sup> ordre.

Les équations 2.1 et 2.5 fournissent une description complète et très détaillée de l'écoulement. Toutefois elles ne peuvent pas être la base d'une méthode numérique quand les pas d'espace sont beaucoup plus grands que la taille des inclusions liquides et gazeuses. Pour traiter ces cas, on doit considérer l'utilisation de modèles homogénéisés.

Pour établir le modèle diphasique à 6 équations ou le modèle de dérive à 4 équations de **FLICA 4** [15], nous employons les méthodes de prise de moyenne décrites dans [4] ou [5], comme annoncé dans l'introduction de ce chapitre. Le procédé d'homogénéisation est donné dans [7] auquel il faut rajouter les conditions d'interface moyennées [7], et en supposant aussi que le tenseur des contraintes visqueuses, le flux de chaleur et l'ensemble des forces volumiques extérieures sont égaux à leurs valeurs moyennes. On obtient donc le système d'équations suivant :

$$\begin{aligned}\frac{\partial}{\partial t}(\alpha_k \rho_k) + \operatorname{div}(\alpha_k \rho_k \mathbf{u}_k) &= \Gamma_k \\ \frac{\partial}{\partial t}(\alpha_k \rho_k \mathbf{u}_k) + \operatorname{div}(\alpha_k \rho_k \mathbf{u}_k \otimes \mathbf{u}_k) + \nabla(\alpha_k p_k) + \operatorname{div}(-\alpha_k \bar{\tau}_k) \\ &= \mathbf{M}_k^\Gamma + p_{kI} \nabla \alpha_k + \mathbf{F}_k^d + \alpha_k \rho_k \mathbf{f}_k^v \\ \frac{\partial}{\partial t}(\alpha_k \rho_k E_k) + \operatorname{div}(\alpha_k \rho_k E_k + \alpha_k p_k) \mathbf{u}_k + \operatorname{div}(\alpha_k (\mathbf{q}_k - \bar{\tau}_k \cdot \mathbf{u}_k)) \\ &= \mathbf{H}_k^\Gamma - p_{kI} \frac{\partial \alpha_k}{\partial t} + \mathbf{F}_k^d \cdot \mathbf{u}_{kI} + Q_{kI} + \alpha_k \rho_k \mathbf{f}_k^v \cdot \mathbf{u}_k\end{aligned}\tag{2.6}$$

avec :

- $\Gamma_k, \mathbf{M}_k^\Gamma, \mathbf{H}_k^\Gamma$  sont des termes de transfert interfacial de masse,
- $Q_{kI}$  est le terme de transfert interfacial de chaleur,
- $\mathbf{F}_k^d$  est le terme de force de frottement interfacial,
- $p_{kI}$  et  $\mathbf{u}_{kI}$  sont respectivement la pression et la vitesse de la phase  $k$  à l'interface,

L'écriture 2.6 correspond à la forme générale des modèles bi-fluides obtenue par la procédure de moyenne . Néanmoins dans ce système, on compte seulement 6 équations pour 7 inconnues  $\rho_k$ ,  $\mathbf{u}_k$ ,  $E_k$  et  $\alpha_k$ . Ce système est ouvert et il faut se donner des relations de fermeture.

Dans les modèles bi-fluides classiques à six équations, ce problème est résolu par l'existence d'une relation entre les pressions  $p_v$  et  $p_l$ . L'hypothèse des pressions communes est très

répandue [4]. En effet il est physiquement légitime de supposer l'égalité des pressions phasiques :

$$p_l = p_v = p_{vI} = p_{lI} = p$$

Et ainsi des lois constitutives de la forme  $e_k = e_k(T_k)$  pour les énergies internes phasiques.

Et donc le modèle bi-fluide classiques à six équations s'écrit sous la forme :

$$\begin{aligned} \frac{\partial}{\partial t}(\alpha_k \rho_k) + \text{div}(\alpha_k \rho_k \mathbf{u}_k) &= \Gamma_k \\ \frac{\partial}{\partial t}(\alpha_k \rho_k \mathbf{u}_k) + \text{div}(\alpha_k \rho_k \mathbf{u}_k \otimes \mathbf{u}_k) + \alpha_k \nabla(p) + \text{div}(-\alpha_k \bar{\tau}_k) \\ &= \mathbf{M}_k^\Gamma + p \nabla \alpha_k + \mathbf{F}_k^d + \alpha_k \rho_k \mathbf{f}_k^v \\ \frac{\partial}{\partial t}(\alpha_k \rho_k E_k) + \text{div}(\alpha_k \rho_k E_k + \alpha_k p) \mathbf{u}_k + \text{div}(\alpha_k (\mathbf{q}_k - \bar{\tau}_k \cdot \mathbf{u}_k)) \\ &= \mathbf{H}_k^\Gamma - p \frac{\partial \alpha_k}{\partial t} + \mathbf{F}_k^d \cdot \mathbf{u}_{kI} + Q_{kI} + \alpha_k \rho_k \mathbf{f}_k^v \cdot \mathbf{u}_k \end{aligned} \quad (2.7)$$

## 2.3 Obtention des équations du modèle de dérive

En faisant des hypothèses simplificatrices sur les vitesses et températures des phases en présence, il est possible de passer d'un modèle à 6 équations à un modèle à 4 équations. L'écoulement diphasique comprenant une phase liquide et une phase vapeur est modélisé par quatre équations de bilan :

- Conservation de la masse totale de mélange
- Bilan de la masse de vapeur
- Bilan de la quantité de mouvement de mélange
- Bilan de l'énergie de mélange

Ce modèle dit de dérive à 4 équations est basé sur des hypothèses simplificatrices assez fortes.

- Premièrement l'écart de vitesse entre phases est régi par une corrélation *a priori*  $\mathbf{u}_r = \mathbf{u}_v - \mathbf{u}_l = f_r(c_v, \mathbf{u}_m, \rho_m)$ .

Cette hypothèse limite l'utilisation du modèle aux faibles écarts de vitesses entre les deux phases et la plupart des corrélations supposent des écoulements co-courants.

- Deuxièmement la vapeur est supposée à saturation :  $T_v = T_v^{sat}$ .

Cette hypothèse empêche entre autres la simulation de vapeur surchauffée.

### 2.3.1 Le modèle de dérive de **FLICA 4**

**FLICA 4** est un logiciel de thermo-hydraulique diphasique 3D développé au Service STMF du CEA. Il est dédié aux calculs d'écoulements diphasiques (eau liquide et vapeur)

au sein des coeurs de réacteurs nucléaires, dans des régimes permanents ou transitoires. Il tient compte de la présence d'obstacles solides au sein de l'écoulement à travers une approche poreuse.

Le modèle mathématique de **FLICA 4** est un modèle à 4 équations qui est établi à partir du modèle à 6 équations (cf. 2.7), en incluant la porosité  $\phi$ , en supprimant deux équations aux dérivées partielles et en les remplaçant par deux équations algébriques donnant l'écart de vitesse entre phases et le déséquilibre thermodynamique.

Un modèle Drift est utilisé pour prendre en compte le glissement entre les deux phases (vapeur et liquide), par conséquent donne l'écart de vitesse  $\mathbf{u}_r = \mathbf{u}_v - \mathbf{u}_l = f_r(c_v, \mathbf{u}_m, \rho_m)$ . Les deux phases sont compressibles et supposées à la même pression. Le déséquilibre thermodynamique entre phases est obtenu en supposant qu'une des deux phases (vapeur dans notre cas) est à saturation.

### L'équation de conservation de masse du mélange

Elle s'obtient en additionnant les équations de bilan de masse liquide et vapeur (2.7). La linéarité des opérateurs présents dans ces deux équations permet l'écriture de l'équation suivante :

$$\frac{\partial}{\partial t} \phi(\rho_l \alpha_l + \rho_v \alpha_v) + \text{div} \phi(\rho_l \alpha_l \mathbf{u}_l + \rho_v \alpha_v \mathbf{u}_v) = 0$$

L'indice  $l$  représente la phase liquide et l'indice  $v$  la phase vapeur

### L'équation de bilan de la masse de vapeur

$$\frac{\partial}{\partial t} (\phi \rho_v \alpha_v) + \text{div} \phi (\rho_v \alpha_v \mathbf{u}_v + K_{cv} \text{grad} c_v) = \phi \Gamma_v$$

$\Gamma_v$  : masse de vapeur créé par unité de temps

$K_{cv}$  est un coefficient de diffusion turbulente optionnel

### L'équation de bilan de quantité de mouvement

L'équation de bilan de quantité de mouvement pour le mélange des deux phases s'obtient aussi en additionnant les équations de conservation de la quantité de mouvement liquide et vapeur dans (2.7) et s'écrit :

$$\frac{\partial}{\partial t} \phi(\rho_v \alpha_v \mathbf{u}_v + \rho_l \alpha_l \mathbf{u}_l) + \text{div} \phi(\sum_{k=v,l} \rho_k \alpha_k \mathbf{u}_k \otimes \mathbf{u}_k + \alpha_k p \mathbf{I} + \alpha_k \Pi_k) = \phi \rho \vec{g} + \phi \vec{\tau}$$

$\mathbf{I}$  : est la matrice identité de dimension de  $\mathbf{u}$

$\Pi_k$  : est le tenseur des contraintes visqueuses et turbulentes de la phase  $k$

## L'équation de bilan de l'énergie

L'équation de bilan de l'énergie s'écrit :

$$\begin{aligned} \frac{\partial}{\partial t}(\phi \sum_{k=v,l} \rho_k \alpha_k E_k) + \text{div} \phi (\sum_{k=v,l} \rho_k \alpha_k \mathbf{u}_k E_k - \alpha_k \Pi_k \mathbf{u}_k - q_k) \\ = Q_{ktot} + \phi \left( \sum_{k=v,l} \rho_k \alpha_k \mathbf{u}_k \right) \cdot \vec{g} \end{aligned}$$

$Q_{ktot}$  : est le terme source de puissance volumique totale reçue par la phase  $k$ .

$q_k$  : est le flux thermique généré par la conduction thermique de la phase  $k$

Et donc le modèle **FLICA 4** à quatre équations s'écrit sous la forme :

$$\begin{aligned} \frac{\partial}{\partial t} \phi (\rho_l \alpha_l + \rho_v \alpha_v) &+ \text{div} \phi (\rho_l \alpha_l \mathbf{u}_l + \rho_v \alpha_v \mathbf{u}_v) &= 0 \\ \frac{\partial}{\partial t} (\phi \rho_v \alpha_v) &+ \text{div} \phi (\rho_v \alpha_v \mathbf{u}_v + K_{cv} \text{grad} cv) &= \phi \Gamma_v \\ \frac{\partial}{\partial t} \phi (\rho_v \alpha_v \mathbf{u}_v + \rho_l \alpha_l \mathbf{u}_l) &+ \text{div} \phi (\sum_{k=v,l} \rho_k \alpha_k \mathbf{u}_k \otimes \mathbf{u}_k + \alpha_k p \mathbb{I} + \alpha_k \Pi_k) &= \phi \rho \vec{g} + \phi \vec{\tau} \\ \frac{\partial}{\partial t} (\phi \sum_{k=v,l} \rho_k \alpha_k E_k) &+ \text{div} \phi (\sum_{k=v,l} \rho_k \alpha_k \mathbf{u}_k E_k - \alpha_k \Pi_k \mathbf{u}_k - q_k) \\ &= Q_{tot} + \phi \left( \sum_{k=v,l} \rho_k \alpha_k \mathbf{u}_k \right) \cdot \vec{g} \end{aligned} \quad (2.8)$$

## Termes source du modèle **FLICA 4**

**Équation de masse totale** La première composante est nulle parce qu'il n'y a pas de création de matière mais seulement transformation du liquide en vapeur.

**Équation de masse vapeur**  $\Gamma_v$  est le taux de changement de phase : quantité de liquide vaporisée par unité de temps, qui est la somme de la vaporisation au contact des parois chauffantes  $\Gamma_{wv}$  et de l'échange de masse à l'interface entre phases  $\Gamma_{lv}$

Le terme de changement de phase  $\Gamma_v$  est non nul si l'enthalpie  $h_m$  du fluide se situe entre l'enthalpie du liquide et celle de la vapeur. Cependant **FLICA 4** modélise aussi le phénomène d'ébullition sous-saturée où les bulles apparaissent uniquement au contact des parois chaudes alors que la température moyenne dans le canal reste inférieure à la température d'ébullition.

**Équation de quantité de mouvement** Le terme source  $\tau$  représente la somme des forces de frottement de mélange sur les parois  $\tau_w$  et sur les obstacles singuliers  $\tau_s$  (les obstacles solides présents dans le réacteur).

Le terme de frottement pariétal est linéarisé sous la forme  $-K \rho_m \mathbf{u}_m$  avec  $K$  une constante dépendante en principe du diamètre hydraulique et du nombre de Reynolds. On ne considérera pas ici le coefficient de paroi chauffante ni le multiplicateur diphasique.



**Équation de l'énergie** La source d'énergie fournie au fluide est décrite par le terme  $Q_{tot}$ , la puissance volumique totale reçue par le fluide au contact des parois chauffantes.

### 2.3.2 Le modèle de dérive dans CoreFlows

Afin de reproduire les difficultés numériques de **FLICA 4** nous allons développer une maquette du code basée sur une version simplifiée du modèle et permettant l'étude de nouvelles approches numériques.

Nous considérons des termes sources simplifiés de changement de phase, frottement pariétal et chauffage. Plutôt que de supposer que la vapeur est à saturation nous supposons l'équilibre thermique :  $T_v = T_l$ .

Cette hypothèse nous a paru plus simple et plus générale que l'hypothèse de saturation de la vapeur. Pour le moment, nous ne prenons pas en compte la vitesse relative  $u_r$ .

Nous nous intéressons donc à un système de quatre équations aux dérivées partielles non linéaires qui prennent la forme conservative suivante :

$$\partial_t U + \nabla \cdot F(U) = S, \quad (2.9)$$

avec les expressions suivantes pour le vecteur des variables conservatives  $U$  :

$$U = \begin{pmatrix} \alpha_v \rho_v + \alpha_l \rho_l \\ \alpha_v \rho_v \\ \alpha_v \rho_v \mathbf{u}_v + \alpha_l \rho_l \mathbf{u}_l \\ \alpha_v \rho_v E_v + \alpha_l \rho_l E_l \end{pmatrix},$$

et la matrice de flux  $F(U) = F_{NV}(U) + F_V(U)$

$$F_{NV}(U) = {}^t \begin{pmatrix} \alpha_v \rho_v {}^t \mathbf{u}_v + \alpha_l \rho_l {}^t \mathbf{u}_l \\ \alpha_v \rho_v {}^t \mathbf{u}_v \\ \alpha_v \rho_v \mathbf{u}_v \otimes \mathbf{u}_v + \alpha_l \rho_l \mathbf{u}_l \otimes \mathbf{u}_l + p \mathbb{I}_d \\ \alpha_v \rho_v H_v {}^t \mathbf{u}_v + \alpha_l \rho_l H_l {}^t \mathbf{u}_l \end{pmatrix}, \quad (2.10)$$

$$F_V(U) = {}^t \begin{pmatrix} 0 \\ 0 \\ \alpha_v \mu_v \vec{\nabla} \mathbf{u}_v + \alpha_l \mu_l \vec{\nabla} \mathbf{u}_l \\ \alpha_v \mu_v \mathbf{u}_v \cdot \vec{\nabla} \mathbf{u}_v + \alpha_l \mu_l \mathbf{u}_l \cdot \vec{\nabla} \mathbf{u}_l + \alpha_v \lambda_v \vec{\nabla} T + \alpha_l \lambda_l \vec{\nabla} T \end{pmatrix}. \quad (2.11)$$

Avec pour chaque phase :

la viscosité  $\mu_k$ ,  $k = l, v$

La conductivité  $\lambda_k, k = l, v$

L'énergie totale :  $E_k = e_k + \frac{1}{2}|\mathbf{u}_k|^2, k = l, v$

L'enthalpie totale :  $H_k = h_k + \frac{1}{2}|\mathbf{u}_k|^2, k = l, v$

Où  $e_k$  est l'énergie interne, et  $h_k = e_k + \frac{p}{\rho_k}$  l'enthalpie associées à la phase  $k$

s

## Lois d'état et paramètres constitutifs

Pour chaque phase la loi d'état est approchée par une loi des gaz raidis

$$p_k = (\gamma_k - 1)\rho_k e_k - \gamma_k p_{0k}$$

et une loi d'énergie interne linéarisées

$$e_k(T) = e_k(345K) + c_{vk}(T - 345K)$$

valables au voisinage du point de saturation  $p_0 = 155bars, T_0^{sat} = 345$ .

Les valeurs de densité, d'énergie interne et de capacité calorifique à volume constant, de viscosité et de conductivité pour chaque phase autour du point  $(p_0, T_0)$  sont issues de la base NIST [20].

## Termes sources

Nous considérons un terme source simplifié de la forme

$$S = \begin{pmatrix} 0 \\ \Gamma_v(c_v, T) \\ \rho_m \vec{g} - K \rho_m ||\mathbf{u}_m|| \mathbf{u}_m \\ \Phi + \rho_m \vec{g} \cdot \mathbf{u}_m - K \rho_m ||\mathbf{u}_m||^3 \end{pmatrix},$$

Avec

$$\begin{aligned} \rho_m &= \alpha_l \rho_l + \alpha_v \rho_v \\ \mathbf{u}_m &= \frac{\alpha_l \rho_l \mathbf{u}_l + \alpha_v \rho_v \mathbf{u}_v}{\alpha_l \rho_l + \alpha_v \rho_v} \\ h_m &= \frac{\alpha_l \rho_l h_l + \alpha_v \rho_v h_v}{\alpha_l \rho_l + \alpha_v \rho_v} \end{aligned}$$

La première composante est nulle parce qu'il n'y a pas de création de matière mais seulement transformation du liquide en vapeur.

$\Gamma_v$  est le taux de changement de phase : quantité de liquide vaporisée par unité de temps.

$\rho_m \vec{g} - K \rho_m ||\mathbf{u}_m|| \mathbf{u}_m$  représente les forces volumiques appliquées au fluide.  $\vec{g}$  représente la gravité et  $-K \rho_m ||\mathbf{u}_m|| \mathbf{u}_m$  le frottement sur les obstacles solides présents dans le réacteur.

$\Phi$  représenta la puissance thermique transférée au liquide par unité de temps.

Le terme de changement de phase  $\Gamma_v$  est non nul si l'enthalpie  $h_m$  du fluide se situe entre l'enthalpie du liquide et celle de la vapeur prises au point de saturation  $(p_0, T_0)$ . Dans ce cas toute la chaleur reçue par le flux thermique sert au changement de phase :

$$\Gamma_v = \begin{cases} \frac{\Phi}{\mathcal{L}} & \text{si } h_l^0 \leq h < h_v^0 \text{ et } c_v < 1 \\ 0 & \text{sinon} \end{cases} . \quad (2.12)$$

Si la puissance thermique est nulle,  $\Gamma_v = 0$  et il n'y a pas de changement de phase. Nous ne modélisons donc pas une ébullition/condensation spontanée du fluide, et les phases ne sont jamais supposées à saturation.

Le terme de frottement pariétal est linéarisée sous la forme  $-K\rho_m||\mathbf{u}_m||\mathbf{u}_m$  avec  $K$  une constante dépendante en principe du diamètre hydraulique et du nombre de Reynolds. On ne considèrera donc pas ici le coefficient de paroi chauffante ni le multiplicateur diphasique.



# Méthode numérique

---

Nous appliquons une méthode de volumes finis à la résolution des équations de bilan modélisant le fluide diphasique.

La solution exacte du problème est approchée par une fonction constante par maille. Ses valeurs sont solutions d'un problème discret obtenu par intégration des équations locales sur chaque maille d'un maillage recouvrant le domaine du calcul.

Les équations de conservation du modèle thermohydraulique s'écrivent sous la forme suivante :

$$\partial_t U + \text{div} \cdot F(U) = S(U) \quad (3.1)$$

où

- $U$  est le vecteur des variables conservatives
- $F(U)$  est le vecteur du flux dans les trois directions. Il se décompose en deux parties :

$$F(U) = F_{NV}(U) + F_V(U)$$

tels que

- $F_{NV}(U)$  correspond aux flux convectifs,
- $F_V(U)$  représente la contribution des flux visqueux,
- $S(U)$  est le vecteur des termes sources.

On précise les notations relatives au maillage :

La trace d'une cellule de contrôle sur un plan horizontal est un quadrangle ou un triangle quelconque.

Les volumes de contrôle ainsi que les variables, constantes dans le volume sont numérotés par l'indice  $i$ .

On utilise les règles suivantes pour les indices :

- $i$  indice d'une cellule de  $C_i$
- $j$  indice de la cellule voisine de  $C_i$
- $ij$  interface entre cellules voisines  $C_i$  et  $C_j$
- $I(i)$  ensemble des indices  $j$  dont les cellules voisines de  $C_j$  sont voisines de  $C_i$
- $v(i)$  ensemble des indices  $j$  tels que la cellule  $C_j$  ait une arête commune avec la cellule  $C_i$

Notons  $Vol(C_i)$  le volume de la cellule de contrôle  $C_i$  et  $\partial C_i$  sa frontière. Notons aussi  $S_{ij}$  la surface de l'interface  $\partial C_i \cap \partial C_j$ .

La formulation volumes finis consiste à associer à chaque cellule de contrôle  $C_i$ , une valeur constante  $U_i$  pour le vecteur des variables conservatives, et à intégrer ensuite les équations de conservation 3.1 sur cette cellule. On note  $\partial\Omega$  la frontière du domaine de calcul  $\Omega$  et  $\mathbf{n} = (n_x, n_y, n_z)$  la normale unitaire sortante de  $\partial\Omega$

La formulation semi-discrète en espace volumes finis de 3.1 est :

$$\int_{C_i} S(U) dv^8 = Vol(C_i) \partial_t U_i + \int_{\partial C_i} F(U) \cdot \mathbf{n} d\sigma^9 \quad (3.2)$$

Pour la discrétisation en temps (schémas explicite et implicite) et le traitement des conditions aux limites nous renvoyons à l'annexe.

### 3.1 Formulation volumes finis de *FLICA 4*

Dans *FLICA 4*, les maillages 3D considérés sont obtenus par une translation suivant la direction axiales (direction  $z$ ) d'un maillage 2D structuré ou non, dans un plan horizontal (plan  $xy$ ).

#### 3.1.1 Flux convectifs

On s'intéresse au calcul des flux convectifs apparaissant dans l'équation 3.2.

La surface de la cellule  $C_i$  est composée de faces polygonales :

$$\int_{\partial C_i} F_{NV}(U) \cdot \mathbf{n} d\sigma = \sum_{j \in I(i)} \int_{\partial C_{ij}} F_{NV}(U) \cdot \mathbf{n} d\sigma + \int_{\partial C_i \cap \partial\Omega} F_{NV}(U) \cdot \mathbf{n} d\sigma \quad (3.3)$$

Dans le premier terme, la somme est étendue à toutes les cellules voisines  $C_j$  ayant une interface commune  $\partial C_{ij} = \partial C_i \cap \partial C_j$  avec la cellule  $C_i$ . Le second terme correspond à l'intégration des termes de bord. Les flux internes, correspondant à une interface strictement intérieure au domaine  $\Omega$  qui sont approchés par :

$$\int_{\partial C_{ij}} F_{NV}(U) \cdot \mathbf{n} d\sigma = S_{ij} \tilde{F}(U_i, U_j, \mathbf{n}_{ij}) \quad (3.4)$$

où :

$\tilde{F}$  est le flux numérique, on l'explicitera plus loin.

Les flux limites correspondant à une interface sur le bord du domaine  $\Omega$ , sont calculés à l'aide de la même fonction de flux numérique, en introduisant une cellule fictive adjacente à la face  $C_{ik}$  et en dehors du domaine de calcul (7.2).

---

8. représente un élément de volume

9. représente un élément de surface

### Schéma VFRoe

**FLICA 4** utilise actuellement un schéma VFRoe avec une correction de pression pour le rendre plus précis. Ces équations sont discrétisées par la méthode des volumes finis. Le domaine de calcul est décomposé en cellules (aussi appelées mailles) qui peuvent être de forme quelconque (maillage non structuré et/ou non conforme). Les versions suivantes de **FLICA 4** utilisent la variante dite VFRoe ([2]), qui consiste à calculer l'état à l'interface ( $U^*$ ) puis prendre le flux physique de cet état :

$$\tilde{F}(U_i, U_j, \mathbf{n}_{ij}) = F(U^*)$$

avec

$$U^*(U_i, U_j, \mathbf{n}_{ij}) = \frac{U_i + U_j}{2} - \text{signe}(A_{ij}) \frac{U_i - U_j}{2}. \quad (3.5)$$

**correction de pression** Dans la variante VFRoe du schéma (équation 3.5), la correction de pression consiste à calculer l'état primitif  $V^*$  associé à  $U^*$  (voir [6]), ensuite remplacer la pression de  $V^*$  par  $\frac{p_i + p_j}{2}$  et enfin calculer un nouvel état conservatif  $U^*$ .

Malheureusement bien que la précision du schéma en soit ainsi augmentée, ces corrections de pression dégradent la stabilité du schéma de Roe originel (oscillations de la solution). (voir figure 3.1)

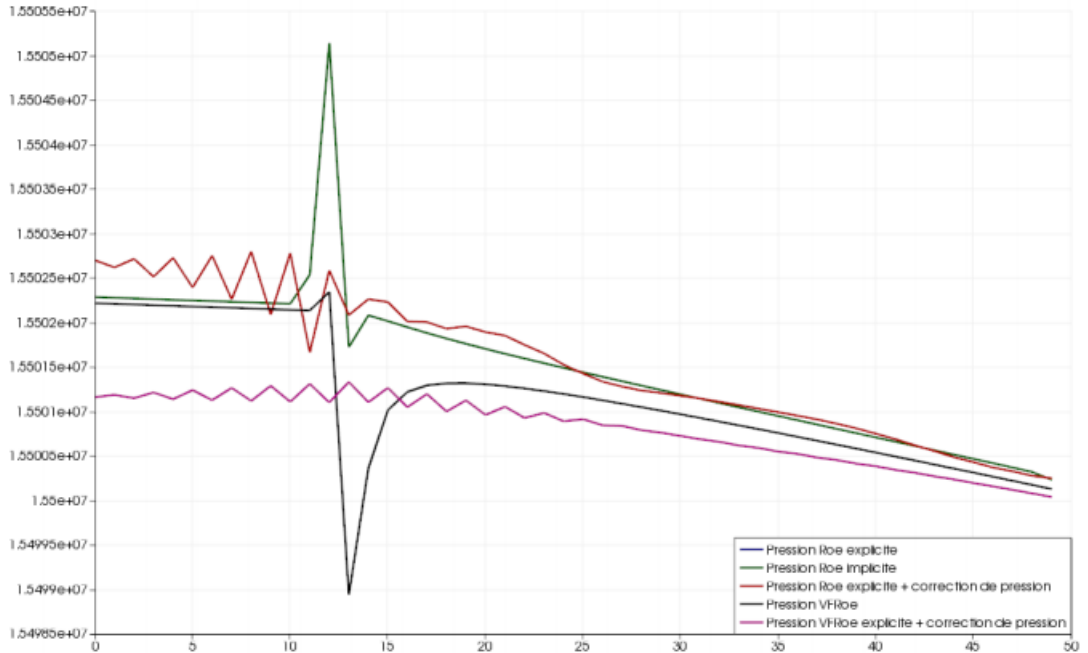


FIGURE 3.1 – Comparaison de schémas numériques : Pression stationnaire  $U_{entree} = 1m/s$

### 3.1.2 Flux visqueux

Nous abordons maintenant le calcul des termes du second ordre associés à la modélisation des contraintes visqueuses du fluide diphasique étudié. La contribution visqueuse du flux total  $F(U)$  est une fonction des variables conservatives  $U$  et du gradient  $\nabla U$  de ces variables. On réécrit les flux visqueux apparaissant dans l'équation 3.2 de la manière suivante :

$$\int_{\partial C_i} (F_V(U)\mathbf{n}_x + G_V(U)\mathbf{n}_y + H_V(U)\mathbf{n}_z)\mathbf{n} \, d\sigma = \int_{\partial C_i} \tilde{F}_V(U, \nabla U, \mathbf{n}) \, d\sigma \quad (3.6)$$

où

$\tilde{F}_V(U, \nabla U, \mathbf{n})$  représente le flux visqueux dans la direction normale à la surface  $\partial C_i$ . Cette dernière expression se décompose en deux termes concernant les interfaces intérieures et sur le bord du domaine  $\Omega$  :

$$\int_{\partial C_i} \tilde{F}_V(U, \nabla U, \mathbf{n}) \, d\sigma = \sum_{j \in I(i)} \int_{\partial C_{ij}} \tilde{F}_V(U, \nabla U, \mathbf{n}) \, d\sigma + \sum_{k \in K(i)} \int_{\partial C_{ik}} \tilde{F}_V(U, \nabla U, \mathbf{n}) \, d\sigma \quad (3.7)$$

Pour calculer les flux visqueux à une interface  $\partial C_{ij}$ , il est nécessaire d'évaluer à cette interface le gradient des variables conservatives. En fait, on remarque que le flux visqueux est une fonction linéaire du gradient de la vitesse et de l'enthalpie de chacune des phases. En maillage structuré, l'évaluation du gradient de ces quantités à une interface est classique à l'aide d'un schéma centré de type différences finies

Dans le cas d'un maillage non structuré, l'évaluation du gradient à l'interface est moins classique. Pour approcher de façon correcte les dérivées partielles (normale et tangentielle) de  $U$  sur une interface en fonction des valeurs au barycentre des mailles, nous introduisons les points suivants (voir Figure 3.2) :

- $N_1$  et  $N_2$  deux points appartenant à l'interface  $C_{ij}$
- $N_G$  et  $N_D$  deux points appartenant à l'interface  $C_{ij}$  passant par le milieu  $M$

On définit alors les dérivées normales et tangentielles de  $U$  par :

$$\frac{U_{N_2} - U_{N_1}}{\|N_1 N_2\|} \text{ et } \frac{U_{N_D} - U_{N_G}}{\|N_D N_G\|}$$

Ces définitions sont des approximations consistantes d'ordre 1 des dérivées exactes au point milieu  $M$  si  $U_{N_1}$ ,  $U_{N_2}$ ,  $U_{N_G}$ ,  $U_{N_D}$  sont respectivement des approximations d'ordre 2 du vecteur des variables conservatives au point  $N_1$ ,  $N_2$ ,  $N_G$  et  $N_D$ .

La définition géométrique de ces points ainsi que le calcul détaillé du gradient à une interface sont donnés dans le document [16]

Cette approche conduit à une estimation des flux visqueux couplant la cellule de contrôle  $C_i$  et toutes les cellules voisines ayant au moins un sommet commun.



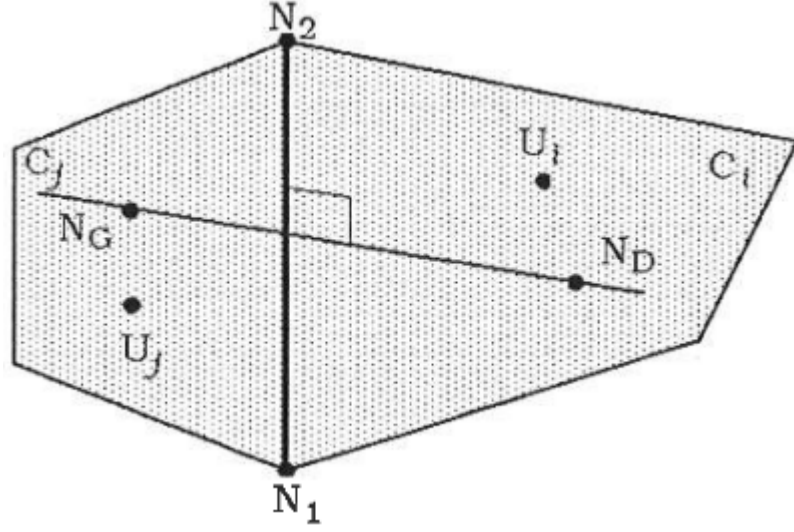


FIGURE 3.2 – Calcul des gradients pour les flux visqueux

### 3.1.3 Discrétisation des termes sources

Les termes sources sont calculés en utilisant un schéma centré. Les variables conservatives étant constantes dans la cellule de contrôle  $C_i$ , on obtient pour le second membre de l'équation 3.2 la discrétisation suivante :

$$\int_{C_i} S(U), dv \approx \text{vol}(C_i) S(U_i).$$

## 3.2 Formulation volumes finis de CoreFlows

Suite au projet de fin d'études que nous avons réalisé en février 2015 nous avons développé une maquette comparant les schémas de Roe, VFRoe avec et sans correction de pression sur des calculs de canaux bouillants 1D et 2D. Les conclusions de ce travail sont qu'il est fondamental d'appliquer un décentrement aux schémas de type Roe pour pouvoir capturer efficacement la solution. Ce traitement des termes source s'inspire des recommandations théoriques dans le cas de termes sources raides voire discontinus. Nous prolongeons ce travail en comparant le schéma de Roe équilibre avec des schémas précis à faible nombre de Mach dans l'environnement CoreFlows basé sur CDMATH et Salomé.

### 3.2.1 Flux convectifs

Les flux convectifs sont de la forme

$$\tilde{F}(U_i, U_j, \mathbf{n}_{ij}) = \frac{F(U_i) + F(U_j)}{2} \mathbf{n}_{ij} - D_{ij} \frac{U_i - U_j}{2}. \quad (3.8)$$

où  $D_{ij}$  est la matrice de décentrement du schéma.

### Schéma upwind

Le flux numérique du schéma de Roe ([21]) est basé sur la résolution d'un problème de Riemann linéarisé monodimensionnel dans la direction normale à une interface entre deux cellules de contrôle

$$\partial_t U + A(U_i, U_j) \partial_{x_n} U = 0$$

avec :

- $x_n$  désigne la direction de la normale  $\mathbf{n}$ , tel que  $U(x_n, 0) = \begin{cases} U_i^t & \text{si } x_n < 0 \\ U_j^t & \text{si } x_n > 0 \end{cases}$ ,
- $A(U_i, U_j)$  matrice jacobienne linéarisé, appelée matrice de Roe (voir annexe 7.3)

Le flux numérique obtenu à l'aide de la résolution exacte de ce problème de Riemann, qui est aussi le flux que les premières versions du code **FLICA 4** utilisent, est donné par l'expression suivante :

$$\tilde{F}(U_i, U_j, \mathbf{n}_{ij}) = \frac{F(U_i) + F(U_j)}{2} \mathbf{n}_{ij} - A_{ij} \frac{U_i - U_j}{2}. \quad (3.9)$$

Le détail du calcul de la matrice de Roe et du décentrement sont donnés en annexe 7.3.

Le premier terme de (3.9 et 3.5) est la partie centrée qui est responsable du fait que le flux numérique  $\tilde{F}$  est consistant avec le flux physique  $F : \tilde{F}(U, U, \mathbf{n}) = F(U) \mathbf{n}$  car les autres termes s'annulent lorsque  $U_i = U_j$ .

Les derniers termes de (3.9 et 3.5) assurent la stabilité du schéma numérique et d'autres propriétés numériques comme le principe du maximum ou la positivité de certaines grandeurs physiques comme la concentration, la densité ou la pression.

### Schéma low Mach

Bien que stables, les schémas de type Roe posent des problèmes de précision à faible nombre de Mach [17]. Lorsque le nombre de Mach d'un écoulement compressible est faible et en l'absence de termes source, l'écoulement devient quasi incompressible et les méthodes de type Godunov, qui utilisent un décentrement basé sur la propagation des ondes acoustiques, deviennent moins précises. Pour résoudre ce problème de précision, les concepteurs de **FLICA 4** ont donc introduit des "*corrections de pression*" pour en augmenter la précision.

Le flux corrigé associé à la formulation originelle du schéma de Roe (équation 3.9) s'écrit :

$$\tilde{F}(U_i, U_j, \mathbf{n}_{ij}) = \frac{F(U_i) + F(U_j)}{2} \mathbf{n}_{ij} - |A_{ij}| \frac{U_i - U_j}{2} + \frac{\tilde{a}(\rho_{mi} \mathbf{u}_{mi} - \rho_{mj} \mathbf{u}_{mj}) \cdot \mathbf{n}_{ij}}{2} \begin{pmatrix} 0 \\ 0 \\ \mathbf{n}_{ij} \\ 0 \end{pmatrix}, \quad (3.10)$$

où  $\tilde{a}$  est la vitesse du son sur l'interface entre  $U_i$  et  $U_j$ .

Le détail du calcul de la correction de pression est donné en annexe section 7.3.5.

### Schéma Staggered

Le nouveau schéma est conçu pour être précis à bas nombre de Mach dans l'esprit des schémas à cellules donneuses ([18]). La méthode est nécessairement implicite avec un flux de la forme

$$F_{i+\frac{1}{2}} = \frac{F(U_i) + F(U_{i+1})}{2} + D_{MAC} \frac{U_i - U_{i+1}}{2}. \quad (3.11)$$

où  $D_{MAC}$  se construit comme la jacobienne de  $F$  sur la face à un détail près qui est que les coefficients liés à la pression sont multipliés par -1. Cette idée correspond au fait que dans les schémas de type MAC les inconnues scalaires (densité, pression et énergie) sont discrétisées sur un maillage différent de celui qui porte le vecteur vitesse.

Plus précisément, la jacobienne du modèle à quatre équations sous l'hypothèse  $\mathbf{u}_v = \mathbf{u}_l$  prend en 1D la forme

$$Jac(U) = \begin{pmatrix} 0 & 0 & {}^t\mathbf{n}_{ij} & 0 \\ -u_n c & u_n & c {}^t\mathbf{n}_{ij} & 0 \\ (\chi + \frac{1}{2}\kappa|\mathbf{u}|^2)\mathbf{n}_{ij} - u_n \mathbf{u} & \xi \mathbf{n}_{ij} & \mathbf{u} \otimes \mathbf{n}_{ij} + u_n \mathbb{I}_d - \kappa \mathbf{n}_{ij} \otimes \mathbf{u} & \kappa \mathbf{n}_{ij} \\ (\chi + \frac{1}{2}\kappa u^2 - H)u_n & \xi u_n & H {}^t\mathbf{n}_{ij} - \kappa u_n {}^t\mathbf{u} & (\kappa + 1)\tilde{u}_n \end{pmatrix}$$

avec  $\chi, \xi$  et  $\kappa$  sont les dérivées partielles de la pression définie par la relation

$$dp = \chi d\rho + \xi d(\rho c) + \kappa d(\rho e).$$

et la matrice  $D_{MAC}$  correspondant est

$$D_{MAC}(U) = \text{signe}(\mathbf{u}_n) \begin{pmatrix} 0 & 0 & {}^t\mathbf{n}_{ij} & 0 \\ -u_n c & u_n & c {}^t\mathbf{n}_{ij} & 0 \\ -(\chi + \frac{1}{2}\kappa|\mathbf{u}|^2)\mathbf{n}_{ij} - u_n \mathbf{u} & -\xi \mathbf{n}_{ij} & \mathbf{u} \otimes \mathbf{n}_{ij} + u_n \mathbb{I}_d + \kappa \mathbf{n}_{ij} \otimes \mathbf{u} & -\kappa \mathbf{n}_{ij} \\ (-\chi - \frac{1}{2}\kappa u^2 - H)u_n & -\xi u_n & H {}^t\mathbf{n}_{ij} + \kappa u_n {}^t\mathbf{u} & (-\kappa + 1)\tilde{u}_n \end{pmatrix}$$

### 3.2.2 Flux visqueux $F_V$

Notre objectif principal étant l'amélioration de la discrétisation des flux convectifs et des termes sources de **FLICA 4**, les flux visqueux de CoreFlows sont discrétisés par une approche simple à deux points moins précise que celle de **FLICA 4**.

Le gradient de  $U$  sur la face entre les cellules  $C_i$  et  $C_j$  de la figure 3.2 est tout simplement

pris égal à

$$\frac{U_i - U_j}{\|G_i G_j\|}$$

où  $G_i$  et  $G_j$  sont les barycentres respectifs des cellules  $C_i$  et  $C_j$ .

### 3.2.3 Traitement des termes source

Le schéma volumes finis semi-discret de l'équation (3.1) s'écrit sous la forme :

$$\frac{dU_i}{dt}(t) + \frac{1}{|C_i|} \sum_{j \in \nu(i)} s_{ij} F_{ij} = S_i \quad (3.12)$$

Le traitement centré consiste à prendre

$$S(U_i) = S(U_i, x_i, t). \quad (3.13)$$

Le traitement décentré consiste à prendre

$$S_i = \frac{1}{\text{perimeter}(i)} \sum_{j \in \nu(i)} s_{ij} S_{ij} \quad (3.14)$$

avec

$$S_{ij} = \frac{S(U_i) + S(U_j)}{2} + \text{signe}(A_{i,j}) \frac{S(U_i) - S(U_j)}{2} \quad (3.15)$$

# Application : CoreFlows

---

## 4.1 Présentation de CoreFlows

COREFLOWS est un logiciel Open Source, développé en C++/*Python*, dédié à la résolution des systèmes d'équations aux dérivées partielles découlant des modèles thermohydraulique et d'écoulements diphasiques dans les coeurs des réacteurs nucléaires.

Le but de COREFLOWS est de proposer quelques modèles de base résolus numériquement par la méthode de volumes finis, avec différents schémas numériques. Il s'agit d'un environnement très simple accessible aux étudiants et suffisamment souple pour permettre aux chercheurs de tester de nouveaux schémas numériques sur des maillages quelconques en 3D.

Le principal objectif de COREFLOWS est l'étude de

- Schémas numériques pour les écoulements compressibles à faible nombre de Mach,
- Schémas équilibres avec des termes sources raides (changement de phase, les pertes de pression,  $\dots$  etc),
- Flux à contre-courant,
- Les schémas préservant la fraction volumique phasique  $\alpha \in [0, 1]$ ,
- Convergence des méthodes de volumes finis,
- Nouveaux préconditionneurs pour les méthodes implicites pour écoulements diphasiques,
- Le couplage de modèles fluides ou couplage multiphysique (la thermohydraulique, la neutronique ou thermohydraulique et thermique solides )

COREFLOWS repose sur la ToolBox [9] du projet CDMATH [8] pour la gestion de maillages et de champs, et la librairie PETSc [10] pour la manipulation de grosse matrices creuses.

## 4.2 Modèles physiques

Les modèles physiques proposés par COREFLOWS sont présentés ci-dessous par ordre de complexité mathématique :

### 4.2.1 Problèmes linaires scalaires

#### Équation de transport

L'équation de transport résolu dans COREFLOWS est la suivante :

$$\partial_t H + \mathbf{u} \cdot \vec{\nabla} H = \Phi + \lambda_{sf}(T_s - T) \quad (4.1)$$

avec

- $H$  L'inconnu principal, qui est l'enthalpie du fluide
- $\vec{u}$  constant, représente la vitesse de transport
- $\Phi$  Puissance thermique
- $T_s$  Température de saturation
- $T = T_0 + \frac{H-H_0}{c_p}$  Température du fluide
- $\lambda_{sf}$  coefficient de transfert de chaleur
- $c_p$  capacité calofirique à pression constante

Dans la section Exemples (4.5.2), on retrouve le script *Python*, pour résoudre 4.1.

#### Équation de diffusion

L'équation de transport résolu dans COREFLOWS est la suivante :

$$\partial_t T = d\Delta T + \frac{\Phi + \lambda_{sf}(T_f - T)}{\rho c_p} \quad (4.2)$$

avec

- $T$  L'inconnu principal, qui est la température
- $\lambda_{sf}$  coefficient de transfert de chaleur
- $\rho$  Densité constante,
- $c_p$  Capacité calofirique à pression constante
- $\Phi$  Puissance thermique
- $T_f$  Température du fluide

### 4.2.2 Équation de Navier-Stokes

Le Modèle monophasique de Navier-Stokes comprend trois équations de conservation : masse, quantité de mouvement et énergie.

$$\begin{cases} \frac{\partial \rho}{\partial t} + \nabla \cdot \vec{q} & = & 0 \\ \frac{\partial \vec{q}}{\partial t} + \nabla \cdot \left( \vec{q} \otimes \frac{\vec{q}}{\rho} + p \mathbb{I}_d \right) - \nu \Delta(\vec{u}) & = & \rho \vec{g} - K \rho ||\mathbf{u}|| \mathbf{u} \\ \frac{\partial (\rho E)}{\partial t} + \nabla \cdot \left[ (\rho E + p) \frac{\vec{q}}{\rho} \right] - \lambda \Delta T & = & \Phi + \rho \vec{g} \cdot \mathbf{u} - K \rho ||\mathbf{u}||^3 \end{cases} \quad (4.3)$$

où

- $\rho$  masse volumique,
- $\vec{u}$  vecteur vitesse,
- $\vec{q} = \rho \vec{u}$  quantité de mouvement,
- $p$  pression,
- $\rho e$  énergie interne,
- $\rho E = \rho e + \frac{||\vec{q}||^2}{2\rho}$  énergie totale,
- $T$  température,
- $\Phi$  a heat source term,
- $\nu$  viscosité,
- $\lambda$  conductivité thermique.

On rajoute au système (4.3), la loi de fermeture du gaz raidis  $p = (\gamma - 1)\rho e - \gamma p_0$  et une linéarisation de la loi d'énergie interne  $e(T)$  valides autour des points  $(P = 1bar, T = 300K)$  ou  $(P = 155bars, T = 618K)$  selon la valeur de l'éumérateur *pressureEstimate*. Pour simplifier le système, on considère la viscosité et la conductivité constante, et on néglige la contribution des forces visqueuses dans l'équation de l'énergie. Comme pour l'équation de transport, on retrouve un exemple de script *Python*, pour résoudre 4.3 dans la section (4.5.2)

### 4.2.3 Modèles diphasiques

Modèles diphasiques sont obtenus en moyennant les équations de conservations voir l'introduction du chapitre 6

Le drift model est le modèle utilisé dans **FLICA4** [15], tandis que les modèles bi-fluide deux fluides sont utilisés dans Cathare[18], Neptune\_CFD [19], Cobra-TF [11].

#### Modèle à quatre équations : Le drift model

Ce modèle consiste une loi de bilan pour la masse vapeur ainsi que d'une loi de conservation de la masse totale, d'une loi de bilan de la quantité de mouvement totale et d'une

loi de bilan de l'énergie totale.

Les inconnues principales sont concentration massique de vapeur  $c_v$ , la pression  $P$ , la vitesse de mélange  $\mathbf{u}_m$ , et la température commune  $T$ . Il utilise des lois d'état des gaz parfaits  $p_g(\rho_g, T)$  and  $p_l(\rho_l, T)$  de même qu'une loi d'énergie interne linéarisée  $e_k(T)$  valables autour des points de saturation ( $P = 1\text{bar}, T = 373\text{K}$ ) ou ( $P = 155\text{bars}, T = 618\text{K}$ ) selon la valeur de l'énumérateur *pressureEstimate*.

Le drift model est un système à quatre équations non linéaires prenant la forme conservative suivante :

$$\left\{ \begin{array}{lll} \partial_t(\alpha_v \rho_v + \alpha_l \rho_l) & + \nabla \cdot (\alpha_v \rho_v \mathbf{u}_v + \alpha_l \rho_l \mathbf{u}_l) & = 0 \\ \partial_t(\alpha_v \rho_v) & + \nabla \cdot (\alpha_v \rho_v \mathbf{u}_v) & = \Gamma_v(h_m, \Phi) \\ \partial_t(\alpha_v \rho_v \mathbf{u}_v + \alpha_l \rho_l \mathbf{u}_l) & + \nabla \cdot (\alpha_v \rho_v \mathbf{u}_v \otimes \mathbf{u}_v + \alpha_l \rho_l \mathbf{u}_l \otimes \mathbf{u}_l + p \mathbb{I}_d) & = \rho_m \vec{g} - K_v \alpha_v \rho_v ||\mathbf{u}_v|| \mathbf{u}_v \\ & & - K_l \alpha_l \rho_l ||\mathbf{u}_l|| \mathbf{u}_l \\ \partial_t(\alpha_v \rho_v E_v + \alpha_l \rho_l E_l) & + \nabla \cdot (\alpha_v \rho_v H_v \mathbf{u}_v + \alpha_l \rho_l H_l \mathbf{u}_l) & = \Phi + \rho \vec{g} \cdot \mathbf{u} \\ & & - K_v \alpha_v \rho_v ||\mathbf{u}_v||^3 - K_l \alpha_l \rho_l ||\mathbf{u}_l||^3 \end{array} \right. ,$$

où l'énergie totale et l'enthalpie totale sont définies comme suit :

$$E_k = e_k + \frac{1}{2} |\mathbf{u}_k|^2, \quad H_k = h_k + \frac{1}{2} |\mathbf{u}_k|^2, \quad k = v, l,$$

où  $e_k$  est l'énergie interne, et  $h_k = e_k + \frac{p}{\rho_k}$  l'enthalpie associée à la phase  $k$  et

$$\begin{aligned} \rho_m &= \alpha_v \rho_v + \alpha_l \rho_l \\ \mathbf{u}_m &= \frac{\alpha_v \rho_v \mathbf{u}_v + \alpha_l \rho_l \mathbf{u}_l}{\alpha_v \rho_v + \alpha_l \rho_l} \\ h_m &= \frac{\alpha_v \rho_v h_v + \alpha_l \rho_l h_l}{\alpha_v \rho_v + \alpha_l \rho_l}. \end{aligned}$$

$\mathbf{u}_r = \mathbf{u}_v - \mathbf{u}_l = \vec{f}_r(c_v, \mathbf{u}_m, \rho_m)$  : la vitesse relative :

Le changement de phase est modélisée à l'aide de la formule

$$\Gamma_v = \begin{cases} \frac{\Phi}{\mathcal{L}} & \text{if } h_l^{sat} \leq h < h_v^{sat} \text{ and } 0 < \alpha_v < 1 \\ 0 & \text{sinon} \end{cases}. \quad (4.4)$$



### Modèle bi-fluide isotherme

Le modèle consiste dans les équations de bilan de masse et quantité de mouvement. Les principales inconnues sont  $\alpha$ ,  $P$ ,  $\mathbf{u}_v$ ,  $\mathbf{u}_l$ . Le modèle utilise les lois des gaz raidis  $p_v(\rho_v)$  et  $p_l(\rho_l)$  et une linéarisation des lois d'énergie interne  $e_k(T)$ , valides autour des points  $(P = 1bar, T = 300K)$  ou  $(P = 155bars, T = 618K)$  selon la valeur de l'énumérateur *pressureEstimate*.

on notera  $k = v, l$  les indices respectifs à chaque phase :  $l$  pour la phase liquide et  $v$  pour la phase vapeur.

La pression moyenne commune aux deux phases est notée  $p$ .

Dans ce modèle, l'équilibre de pression entre les deux phases nous ramène à résoudre le système suivant :

$$\left\{ \begin{array}{ll} \frac{\partial m_v}{\partial t} + \nabla \cdot \vec{q}_v & = 0, \\ \frac{\partial m_l}{\partial t} + \nabla \cdot \vec{q}_l & = 0, \\ \frac{\partial \vec{q}_v}{\partial t} + \nabla \cdot (\vec{q}_v \otimes \frac{\vec{q}_v}{m_v}) + \alpha_v \vec{\nabla} p & \\ + \Delta p \nabla \alpha_v - \nu_v \Delta \mathbf{u}_v & = m_v \vec{g} - K_v m_v ||\mathbf{u}_v|| \mathbf{u}_v \\ \frac{\partial \vec{q}_l}{\partial t} + \nabla \cdot (\vec{q}_l \otimes \frac{\vec{q}_l}{m_l}) + \alpha_l \vec{\nabla} p & \\ + \Delta p \nabla \alpha_l - \nu_l \Delta \mathbf{u}_l & = m_l \vec{g} - K_l m_l ||\mathbf{u}_l|| \mathbf{u}_l, \end{array} \right. \quad (4.5)$$

où  $\alpha_v + \alpha_l = 1$ ,  $m_k = \alpha_k \rho_k$  et  $\vec{q}_k = \alpha_k \rho_k \mathbf{u}_k$ .

Ici,  $\nu_k$  est la viscosité de la phase  $k$

### Modèle à cinq équations

Les principales inconnues sont  $\alpha$ ,  $P$ ,  $\mathbf{u}_v$ ,  $\mathbf{u}_l$  et  $T = T_v = T_L$ . Le modèle utilise la loi des gaz raidis  $p_v(\rho_v, T)$  et  $p_l(\rho_l, T)$  valides autour des points de saturation  $(P = 1bar, T = 373K)$  ou  $(P = 155bars, T = 618K)$  selon la valeur de l'énumérateur *pressureEstimate*.

$$\left\{ \begin{array}{ll} \frac{\partial m_v}{\partial t} + \nabla \cdot \vec{q}_v & = \Gamma_v(h_v, \Phi), \\ \frac{\partial m_l}{\partial t} + \nabla \cdot \vec{q}_l & = \Gamma_l(h_l, \Phi), \\ \frac{\partial \vec{q}_v}{\partial t} + \nabla \cdot (\vec{q}_v \otimes \frac{\vec{q}_v}{m_v}) + \alpha_v \nabla p & \\ + \Delta p \nabla \alpha_v - \nu_v (\Delta \frac{\vec{q}_v}{m_v}) & = m_v \vec{g} - K_v m_v ||\mathbf{u}_v|| \mathbf{u}_v \\ \frac{\partial \vec{q}_l}{\partial t} + \nabla \cdot (\vec{q}_l \otimes \frac{\vec{q}_l}{m_l}) + \alpha_l \nabla p & \\ + \Delta p \nabla \alpha_l - \nu_l (\Delta \frac{\vec{q}_l}{m_l}) & = m_l \vec{g} - K_l m_l ||\mathbf{u}_l|| \mathbf{u}_l, \\ \partial_t \rho_m E_m + \nabla \cdot (\alpha_v \rho_v H_v^t \mathbf{u}_v + \alpha_l \rho_l H_l^t \mathbf{u}_l) & = \Phi + \rho \vec{g} \cdot \mathbf{u} - K_v m_v ||\mathbf{u}_v||^3 - K_l m_l ||\mathbf{u}_l||^3 \end{array} \right.$$

où

$$\begin{aligned}\rho_m &= \alpha_v \rho_v + \alpha_l \rho_l \\ E_m &= \frac{\alpha_v \rho_v E_v + \alpha_l \rho_l E_l}{\alpha_v \rho_v + \alpha_l \rho_l}.\end{aligned}$$

Le changement de phase est modélisé à l'aide de la formule

$$\Gamma_v = \begin{cases} \frac{\Phi}{\mathcal{L}} & \text{if } h_l^{sat} \leq h < h_v^{sat} \text{ and } 0 < \alpha_v < 1 \\ 0 & \text{sinon} \end{cases}. \quad (4.6)$$

### 4.3 Méthodes numériques

COREFLOWS propose une discrétisation en volumes finis explicite ou implicite (cf l'introduction de [25]). Cette discrétisation en volumes finis permet une manipulation plus simple de géométries quelconques et de maillages générés par Salomé [26].

Les quatre méthodes numériques disponibles dans COREFLOWS pour discrétiser le modèle de dérive sont

- le schéma upwind, équivalent au schéma de Roe [21], avec une matrice de Roe approchée
- le schéma centré, stable en implicite, d'ordre 2 en espace
- le schéma lowMach qui consiste en une correction de pression pour rendre le schéma upwind précis à faible nombre de Mach
- le schéma staggered qui est un schéma inspiré du schéma MAC sur grille décalée pour obtenir une précision à faible nombre de Mach sans apparition de modes en échiquiers

Nous renvoyons à la sous-section 3.2 pour plus de détails sur ces méthodes.

Il est possible d'améliorer la précision du schéma en utilisant une correction entropique. C'est en particulier le cas pour les écoulements à contre-courant (voir [23]), car ils peuvent présenter des points soniques. Il est possible d'améliorer le traitement des termes sources raides en leur appliquant un décentrement inspiré de [22] (voir annexe 7.4).

Les schémas explicites ne nécessitent pas de résolution de système linéaires mais sont soumis à une condition CFL très contraignante sur le pas de temps. Lorsque cela est possible, l'utilisation de grande CFL permet d'atteindre rapidement le régime stationnaire. Les schémas implicites demandent la résolution des systèmes non linéaires en utilisant une méthode de type Newton. Ils permettent en théorie l'emploi de grands pas de temps, mais en pratique la taille des pas de temps est limitée par la capacité à résoudre les systèmes linéaires et à faire converger les itérations de Newton. Il est possible pour des calculs difficiles de procéder en plusieurs étapes en résolvant le problème sur des maillages de plus en plus fins.

## 4.4 Développement

COREFLOWS a été développé dans le langage de programmation orienté objet C++ en utilisant les bibliothèques standards et multi-plateforme stl.

COREFLOWS repose sur la ToolBox [9] du projet CDMATH [8] pour la gestion de maillages et de champs, et la librairie PETSc [10] pour la manipulation de grosse matrices creuses.

### 4.4.1 Architecture logicielle

L'architecture globale de notre application est assez simple. En effet elle se résume en huit classes principales.

COREFLOWS est composé de six classes concrètes portant sur des modèles spécifiques :

- **SinglePhase** Modélisation des équations compressibles de Navier-Stokes (section 4.2.2)
- **DriftModel** modélise le modèle à 4équations (section 4.2.3)
- **IsothermalTwoFluid** la mise en œuvre du modèle bi-fluide isentropique (section 4.2.3)
- **FiveEqsTwoFluid** modéliastion de l'égalité de temperature entre deux fluides (section 4.2.3)
- **Problem transport** équation d'advection scalaire pour l' enthalpie fluide (section 4.2.1)
- **ProblemDiffusion** équation de la chaleur scalaire pour la température des barres d'uranium (section 4.2.1)

En plus de ces classes , il ya deux classes abstraites qui servent à la factorisation des fonctions communes à plusieurs modèles :

- **ProblemFluid** contient les méthodes qui sont communes aux modèles fluides, non-scalaires,
- **ProblemCoreFlows** contient les méthodes qui sont communes à tous les modèles, scalaires ou pas.

Le diagramme 4.1 illustre cette architecture

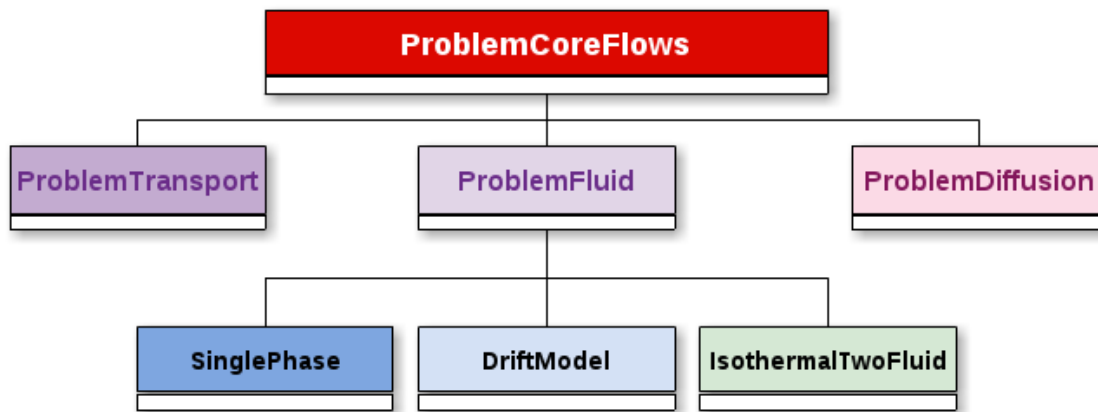


FIGURE 4.1 – Hiérarchie des classes

### 4.4.2 Génie Logiciel

#### CMake

COREFLOWS utilise le moteur de production CMake, qui permet d'automatiser la génération de fichiers de construction standard (makefiles) ainsi que l'exécutable.

CMake est un système de construction logicielle libre, multi-plateforme et multi-langage. L'un des avantages de CMake est qu'il permet de générer des fichiers projets pour un grand nombre d'IDE comme Eclipse [29] ou CodeBlocks [30]. Ainsi, chaque développeur peut utiliser son propre IDE<sup>12</sup> sans que cela n'impacte le projet ou ne restreigne les autres utilisateurs.

CMake s'utilise de manière très peu intrusive en ajoutant simplement des fichiers CMakeLists.txt dans chacun des répertoires du projet, contenant les directives de construction : création d'une bibliothèque, cible personnalisée pour la création d'une documentation, directives d'installation, etc. On notera que CMake gère le build dit « out-of-sources », c'est-à-dire la construction du logiciel dans un répertoire séparé des sources ce qui présente l'avantage de laisser l'arborescence du projet propre.

COREFLOWS utilise aussi "gcov", un teste de couverture permettant de quantifier la proportion de code testé et donc d'avoir une vue sur les parties du code qui mériteraient d'être testées pour s'assurer de leur qualité.

#### Interface graphique : Qt-Designer

L'interface graphique a été développée en Qt-Designer [36], qui permet de concevoir les fenêtres visuellement, mais aussi modifier les propriétés des widgets, d'utiliser des layouts et d'effectuer la connexion entre signaux et slots.

---

12. Integrated Development Environment

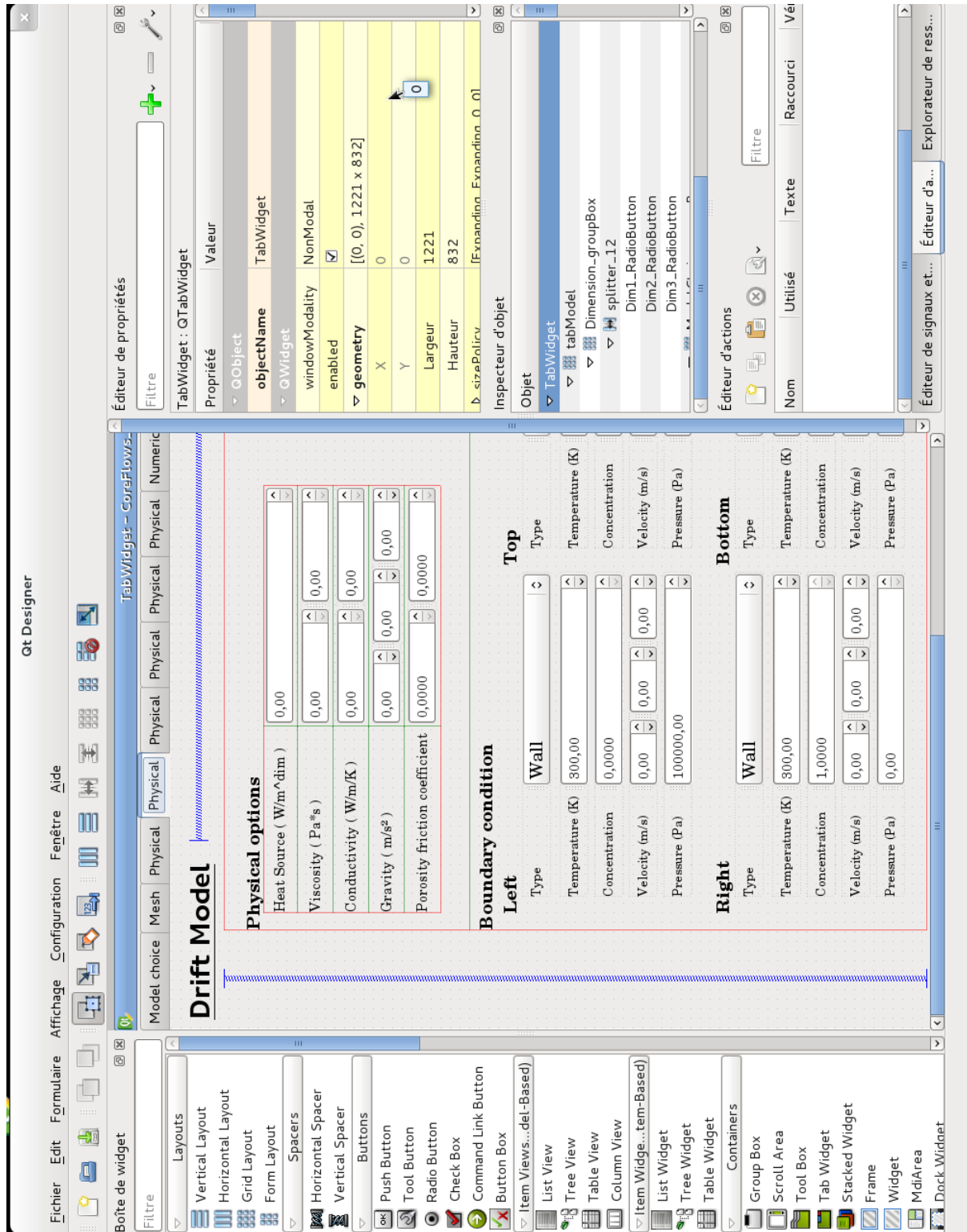


FIGURE 4.2 – Environnement de Développement de l'interface graphique de COREFlows

## API *Python* : SWIG

SWIG<sup>11</sup> a permis à COREFLOWS d'encapsuler le code source écrit en C++ avec le langage script *Python*.

Swig est un "générateur d'interfaces" qui permet de créer des bindings à partir de fichiers d'en-tête.

Il est actuellement capable de générer des modules pour plus de quinze langages et ce, d'une manière quasiment uniforme. Le mécanisme de génération est représenté sur la figure 4.3.

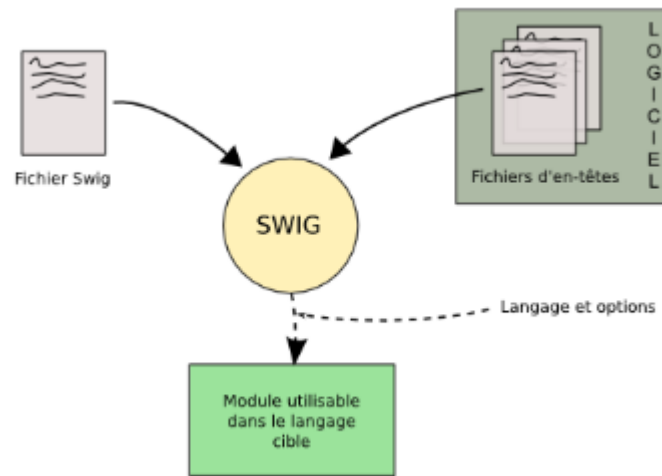


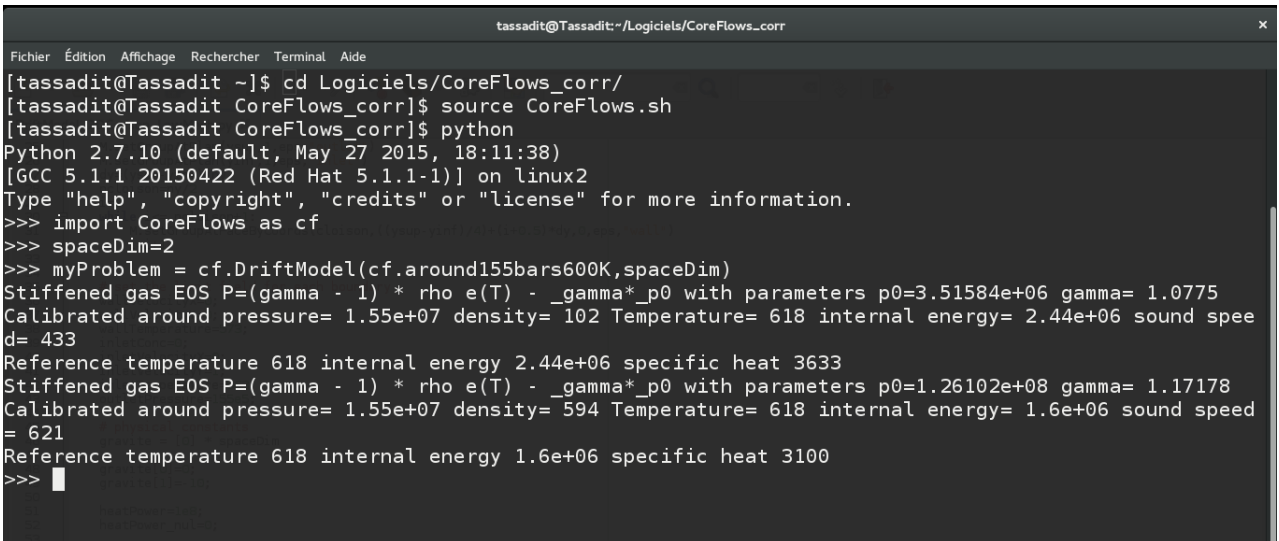
FIGURE 4.3 – Mécanisme de génération de SWIG

Le résultat est un module *Python* utilisable par une personne connaissant les méthodes de COREFLOWS. L'API *Python* est avantagé par la flexibilité de son langage, son interactivité, et son prototypage lisible. La documentation des sources de COREFLOWS reste valide en *Python*.

Le SWIG a apporté la puissance de *Python* à la performance de C++.

---

11. Simplified Wrapper and Interface Generator



```
tassadit@Tassadit:~/Logiciels/CoreFlows_corr
Fichier  Édition  Affichage  Rechercher  Terminal  Aide
[tassadit@Tassadit ~]$ cd Logiciels/CoreFlows_corr/
[tassadit@Tassadit CoreFlows_corr]$ source CoreFlows.sh
[tassadit@Tassadit CoreFlows_corr]$ python
Python 2.7.10 (default, May 27 2015, 18:11:38)
[GCC 5.1.1 20150422 (Red Hat 5.1.1-1)] on linux2
Type "help", "copyright", "credits" or "license" for more information.
>>> import CoreFlows as cf
>>> spaceDim=2
>>> myProblem = cf.DriftModel(cf.around155bars600K,spaceDim)
Stiffened gas EOS  $P=(\gamma - 1) * \rho e(T) - \gamma p_0$  with parameters  $p_0=3.51584e+06$   $\gamma= 1.0775$ 
Calibrated around pressure= 1.55e+07 density= 102 Temperature= 618 internal energy= 2.44e+06 sound speed= 433
Reference temperature 618 internal energy 2.44e+06 specific heat 3633
Stiffened gas EOS  $P=(\gamma - 1) * \rho e(T) - \gamma p_0$  with parameters  $p_0=1.26102e+08$   $\gamma= 1.17178$ 
Calibrated around pressure= 1.55e+07 density= 594 Temperature= 618 internal energy= 1.6e+06 sound speed= 621
Reference temperature 618 internal energy 1.6e+06 specific heat 3100
>>> █
```

FIGURE 4.4 – Exemple d'utilisation de COREFLOWS sur *Python*

### 4.4.3 Documentation : Doxygen

Doxygen est un générateur de documentation logicielle à partir du code source d'un programme. Pour cela, il tient compte du langage dans lequel le code source a été écrit, ainsi que les commentaires rédigés dans un format particulier au niveau des classes, des méthodes et des attributs. La documentation peut être générée dans l'un ou plusieurs des formats suivants : HTML, Latex, PDF avec des hyperliens, PostScript ou RTF. Doxygen permet aussi d'extraire les informations suivantes des sources :

- + Prototype et documentation des classes et leur hiérarchie ;
- + Différents types de graphes : diagrammes de classes, de collaboration, d'appels, d'inclusion, etc.
- + Documentation des structures de données ;
- + Liste des fichiers inclus ;
- + Prototype et documentation des fonctions, qu'elles soient locales, privées ou publiques, etc. ;
- + Liste des modules (groupements définis dans la documentation) ;
- + Un index de tous les identifiants ;
- + Des fichiers sources annotés (par exemple avec les numéros de lignes) et navigables (par exemple avec HTML, avec lequel des identifiants renvoient vers la documentation associée).

# CoreFlows 1.0

A library providing basic two phase flow models and finite volume numerical methods for students and researchers in computational multiphase fluid dynamics

<b>Main Page</b>	<b>Related Pages</b>	<b>Classes</b>	<b>Files</b>	Search
------------------	----------------------	----------------	--------------	--------

## CoreFlows User Guide

## Presentation of CoreFlows

CoreFlows is an open source C++/Python library intended at solving PDE systems arising from the thermalhydraulics of two phase flows in power plant boilers. It is a simple environment meant at students and researchers to test new numerical methods on general geometries with unstructured meshes. It is developped at CEA Saclay by Michael Ndjinga and his students since 2014 and proposes a few basic models and finite volume numerical methods. Some of the main objectives are the study of

- Numerical schemes for compressible flows at low Mach numbers
- Well balanced schemes for stiff source terms (heat source, phase change, pressure losses)
- Flow inversion and counter-current two phase flows
- Schemes that preserve the phasic volume fraction  $\alpha \in [0, 1]$
- Convergence of finite volume methods
- New preconditioners for implicit methods for two phase flows
- The coupling of fluid models or multiphysics coupling (eg thermal hydraulics and neutronics or thermal hydraulics and solid thermics)

CoreFlows relies on the [Toolbox \[21\]](#) of the project [CDMATH \[20\]](#) for the handling of meshes and fields, and on the library [Petsc \[19\]](#) 3.4.5 for the handling of large sparse matrices.

## Contents

This document is the user guide of the CoreFlows library. It is organized as follows :

- [The Physical Models](#)
  - [The linear scalar problems](#)
    1. [The transport equation](#)
    2. [The diffusion equation](#)
  - [The Navier-Stokes model](#)
  - [The two-phase flow models](#)
    1. [The Drift model](#)
    2. [The isothermal two-fluid model](#)
    3. [The five equation two-fluid model](#)
- [Software structure](#)
- [The numerical methods](#)
- Summary of available [functionalities](#)
- [CoreFlows example scripts](#)

## Installation and use

In order to install CoreFlows you will need the packages [CMAKE \[25\]](#) , [HDF5 \[23\]](#) and possibly

- [SWIG \[22\]](#) if you want to use python scripts
- [DOXYGEN \[24\]](#) if you want to generate this html documentation.

Instructions for installation and use of CoreFlows can be found here [Installation and use of CoreFlows](#).

## References

[A specific page dedicated to references is available here](#).



## 4.5 Exemples d'utilisation de CoreFlows

### 4.5.1 Interface graphique

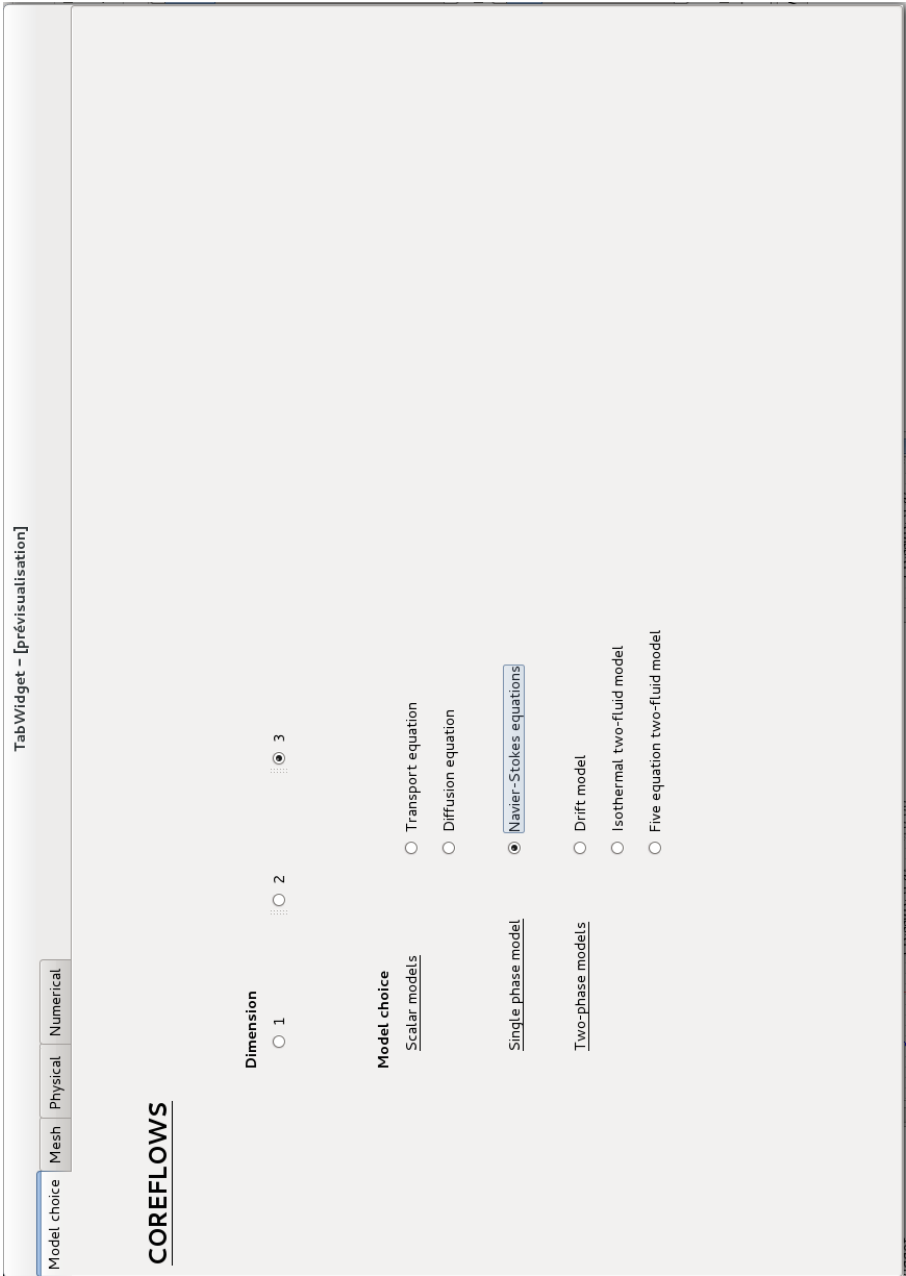


FIGURE 4.6 – Interface de COREFlows 1/4

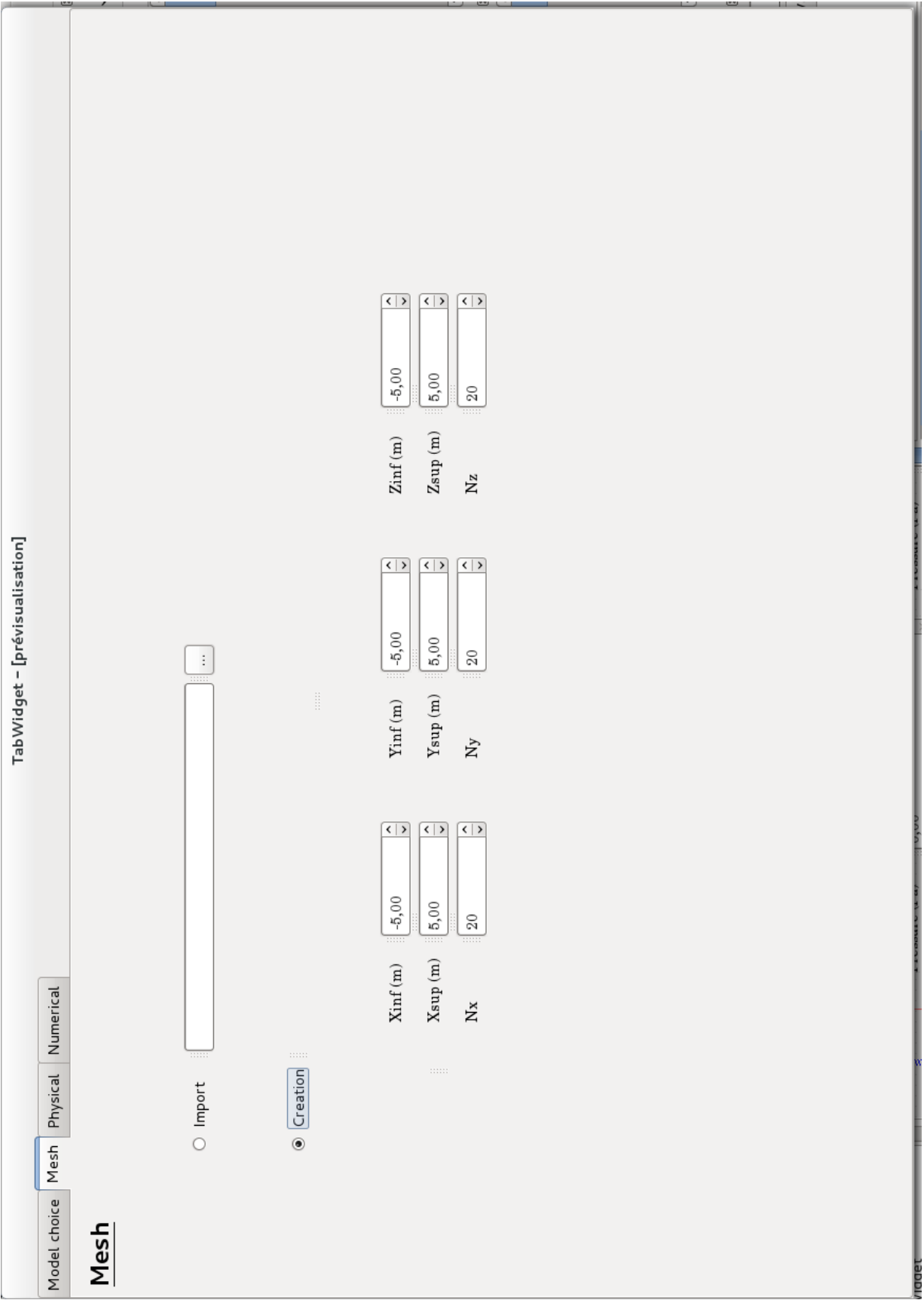


FIGURE 4.7 – Interface de COREFlows 2/4

TabWidget - [Prévisualisation]

Model choiceMeshPhysicalNumerical

Single Phase

Physical options

Heat Source ( W/m^dim )0,00

Viscosity (Pa\*s)0,00

Conductivity (W/m/K)0,00

Gravity (m/s²)0,000,000,00

Porosity friction0,00

Initial condition

Pressure (Pa)100000,00

Velocity (m/s)0,000,000,00

Temperature (K)300,00

Fluid Parameters

PhaseLiquid

Estimate pressureAround 1bar 300K

Boundary condition

Left

TypeWall

Temperature (K)300,00

Velocity (m/s)0,000,000,00

Pressure (Pa)100000,00

Top

TypeOutlet

Temperature (K)300,00

Velocity (m/s)0,000,000,00

Pressure (Pa)100000,00

Front

TypeWall

Temperature (K)300,00

Velocity (m/s)0,000,000,00

Pressure (Pa)100000,00

Right

TypeWall

Temperature (K)300,00

Velocity (m/s)0,000,000,00

Pressure (Pa)100000,00

Bottom

TypeInlet

Temperature (K)300,00

Velocity (m/s)0,000,000,00

Pressure (Pa)100000,00

Back

TypeWall

Temperature (K)300,00

Velocity (m/s)0,000,000,00

Pressure (Pa)100000,00

FIGURE 4.8 – Interface de COREFlows 3/4

TabWidget – [prévisualisation]

Model choice Mesh Physical Numerical

## Numerical Options

**Simulation parameters**

Number of time steps

Save frequency

Maximum time

Precision

CFL number

**Method**

**Linear Solver**

**Scheme**

**Preconditioner**

**File Name**

Launch simulation

FIGURE 4.9 – Interface de COREFlows 4/4

### 4.5.2 Scripts *Python*

L'interface graphique de COREFLOWS est très simple à utiliser, néanmoins, elle est assez limitée, on peut faire des calculs plus complexes, par conséquent l'utilisation des scripts *Python*.

#### Exemple 1 : Équation de Transport

```
import \textsc{CoreFlows} as cf

def TransportEquation_1DHeatedChannel():
    spaceDim = 1;
    # Prepare for the mesh
    xinf = 0 ;
    xsup=4.2;
    nx=2;
    # set the limit field for each boundary
    inletEnthalpy=1.3e6;
    # Set the transport velocity
    transportVelocity=[5];
    myProblem = cf.TransportEquation(cf.Liquid,cf.around155bars600K,
    transportVelocity);
    nVar = myProblem.getNumberOfVariables();
    # Prepare for the initial condition
    VV_Constant = [1.3e6]; #initial enthalpy
    #Set rod temperature and heat exchange coefficient
    rodTemp=623;#Rod clad temperature
    heatTransfertCoeff=1000;#fluid/solid heat exchange coefficient
    myProblem.setRodTemperature(rodTemp);
    myProblem.setHeatTransfertCoeff(heatTransfertCoeff);
    #Initial field creation
    print("Building initial data " );
    myProblem.setInitialFieldConstant(spaceDim,VV_Constant,xinf,xsup,nx,"inlet",
    "neumann");
    # Set the boundary conditions
    myProblem.setInletBoundaryCondition("inlet", inletEnthalpy);
    myProblem.setNeumannBoundaryCondition("neumann")
    # Set the numerical method
    myProblem.setNumericalMethod(cf.upwind, cf.Explicit);
    # name file save
```

```
fileName = "1DHeatedChannel";
    # parameters calculation
MaxNbOfTimeStep = 3 ;
freqSave = 5;
cfl = 0.95;
maxTime = 5;
precision = 1e-6;
myProblem.setCFL(cfl);
myProblem.setPrecision(precision);
myProblem.setMaxNbOfTimeStep(MaxNbOfTimeStep);
myProblem.setTimeMax(maxTime);
myProblem.setFreqSave(freqSave);
myProblem.setFileName(fileName);
myProblem.setDISPLAY( True,True, True, True);
    # evolution
myProblem.initialize();
print("Running python "+ fileName );
ok = myProblem.run();
if (ok):
print( "Simulation python " + fileName + " is successful !" );
pass
else:
print( "Simulation python " + fileName + " failed ! " );
pass
print( "----- End of calculation !!! -----" );
myProblem.terminate();
return ok
```

**Exemple 2 : Équation de Navier-Stokes - monophasique -**

```
import \textsc{CoreFlows} as cf
def SinglePhase_3DHeatDrivenCavity():
spaceDim = 3;
    #Preprocessing: mesh data
xinf=0;
xsup=1;
yinf=0;
ysup=1;
zinf=0;
zsup=1;
nx=10;
ny=10;
nz=10;

    # set the limit field for each boundary
coldWallVelocityX=0;
coldWallVelocityY=0;
coldWallVelocityZ=0;
coldWallTemperature=563;
hotWallVelocityX=0;
hotWallVelocityY=0;
hotWallVelocityZ=0;
hotWallTemperature=613;

    # physical constants
gravite = [0] * spaceDim
gravite[2]=-10;
gravite[1]=0;
gravite[0]=0;
viscosite=[8.85e-5];
conductivite=[1000];#Wall heat transfert due to nucleate boiling.

    #-----
myProblem = cf.SinglePhase(cf.Liquid,cf.around155bars600K,spaceDim);
nVar = myProblem.getNumberOfVariables();
#Initial field creation
print("Building initial data " );
    # Prepare for the initial condition
VV_Constant = [0] * nVar
```

```
# constant vector
VV_Constant[0] = 155e5;
VV_Constant[1] = 0 ;
VV_Constant[2] = 0;
VV_Constant[3] = 0;
VV_Constant[4] = 573;

    #Initial field creation
myProblem.setInitialFieldConstant(spaceDim,VV_Constant,xinf,xsup,nx,"hotWall",
"hotWall",yinf,ysup,ny,"hotWall","hotWall",zinf,zsup,nz, "hotWall", "coldWall");

    # Set the boundary conditions
myProblem.setWallBoundaryCondition("coldWall", coldWallTemperature, coldWallVelocityX,
coldWallVelocityY, coldWallVelocityZ);
myProblem.setWallBoundaryCondition("hotWall", hotWallTemperature, hotWallVelocityX,
hotWallVelocityY, hotWallVelocityZ);

    # set physical parameters
myProblem.setViscosity(viscosite);
myProblem.setConductivity(conductivite);
myProblem.setGravity(gravite);

    # set the numerical method
myProblem.setNumericalMethod(cf.upwind, cf.Implicit);
myProblem.setLinearSolver(cf.GMRES,cf.ILU,True);
myProblem.setEntropicCorrection(False);
myProblem.setWellBalancedCorrection(False);

    # name file save
fileName = "3DHeatDrivenCavity";

    # simulation parameters
MaxNbOfTimeStep = 3 ;
freqSave = 1;
cfl = 10;
maxTime = 50;
precision = 1e-6;
myProblem.setCFL(cfl);
myProblem.setPrecision(precision);
myProblem.setMaxNbOfTimeStep(MaxNbOfTimeStep);
myProblem.setTimeMax(maxTime);
myProblem.setFreqSave(freqSave);
myProblem.setFileName(fileName);
myProblem.setNewtonSolver(precision,20);
myProblem.saveConservativeField(True);
```



```
if(spaceDim>1):
myProblem.saveVelocity();
pass
    # evolution
myProblem.initialize();
print("Running python "+ fileName );
ok = myProblem.run();
if (ok):
print( "Simulation python " + fileName + " is successful !" );
pass
else:
print( "Simulation python " + fileName + " failed ! " );
pass
print( "----- End of calculation !!! -----" );
myProblem.terminate();
return ok
```



# Résultats numériques

---

Dans cette partie nous allons d'abord valider nos trois méthodes numériques (low-Mach, upwind et staggered) sur la configuration classique de la cavité entraînée par son couvercle.(sous-section 5.1).

Ensuite nous allons étudier une configuration propre la thermohydraulique qui est celui de la dynamique d'un écoulement piloté par la conduction thermique (sous-section 5.2).

Enfin nous allons étudier une configuration spécifique aux réacteurs nucléaires à savoir la circulation entre deux canaux chauffés différemment ce qui était l'objectif de mon stage (sous-section 5.3) .

## 5.1 La cavite entraînée

Le premier cas test sert à mettre en évidence les limites du schéma upwind lorsque l'écoulement est à faible nombre de Mach et de Froude, et la dynamique quasi incompressible.

On considère un domaine 2D  $[0, 1] \times [0, 1]$ , où Toutes les frontières du domaine sont des parois (fixés), sauf le celle du haut, qui possède une vitesse dans la direction  $\vec{x}$  de valeur  $U_x = 1m/s$ .

On impose une températures de  $273^\circ C$ , une pression de  $155bars$  et une viscosité du 0.025.

On considère les maillages suivants :

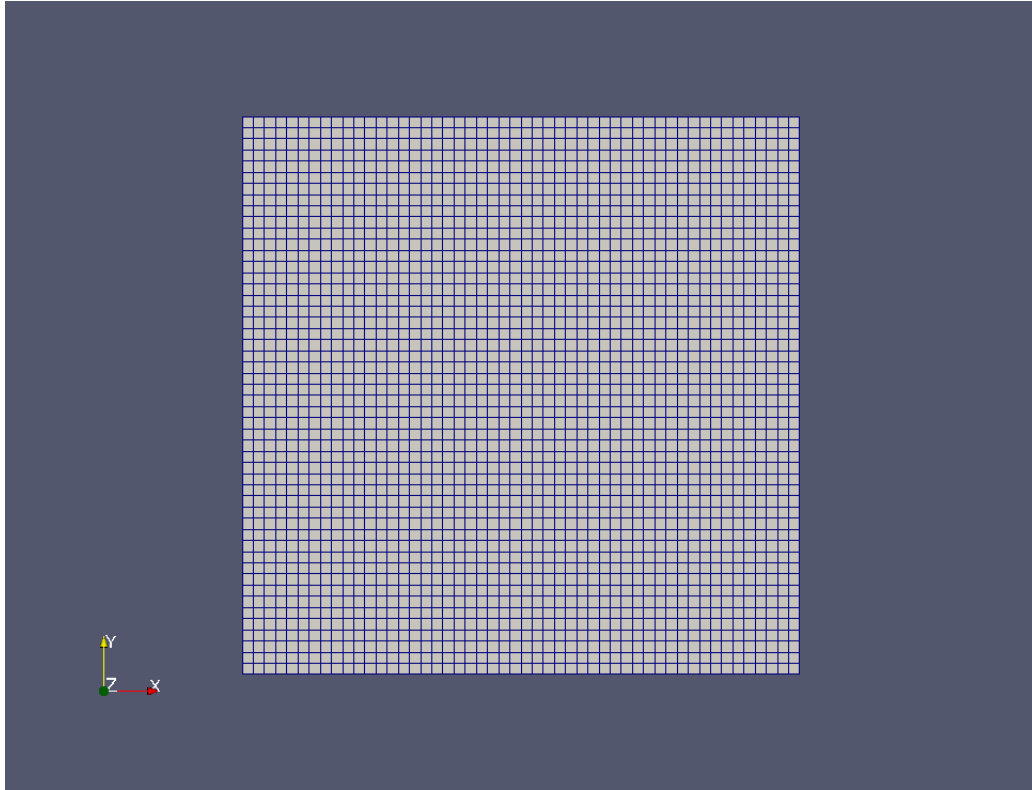


FIGURE 5.1 – Cavit  entrain e par son couvercle : Maillage structur  2500 mailles

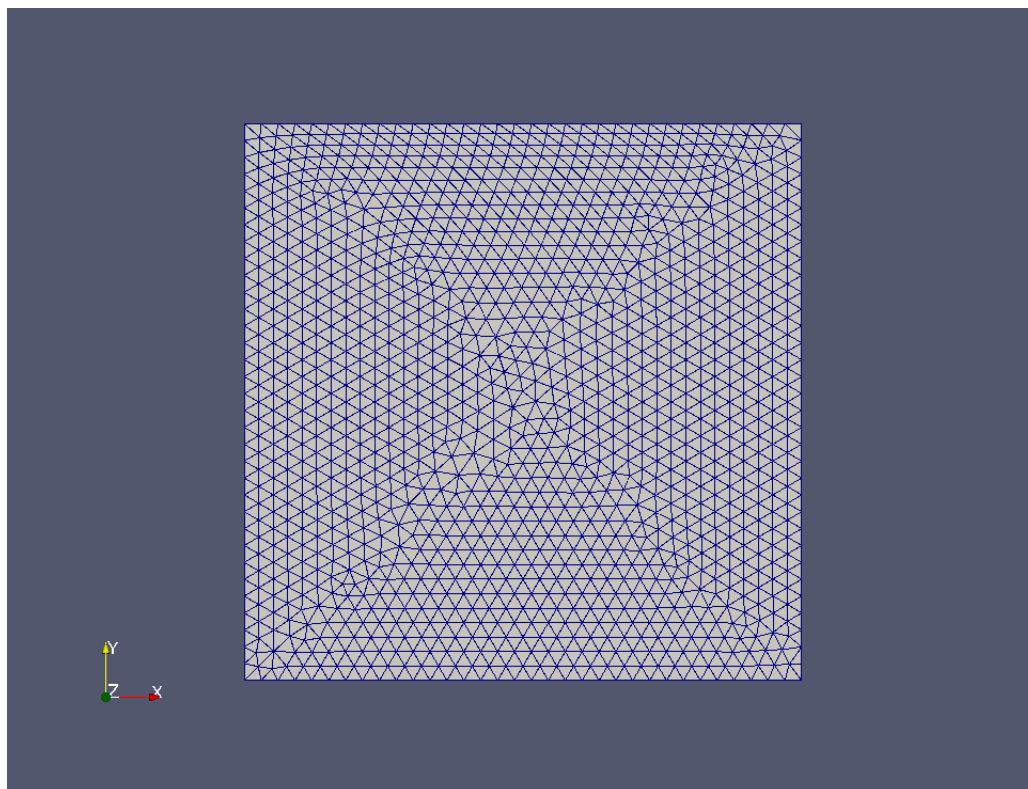


FIGURE 5.2 – Cavité entraînée par son couvercle : Maillage non-structuré  $\simeq 2600$  mailles

Les résultats du schéma upwind sont en effet peu précis (voir figures 5.3 & 5.4 ).

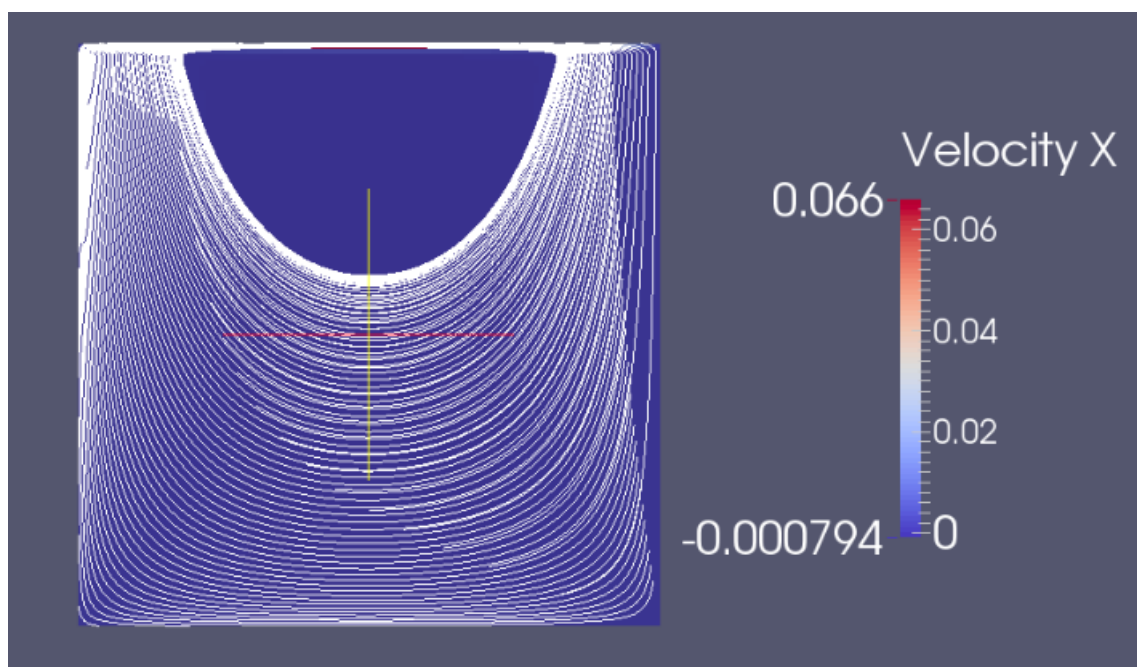


FIGURE 5.3 – Cavité entraînée par son couvercle : Schéma Upwind - Maillage structuré

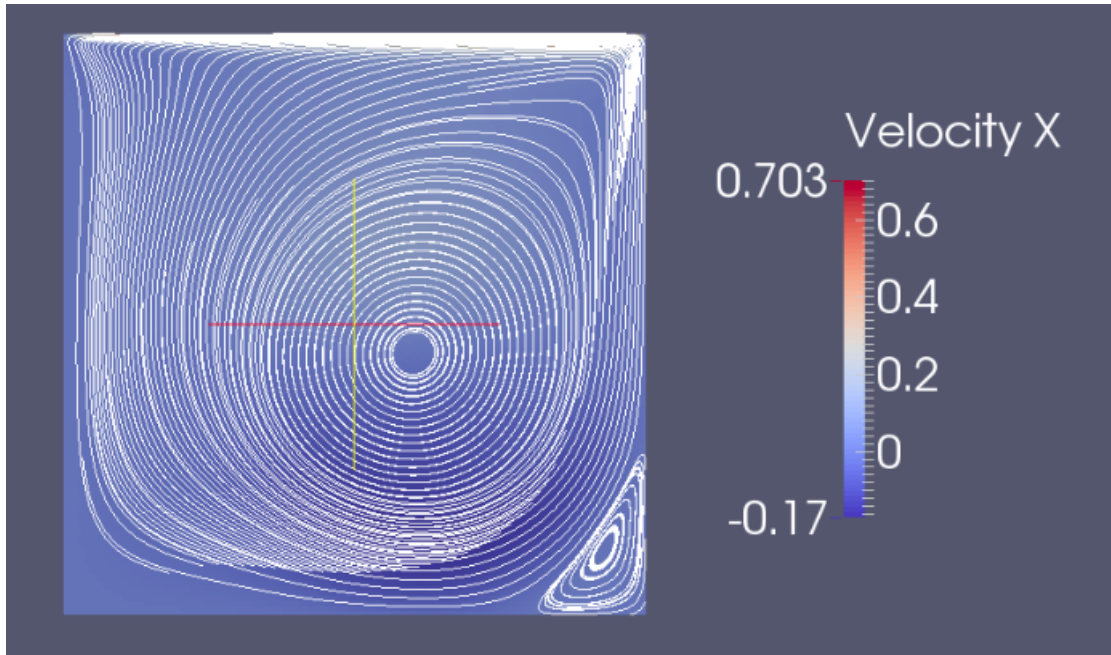


FIGURE 5.4 – Cavit  entrain e par son couvercle : Sch ma Upwind - Maillage non-structur 

Le sch ma lowMach, est lui plus pr cis mais pose des probl me de robustesse en raison de modes en  chiquiers qu'on voit tr s clairement dans les figures 5.5 & 5.6.

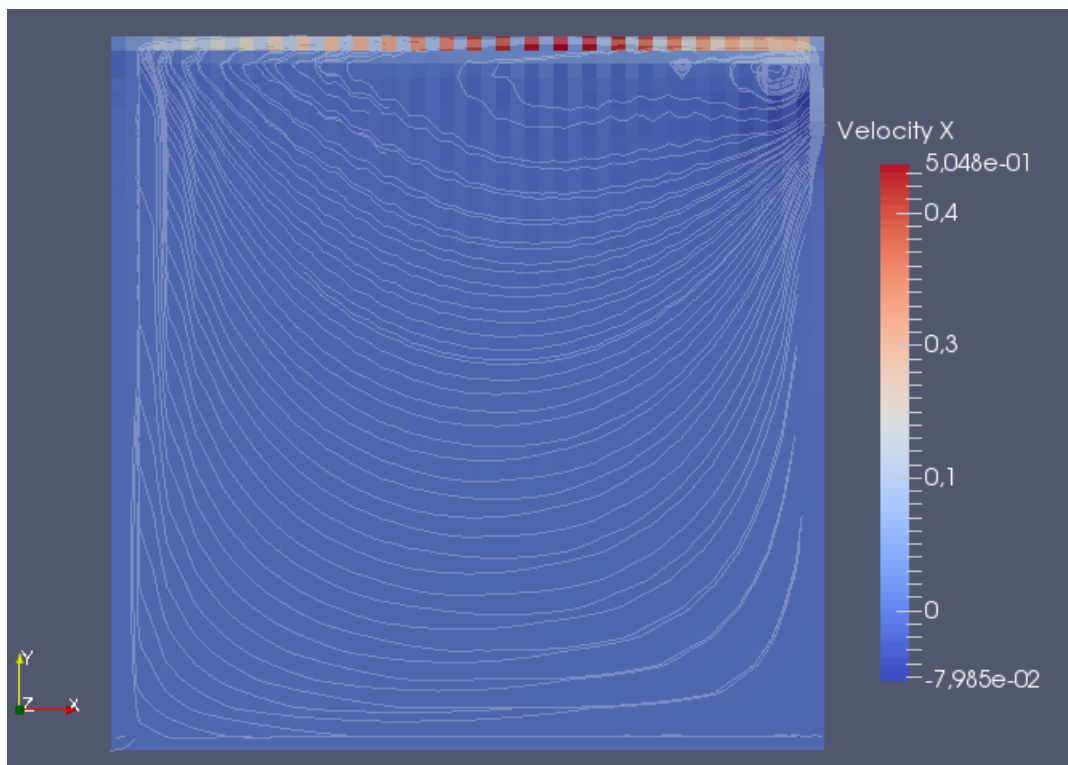


FIGURE 5.5 – Cavit  entrain e par son couvercle : Sch ma LowMach Explicite - Maillage structur 

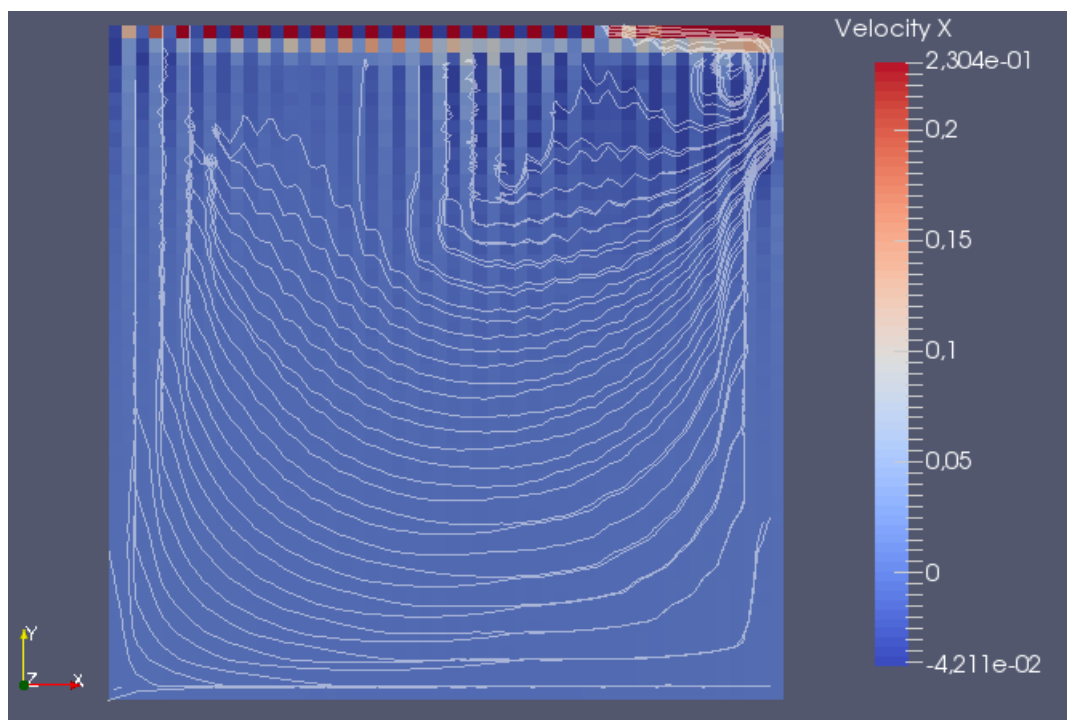


FIGURE 5.6 – Cavité entraînée par son couvercle : Schéma LowMach Implicite - Maillage structuré

Le schéma staggered, lui donne des résultats précis sans oscillations parasites (5.7 & 5.8).

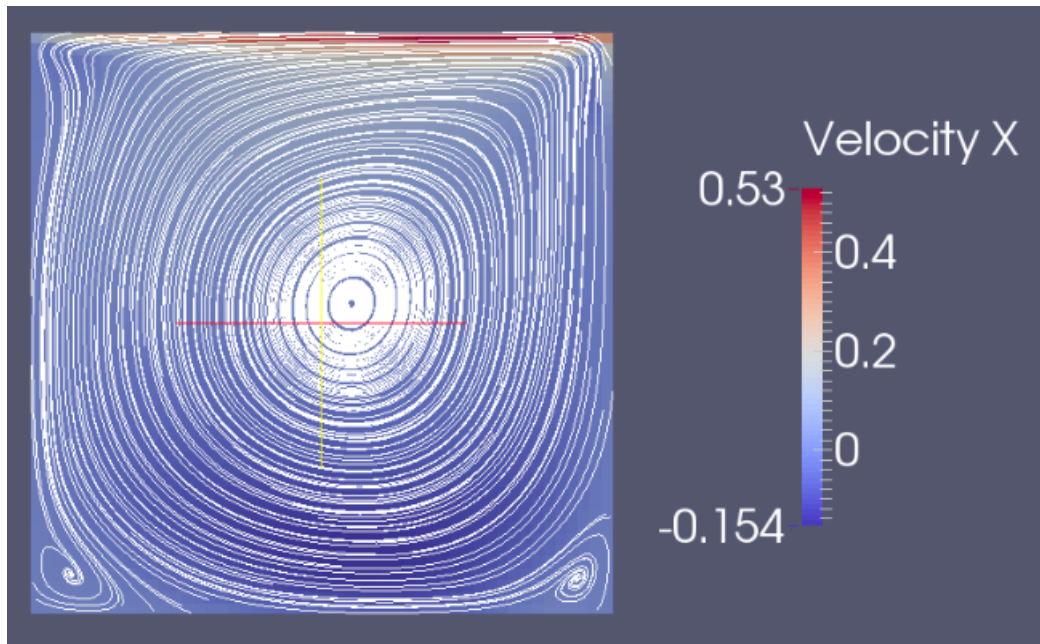


FIGURE 5.7 – Cavité entraînée par son couvercle : Schéma Staggered - Maillage structuré

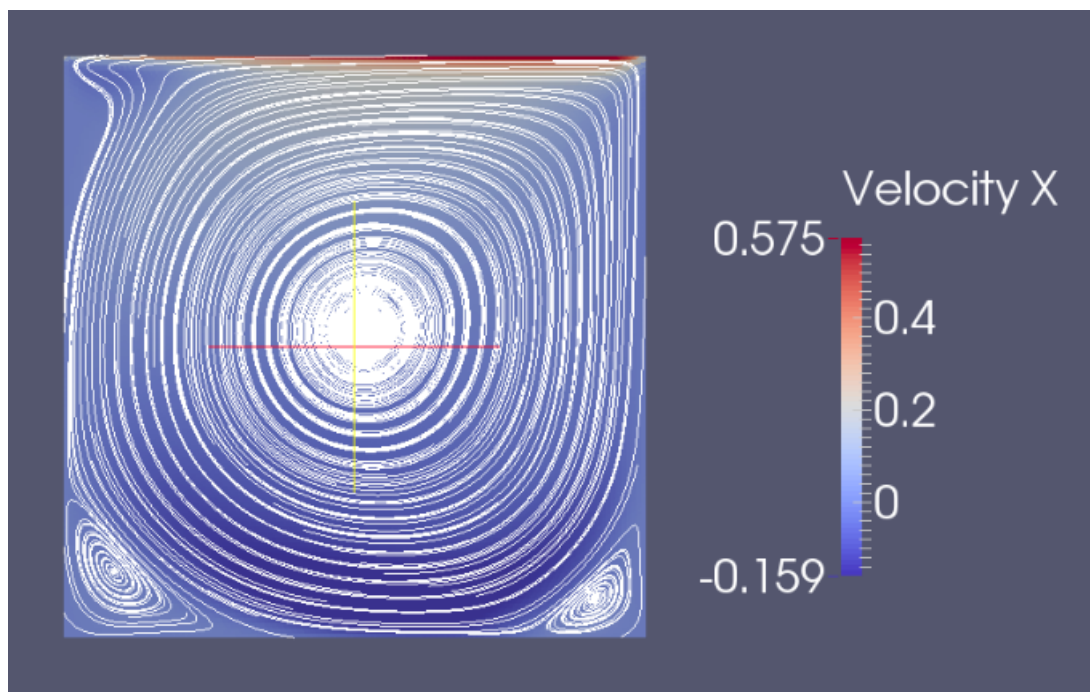


FIGURE 5.8 – Cavité entraînée par son couvercle : Schéma Staggered- Maillage non-structuré



## 5.2 Écoulement piloté par la conduction

Le deuxième cas test correspond à un problème de Riemann qui est un mélange de deux lames d'eau de températures différentes, issues par exemples de deux canaux chauffés différemment avec  $T_1 = 563K$  et  $T_2 = 623K$ , dans un domaine de  $[0, 20cm] \times [0, 40cm]$ . On impose une vitesse d'entrée dans la direction  $\vec{y}$  de valeur  $U_y = 1m/s$ , et une pression à la sortie de  $155bar$ .

On considère une viscosité de  $1.5Pa.s$ , et conductivité de  $5000W/m/s$

La figure 5.9 représente la configuration initiale du calcul. Avec un maillage de  $40 \times 80$  mailles .

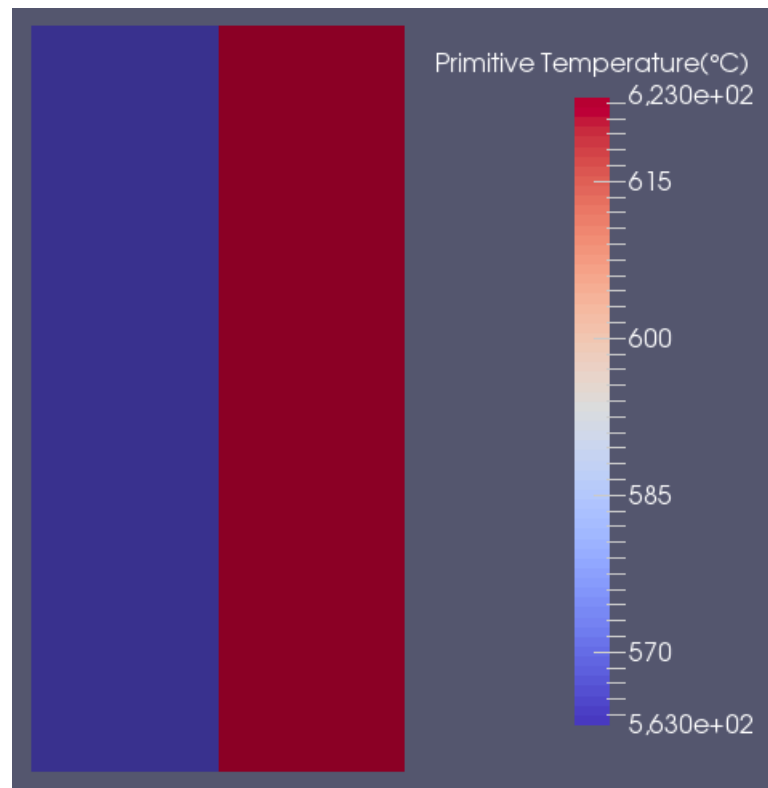


FIGURE 5.9 – Mélange de deux lames d'eau de températures  $T_1 = 563K$  et  $T_2 = 623K$  à  $t = 0s$

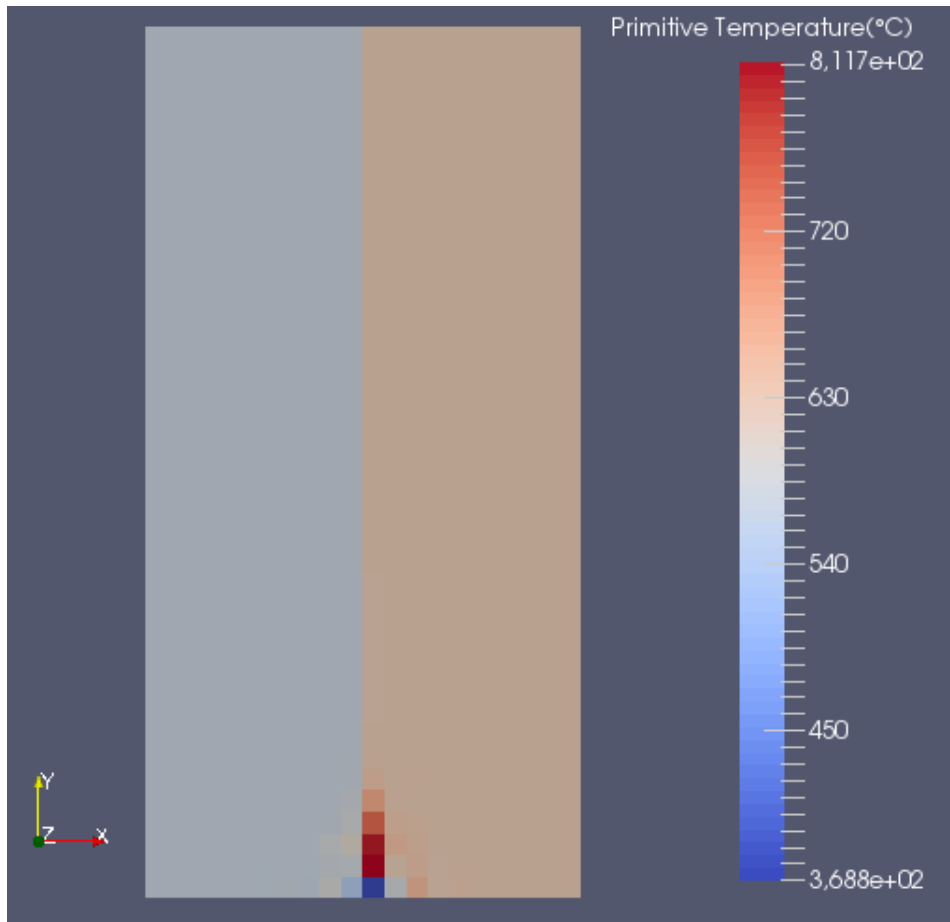


FIGURE 5.10 – Mélange de deux lames d’eau de températures différentes, Schéma : Low-Mach

On constate comme dans le cas précédent que le schéma lowMach présente de fortes oscillations. Les schéma upwind et staggered donnent des résultats assez similaires.

On remarque sur ce calcul de conduction thermique pure sans terme source, le schéma staggered (figure 5.11) est moins diffusif que le schéma upwind (figure 5.12).

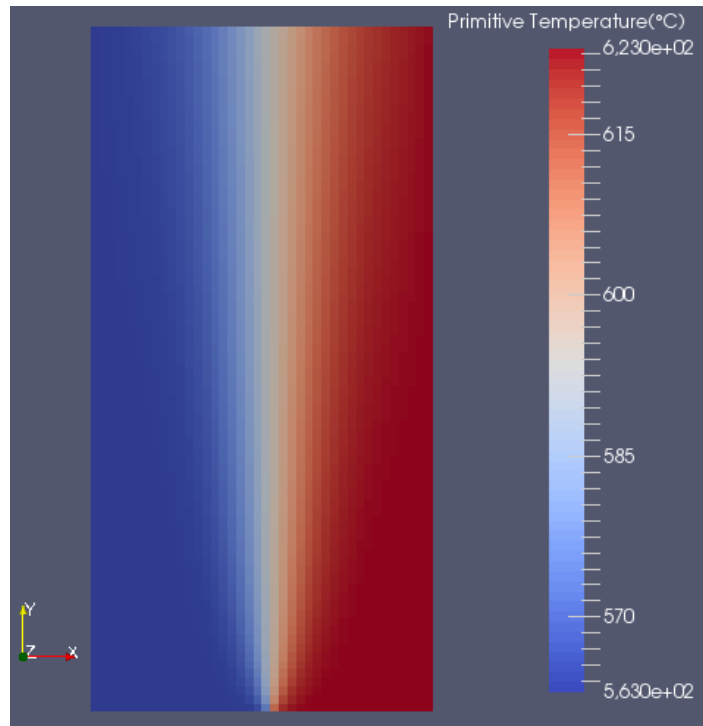


FIGURE 5.11 – Mélange de deux lames d’eau de températures différentes, Schéma : Staggered

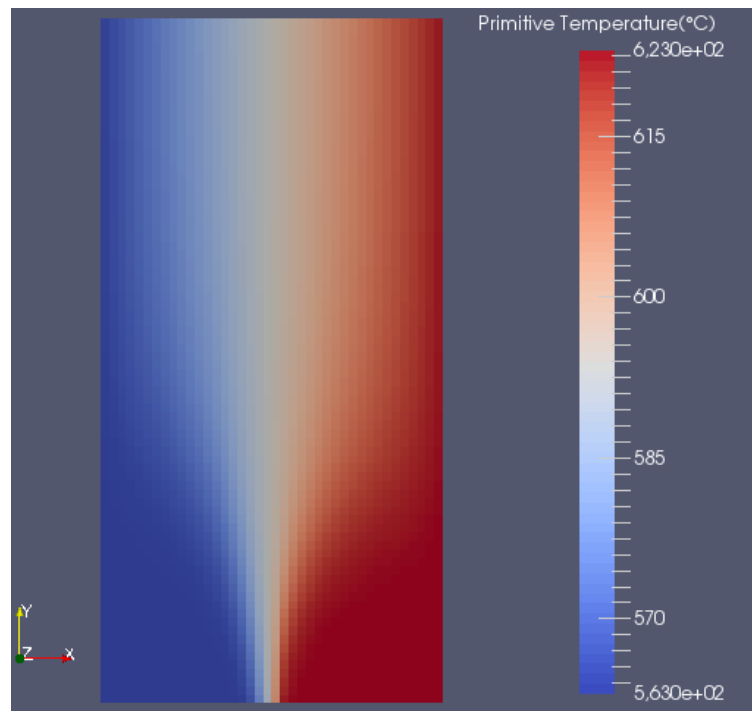


FIGURE 5.12 – Mélange de deux lames d’eau de températures différentes, Schéma : Upwind

### 5.3 Recirculation entre canaux parallèles

Le dernier cas est le plus difficile, Nous allons modéliser le coeur d'un REP900 en 2D par un domaine rectangulaire  $\Omega = [0, R] \times [0, L]$  qui est chauffé avec un flux de  $10^8 W/m^2$ , dans la zone  $\Omega = [0, \frac{R}{2}] \times [\frac{L}{4}, \frac{3L}{4}]$  avec les valeurs numériques suivantes :

$$L = 4m$$

$$R = 2m.$$

On impose une vitesse d'entrée dans la direction  $\vec{y}$  de valeur  $U_y = 0.75m/s$ , et une pression à la sortie de  $155bar$ .

On considère une gravité de  $\vec{g} = (0, -10)$

La figure 5.13 représente la configuration initiale du calcul d'un maillage de 800 mailles (20X40).

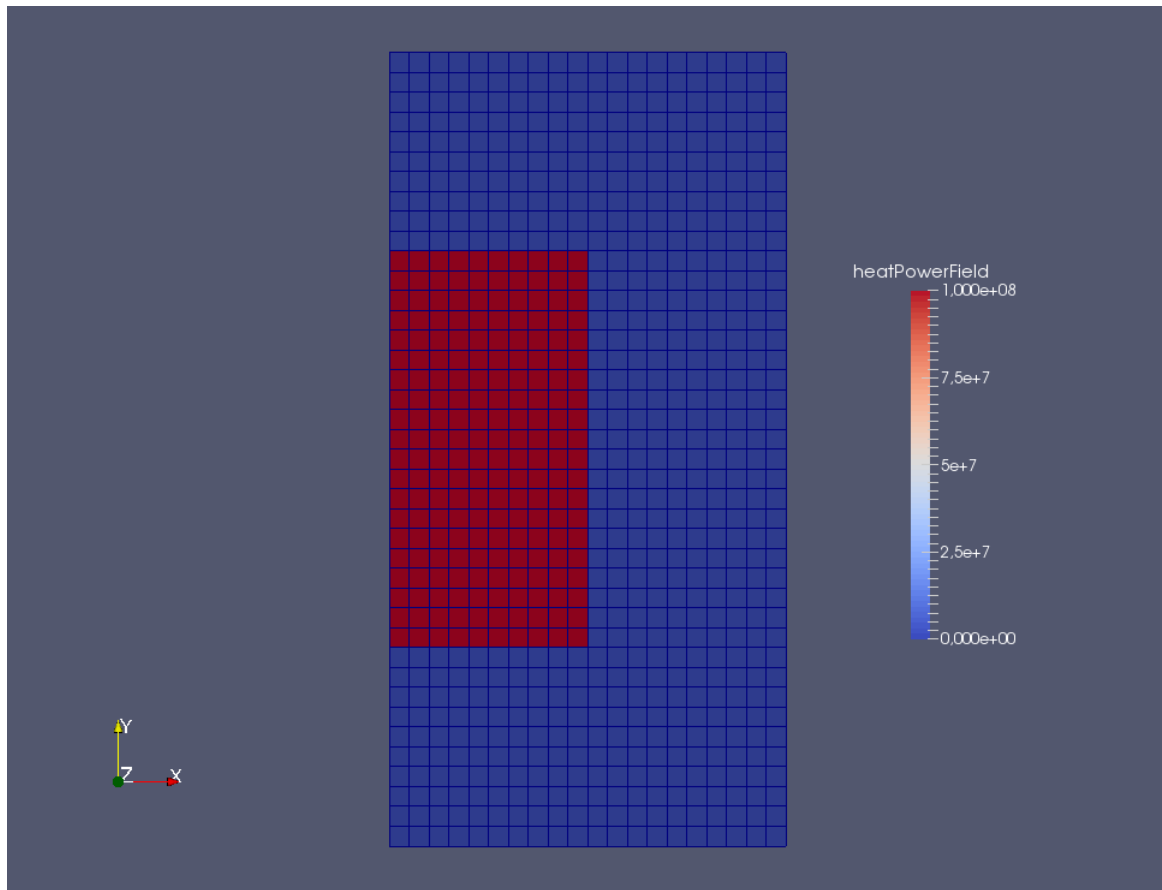


FIGURE 5.13 – Configuration initiale du calcul

#### 5.3.1 Cas monophasique

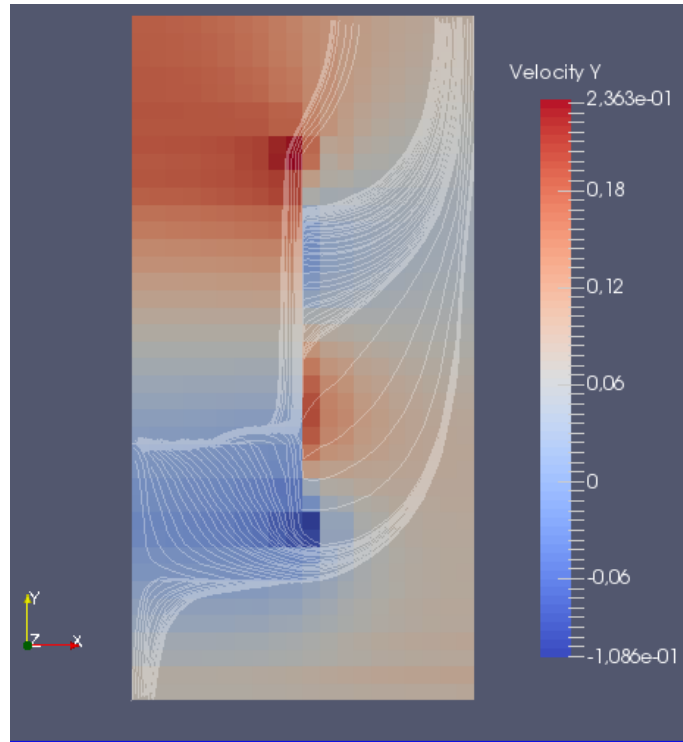


FIGURE 5.14 – Canal avec coloison chauffé partiellement : Vitesse  $U_y$ , Schéma LowMach

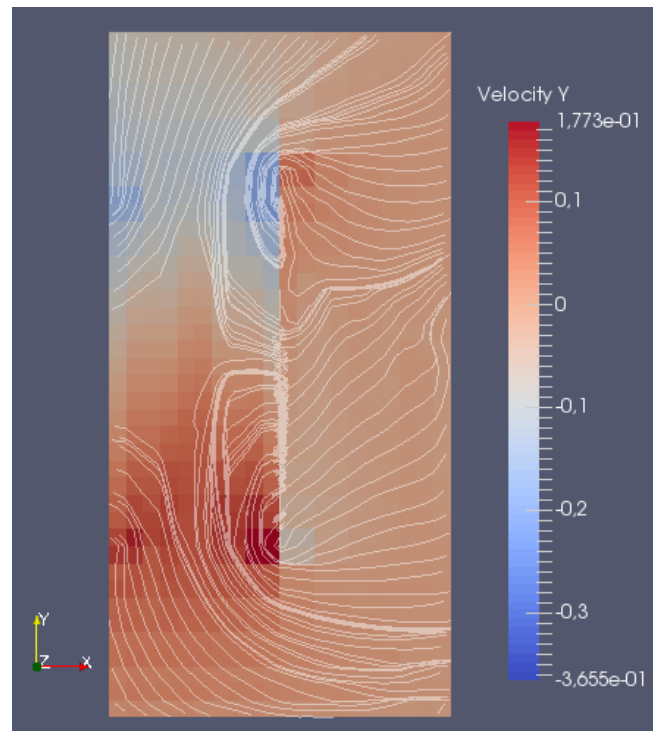


FIGURE 5.15 – Canal avec coloison chauffé partiellement : Vitesse  $U_y$ , Schéma LowMach-équilibre

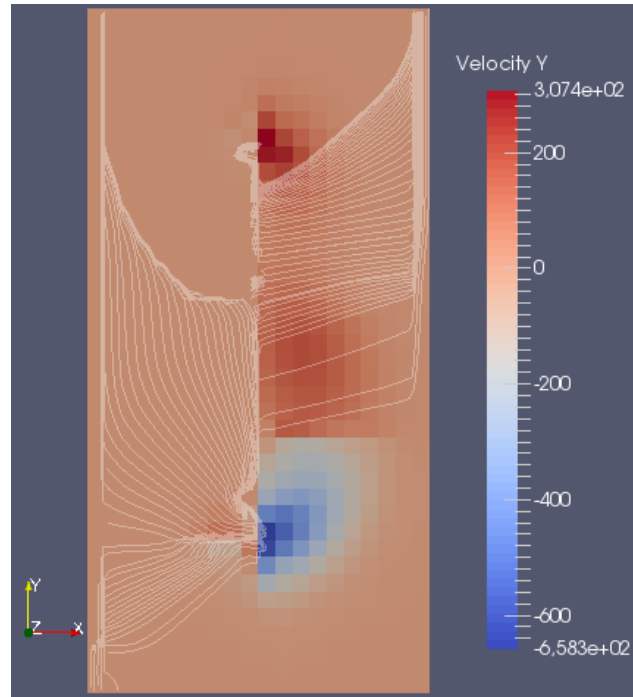


FIGURE 5.16 – Canal avec coloisson chauffé partiellement : Vitesse  $U_y$ , Schéma Staggered

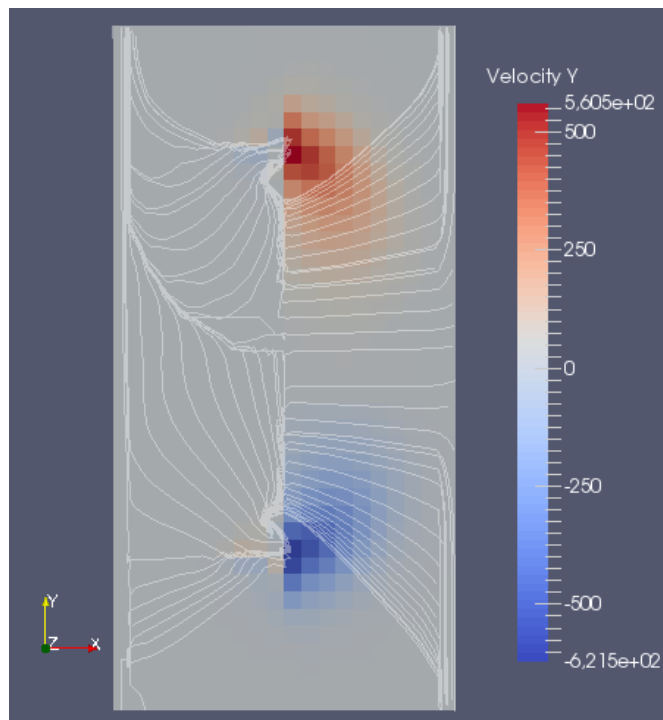


FIGURE 5.17 – Canal avec coloisson chauffé partiellement : Vitesse  $U_y$ , Schéma Staggered-équilibre

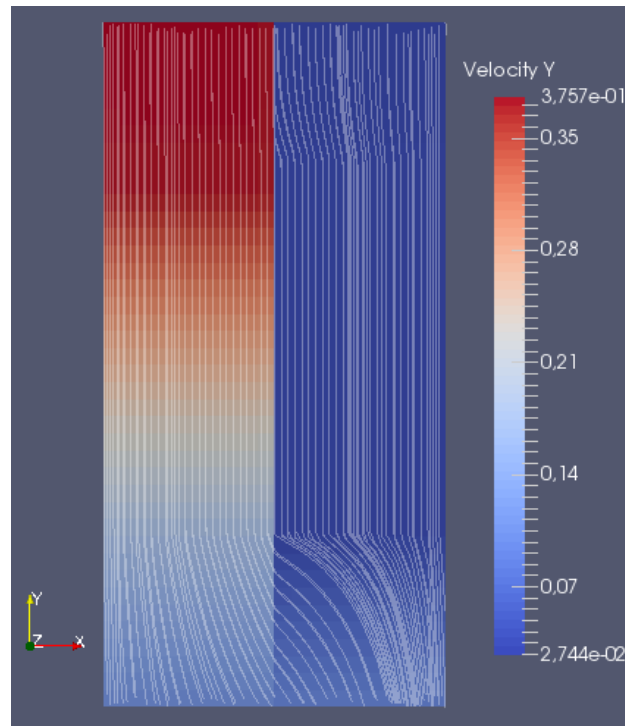


FIGURE 5.18 – Canal avec coloison chauffé partiellement : Vitesse  $U_y$ , Schéma Upwind

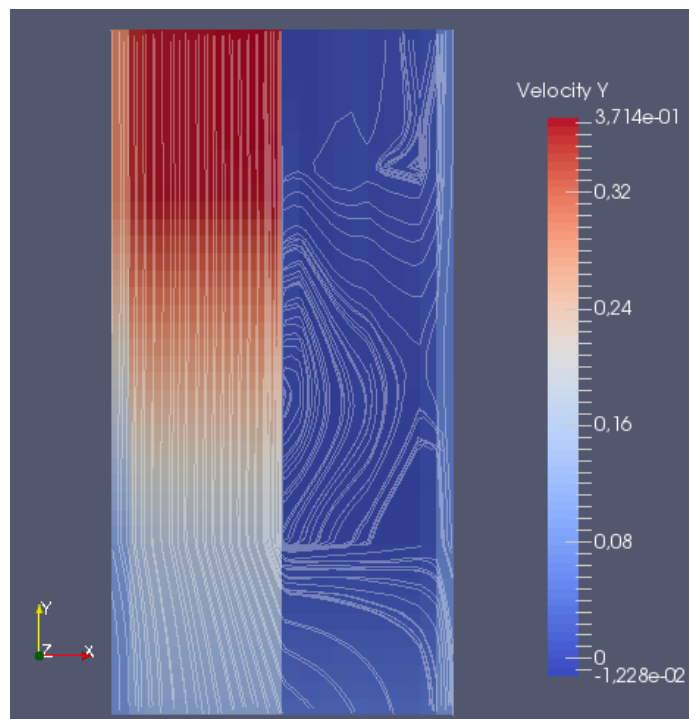


FIGURE 5.19 – Canal avec coloison chauffé partiellement : Vitesse  $U_y$ , Schéma Upwind-équilibre

### Cas monophasique : incliné

Dans cette configuration, on reprend tous les paramètres de la section 5.3, sauf la gravité, dans ce cas,  $\vec{g} = (7, -7)$  La figure 5.20 représente la configuration initiale du canal.

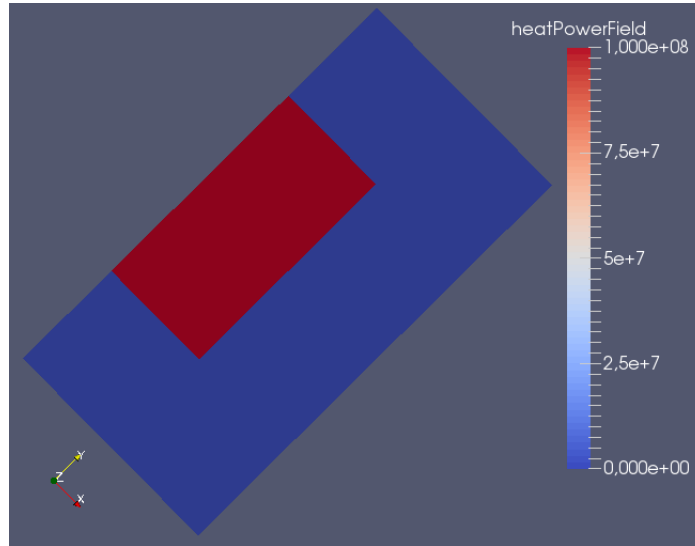


FIGURE 5.20 – Configuration initiale du canal

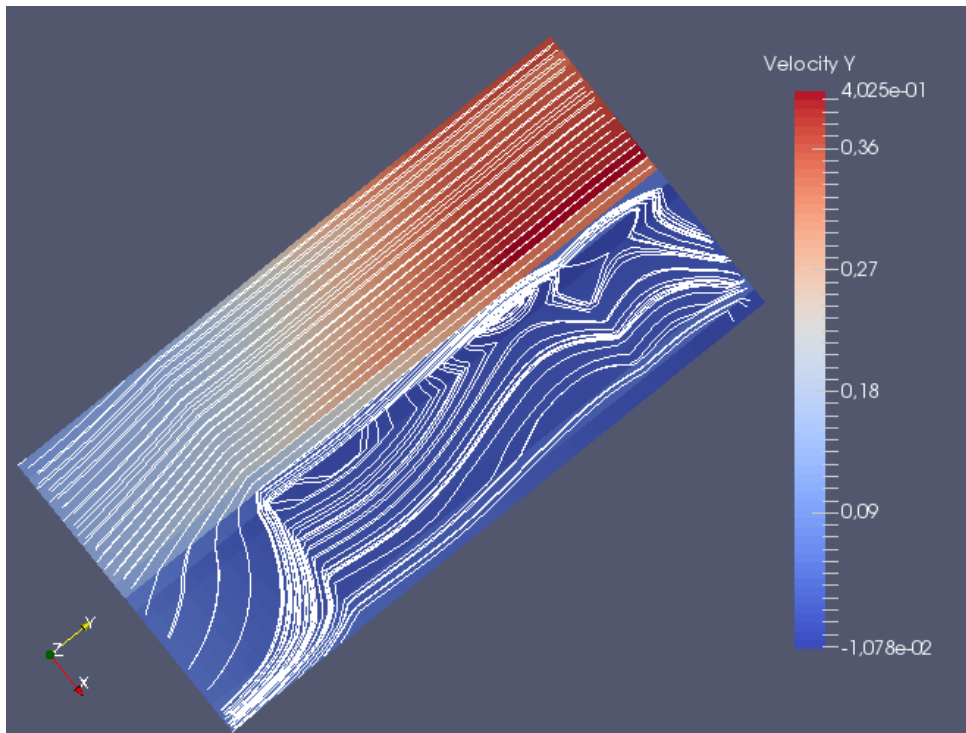


FIGURE 5.21 – Canal avec coloisin chauffé partiellement : Vitesse  $U_y$ , Schéma Upwind-équilibre



les schémas lowMach et staggered explosent en raison de la présence d'une paroi. Seul le schéma upwind va jusqu'au bout du calcul.

### 5.3.2 Cas diphasique

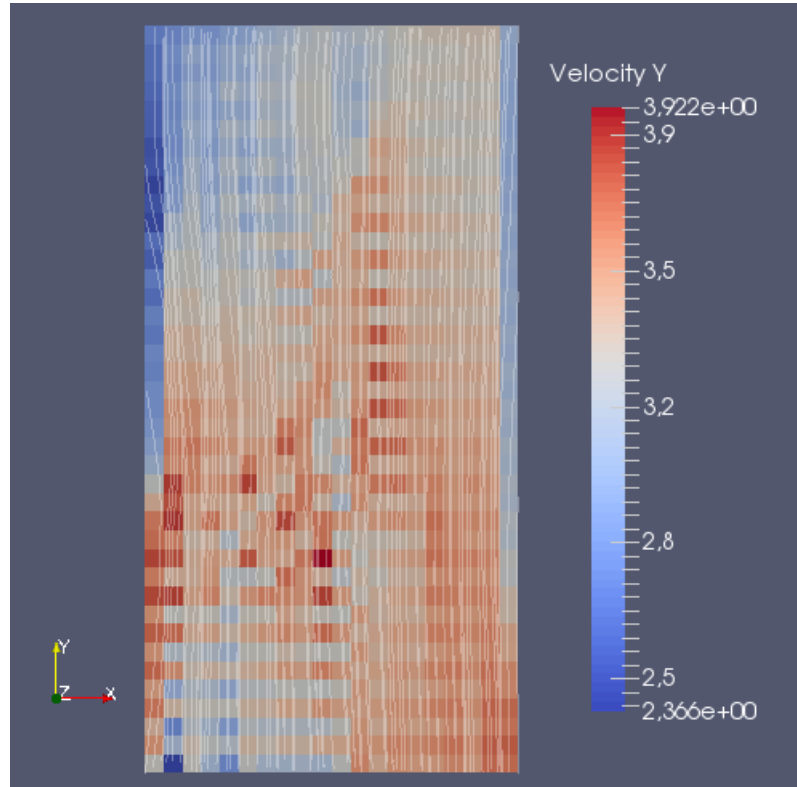


FIGURE 5.22 – Canal avec coloison chauffé partiellement : Vitesse  $U_y$ , Schéma LowMach

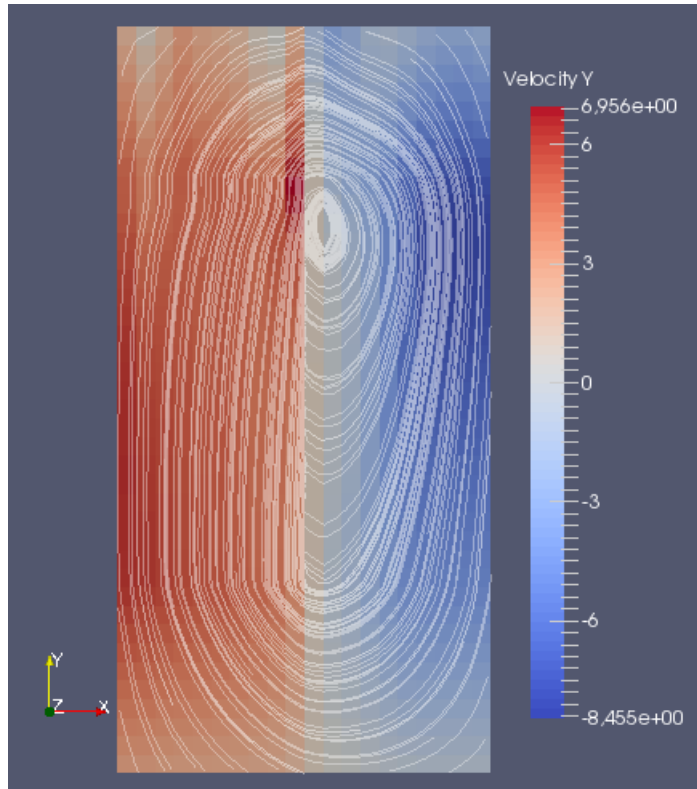


FIGURE 5.23 – Canal avec coloisson chauffé partiellement : Vitesse  $U_y$ , Schéma Upwind-équilibre

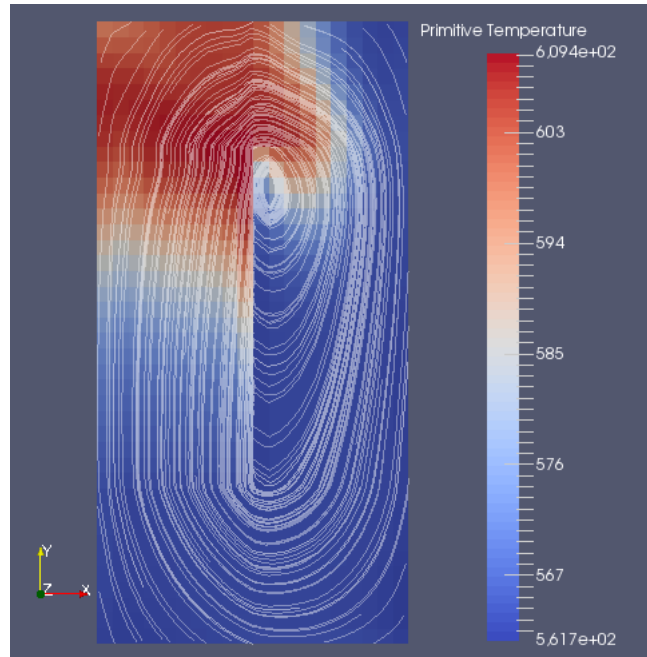


FIGURE 5.24 – Canal avec coloison chauffé partiellement : Température, lignes de courant de la vitesse  $U_y$  – Schéma Upwind-équilibre

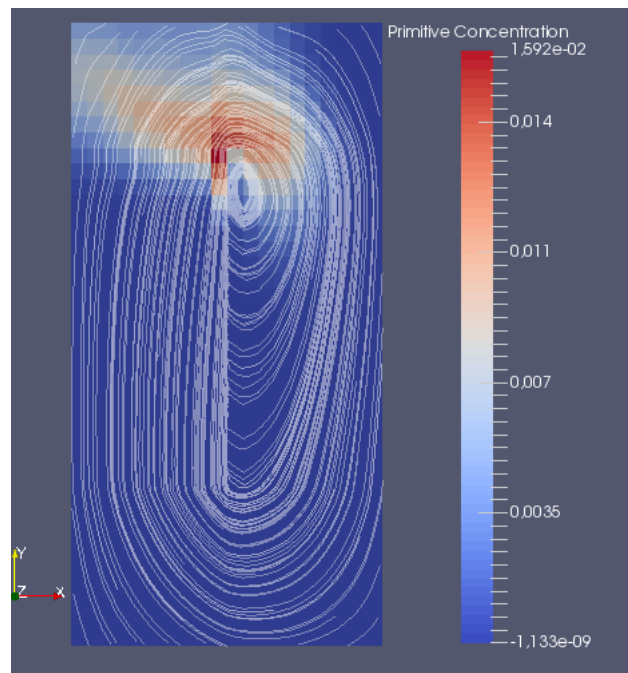


FIGURE 5.25 – Canal avec coloison chauffé partiellement : Concentration, lignes de courant de la vitesse  $U_y$  – Schéma Upwind-équilibre

### Cas diphasique incliné

On reprend la même configuration que la section 5.3.1

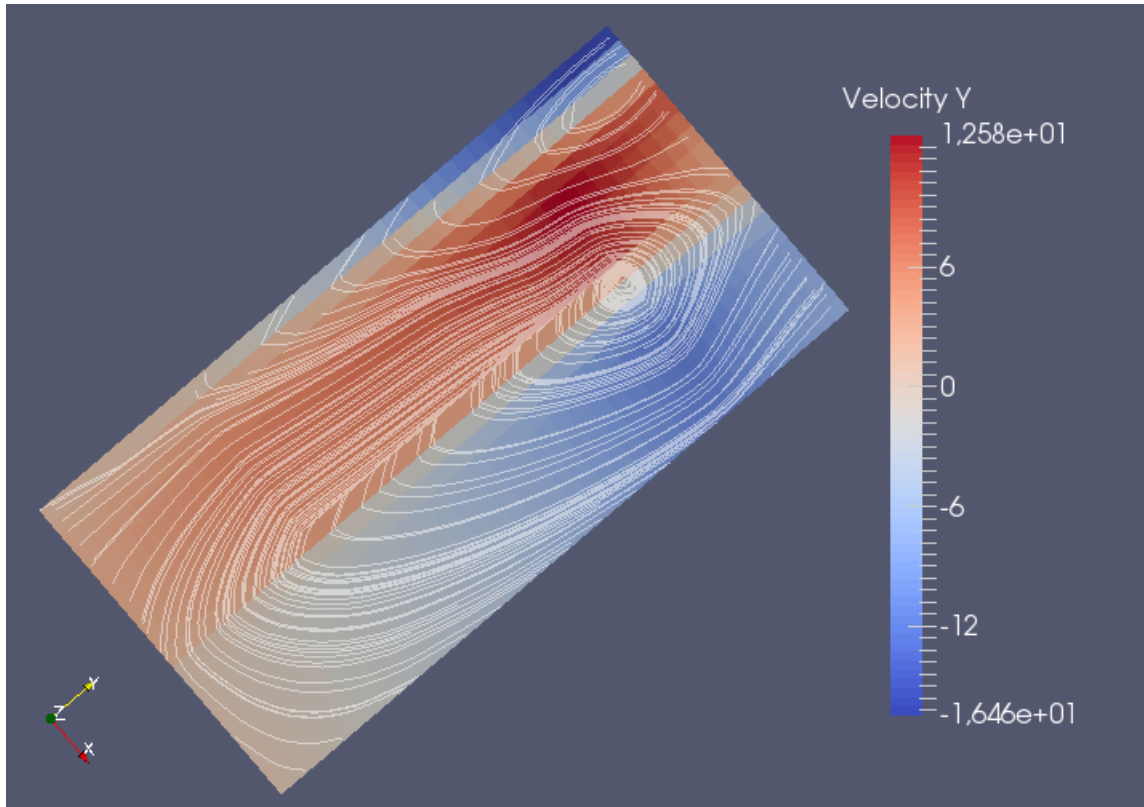


FIGURE 5.26 – Canal avec cloison chauffé partiellement : Vitesse  $U_y$ , Schéma Upwind-équilibre

Le schéma upwind well-balanced (équilibre) étaient beaucoup plus robustes et la capture de la recirculation n'a pas posé de difficulté contrairement aux schémas bas Mach qui divergent autour de la cloison.

Ces résultats montrent que la précision à faible nombre de Mach ne garantit pas un meilleur comportement des méthodes numériques

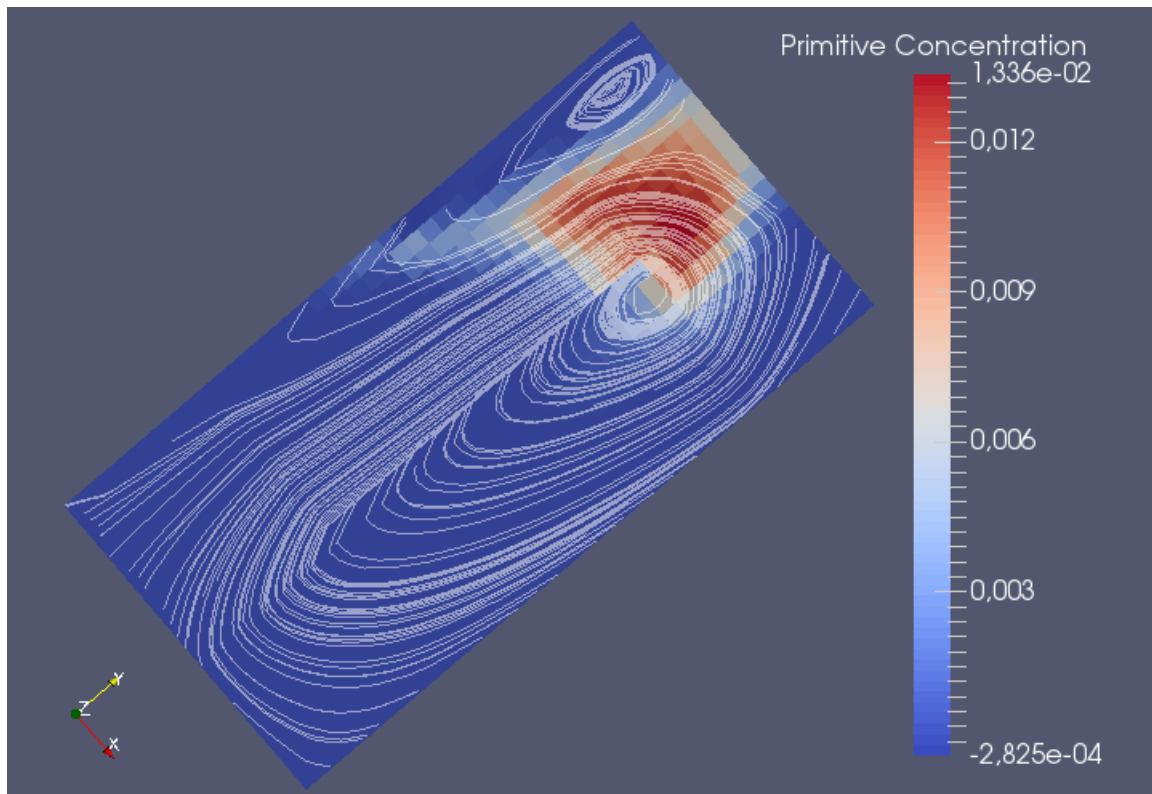


FIGURE 5.27 – Canal avec cloison incliné chauffé partiellement : Concentration, lignes de courant de la vitesse  $U_y$  – Schéma Upwind-équilibre



# Conclusions

---

## Sur le plan scientifique :

Le problème de recirculation des écoulements bouillants dans des canaux parallèles produit des instabilités physiques et numériques, qui rendent certains calculs difficiles à mener. Cela nous a amené à tester deux nouvelles approches numériques sur cette configuration.

Le premier schéma consiste à utiliser un décentrement des termes sources afin d'obtenir un schéma équilibre (upwind well-balanced).

Le deuxième schéma est inspiré des schémas sur maillage décalé et est précis à faible (staggered) nombre de Mach.

Après avoir validé nos développements sur un cas test académique (cavité entraînée par son couvercle) nous avons comme attendu, obtenu des résultats meilleurs avec le schéma précis à faible nombre de Mach (staggered).

- Sur un calcul de conduction thermique pure (pas de terme source), nous avons constaté que le schéma staggered est moins diffusif que le schéma upwind.
- Cependant dans les calculs présentant un terme de puissance thermique important, les résultats obtenus avec le schéma upwind well balanced étaient beaucoup plus robustes et la capture de la recirculation n'a pas posé de difficulté contrairement aux schémas bas Mach qui divergent autour de la cloison.

De tout cela, nous avons conclu que la précision à faible nombre de Mach ne garantit pas un meilleur comportement des méthodes numériques dans la simulation des coeurs de réacteurs nucléaires.

## Perspectives :

- Il serait intéressant d'implémenter ces schémas numériques dans le code industriel **FLICA 4**, pour évaluer leur robustesse avec des corrélations plus complexes et des lois d'état tabulées.
- Une suite de ce travail serait l'introduction d'une porosité et de perte de charges singulières pour étudier la robustesse des schémas proposés.
- Une analyse du problème de distribution de débit entre canaux parallèles permettrait une validation de la physique de nos résultats numériques.

**Sur le plan personnel** : Issue d’une formation mathématique, ce stage m’a permis d’approfondir considérablement mes connaissances en physique, notamment en mécaniques des fluides. Il m’a permis aussi de découvrir et prendre en main plusieurs librairies Open Source, de gestion de maillage, calcul scientifique et de visualisation.

C’était également l’occasion d’échanger et de collaborer avec des ingénieurs du laboratoire LGLS, ce qui m’a été particulièrement bénéfique.

J’ai été également contrainte à surmonter quelques obstacles dus aux standards de sécurité informatique particulièrement élevés au centre CEA de Saclay.

J’estime finalement que le bilan du stage est très positif, car en plus d’avoir atteint les objectifs fixés initialement, j’ai bénéficié d’une excellente expérience professionnelle. J’ai de plus élargi mon réseau professionnel en rencontrant des ingénieurs chercheurs et des thésards.



# Annexe

## 7.1 Discrétisation en temps

### 7.1.1 Schéma explicite

Notons  $U_i^{n+1}$  les variables conservatives à l'instant  $n + 1$ , dans la cellule de contrôle  $C_i$ . Le schéma explicite en temps, d'ordre 1 en espace, s'écrit à partir de l'équation (3.2) la discrétisation suivante :

$$U_i^{n+1} = U_i^n - \frac{\Delta t}{\text{vol}(C_i)} \sum_{j \in v(i)} S_{ij} \tilde{F}(U_i^n, U_j^n, \mathbf{n}_{ij}) + \Delta t S(U_i^n) \quad (7.1)$$

L'équation (7.1) donne explicitement les variables conservatives à l'instant  $n + 1$ , en fonction de ces mêmes variables à l'instant  $n$ . Cependant, l'utilisation de cette formule explicite nécessite des pas de temps petits limités par une condition CFL du type :

$$\Delta t \leq \frac{\Delta x}{|\lambda_M|}, \quad \Delta t \leq \frac{1}{K},$$

où  $\lambda_M$  est la plus grande valeur propre des matrices  $\nabla F \mathbf{n}_{ij}$ . En cas d'égalité des vitesses,  $\lambda_M = \|\mathbf{u}_m\|^{10} + a_m$  où  $a_m$  est la vitesse du son du mélange diphasique définie par

$$a_m^2 = \kappa(H_m - \frac{1}{2}u^2) + \chi + c_m \xi \quad (7.2)$$

avec  $\chi, \xi$  et  $\kappa$  définis dans [6].

### 7.1.2 Schéma implicite

L'utilisation de schémas implicites permet de s'affranchir des conditions de stabilité dues au pas de temps et conduit à des méthodes numériques plus efficaces. Le schéma implicite dérivé de (3.2) s'écrit :

$$U_i^n = U_i^{n-1} - \frac{\Delta t}{\text{vol}(C_i)} \sum_{j \in v(i)} S_{ij} \tilde{F}(U_i^n, U_j^n, \mathbf{n}_{ij}) + \Delta t S(U_i^n) \quad (7.3)$$

10.  $\mathbf{u}_m = \frac{\alpha_l \rho_l \mathbf{u}_l + \alpha_v \rho_v \mathbf{u}_v}{\alpha_l \rho_l + \alpha_v \rho_v}$

L'équation (7.3) est écrite pour toutes les cellules de contrôle  $C_i$  définit un système d'équation non linéaire nécessiterait une méthode itérative coûteuse en temps de calcul.

Il faut donc résoudre une équation non linéaire  $f(\mathcal{U}) = 0$ ,  
où  $\mathcal{U} = (U_1, \dots, U_N)$  et  $f = (f_1, \dots, f_N)$  où  $f_i(\mathcal{U})$  est donné par

$$f_i(\mathcal{U}) = \frac{U_i - U_i^n}{\Delta t} + \frac{1}{v_i} \sum_{j \in \nu(i)} s_{ij} \tilde{F}(U_i, U_j, \vec{n}_{ij}) - S(U_i).$$

Classiquement l'équation  $f(\mathcal{U}) = 0$  est résolue par des itérations de Newton

$$\nabla f(\mathcal{U}^k) (\mathcal{U}^{k+1} - \mathcal{U}^k) = -f(\mathcal{U}^k).$$

Chaque itération de Newton nécessite un calcul du gradient de  $f$  et la résolution d'un système linéaire.

Le calcul exacte du gradient de  $f$  est impossible car dériver la fonction  $D_{ij}(\mathcal{U})$  est très complexe. Il n'est cependant pas nécessaire de calculer exactement le gradient pour obtenir la convergence du schéma de Newton. Si la suite  $\mathcal{U}^k$  converge, alors la limite  $\mathcal{U}^\infty$  vérifie  $f(\mathcal{U}^\infty) = 0$ .

Dans Flica4 afin de calculer une approximation du gradient de l'opérateur de convection, on s'inspire de la relation

$$\sum_{j \in \nu(i)} s_{ij} \tilde{F}(U_i, U_j, \vec{n}_{ij}) = - \sum_{j \in \nu(i)} s_{ij} A_{ij}^-(U_i, U_j) (U_i - U_j), \quad (7.4)$$

où  $A_{ij}^- = \frac{1}{2}(A_{Roe}(U_i, U_j) - |A_{Roe}(U_i, U_j)|)$ , où  $A_{Roe}(U_i, U_j)$  est la matrice de Roe du flux  $F$  entre les états  $U_i$  et  $U_j$ .

L'équation (7.4) vient du fait qu'en utilisant la définition de la matrice de Roe (7.8), (3.9) se réécrit

$$\tilde{F}(U_i, U_j, \vec{n}_{ij}) = F(U_i) \vec{n}_{ij} - A^-(U_i, U_j) \frac{U_i - U_j}{2},$$

La matrice de Roe étant inconnue dans le modèle de Flica4, on utilise l'approximation,

$$A_{ij}^- \approx A_{ij}^{m-}$$

avec  $A_{ij}^{m-} = \frac{1}{2}(A_{Roe}^m(U_i, U_j) - |A_{Roe}^m(U_i, U_j)|)$

où  $A_{Roe}^m(U_i, U_j)$  est la matrice de Roe du flux  $F^m$  (équation 7.9) entre les états  $U_i$  et  $U_j$ .

Le schéma itératif final s'écrit

$$\begin{aligned} \frac{U_i^{k+1} - U_i^k}{\Delta t} - \frac{1}{v_i} \sum_{j \in \nu(i)} s_{ij} A_{ij}^{m-}(U_i^k, U_j^k) ((U_i^{k+1} - U_i^k) - (U_j^{k+1} - U_j^k)) \\ - \nabla S(U_i^k)(U_i^{k+1} - U_i^k) = - \frac{U_i^k - U_i^n}{\Delta t} - \frac{1}{v_i} \sum_{j \in \nu(i)} s_{ij} \tilde{F}(U_i^k, U_j^k, \vec{n}_{ij}) + S(U_i^k). \end{aligned} \quad (7.5)$$

Lorsque nous sommes en présence d'une face située à la frontière du domaine, l'équation (7.5) impose de connaître  $U_b(U_i^k)$  et d'approcher  $U_b(U_i^{k+1}) - U_b(U_i^k)$ . L'approche la plus simple consiste à traiter les conditions aux limites en utilisant la linéarisation

$$U_b(U_i^{k+1}) - U_b(U_i^k) \approx \nabla U_b(U_i^k)(U_i^{k+1} - U_i^k), \quad (7.6)$$

qui nécessite le calcul de  $U_b(U_i^k)$  et  $\nabla U_b(U_i^k)$  pour chaque condition limite (annexe 7.2).

## 7.2 Conditions aux limites

Il existe deux types de conditions aux limites en mécanique des fluides : celles qui correspondent à des parois solides (débit normal nul) et celles qui correspondent à des frontières du domaine de calcul fixées par l'utilisateur.

Le traitement des conditions aux limites dans un schéma de volumes finis est rendu délicat par le fait que les variables sont définies dans les volumes et non aux interfaces des mailles.

Pour imposer les conditions aux limites physiques à la frontière du domaine de calcul, on rajoute une maille fictive ( $U_b, V_b$ ) de l'autre côté de l'interface. Cette maille a un rôle purement informatique : elle permet de traiter les interfaces limites de la même façon que les interfaces intérieures ( $U, V$ ). Une fois l'état dans la maille fictive déterminé, il ne reste plus en effet qu'à calculer le flux numérique comme s'il s'agissait d'une interface intérieure.

Le problème consiste maintenant à affecter à la maille fictive un état tel que le flux numérique calculé à l'interface satisfasse les conditions aux limites requises. D'un point de vue mathématique, le nombre de conditions aux limites que l'on peut imposer à une interface donnée doit correspondre au nombre de caractéristiques entrantes dans le domaine.

Pour les équations de FLICA4, Nous considérons 4 types de conditions aux limites.

— Neumann :

$$V_b(V) = V, \quad U_b(U) = U$$

— Paroi :

$$V_b : \begin{pmatrix} c_v \\ p \\ \mathbf{u} \\ T \end{pmatrix} \rightarrow \begin{pmatrix} c_v \\ p \\ -\mathbf{u} \\ T \end{pmatrix}, \quad U_b : \begin{pmatrix} \rho_m \\ \rho_m c_v \\ \rho_m \mathbf{u}_m \\ \rho_m E_m \end{pmatrix} \rightarrow \begin{pmatrix} \rho_m \\ \rho_m c_v \\ -\rho_m \mathbf{u}_m \\ \rho_m E_m \end{pmatrix}$$

— Entrée : concentration  $c_{ve} = 1 - c_{le}$ , vitesse  $\mathbf{u}_e$ , température  $T_e$  imposés

$$V_b : \begin{pmatrix} c_v \\ p \\ \mathbf{u} \\ T \end{pmatrix} \rightarrow \begin{pmatrix} c_{ve} \\ p \\ \mathbf{u}_e \\ T_e \end{pmatrix}, \quad U_b : \begin{pmatrix} \rho \\ \rho c_v \\ \rho \mathbf{u} \\ \rho E \end{pmatrix} \rightarrow \begin{pmatrix} \rho_e \\ \rho_e c_{ve} \\ \rho_e \mathbf{u}_e \\ \rho_e E_e \end{pmatrix}$$

avec

$$\begin{aligned} \rho_e &= \frac{\rho_v(p, T_e) \rho_l(p, T_e)}{\rho_v(p, T_e) c_{le} + \rho_l(p, T_e) c_{ve}} \\ E_e &= c_{ve} (e_v(T_e) + \frac{1}{2} |\mathbf{u}_{ve}|^2) + c_{le} (e_l(T_e) + \frac{1}{2} |\mathbf{u}_{le}|^2) \\ \mathbf{u}_{ve} &= \mathbf{u}_e + c_{le} \mathbf{u}_r(c_{ve}, \mathbf{u}_e, \rho_e) \\ \mathbf{u}_{ve} &= \mathbf{u}_e - c_{ve} \mathbf{u}_r(c_{ve}, \mathbf{u}_e, \rho_e). \end{aligned}$$

— Sortie : pression  $p_s$  imposée

$$V_b : \begin{pmatrix} c_v \\ p \\ \mathbf{u} \\ T \end{pmatrix} \rightarrow \begin{pmatrix} c_v \\ p_s \\ \mathbf{u} \\ T \end{pmatrix}, \quad U_b : \begin{pmatrix} \rho \\ \rho c_v \\ \rho \mathbf{u} \\ \rho E \end{pmatrix} \rightarrow \begin{pmatrix} \rho_s \\ \rho_s c_v \\ \rho_s \mathbf{u} \\ \rho_s E \end{pmatrix}$$

avec

$$\rho_s = \frac{\rho_v(p_s, T) \rho_l(p_s, T)}{\rho_v(p_s, T) c_l + \rho_l(p_s, T) c_v}.$$

## 7.3 Annexe : linéarisation de Roe

### 7.3.1 Définition de la matrice de ROE

La linéarisation de Roe est une technique introduite par P. Roe [21] pour linéariser de façon précise les systèmes de lois de conservation de la forme :

$$\partial_t U + \partial_x f(U) = 0, \tag{7.7}$$

Soit  $A(U) = \nabla_U f(U)$  la matrice Jacobienne du flux, laquelle est diagonalisable dans  $\mathbb{R}$  dans une base complète de vecteurs propres réels.

Nous écrivons sous une forme non conservative (7.7)

$$\partial_t U + A(U) \partial_x U = 0$$

*Remarque.* Cette écriture est correcte pour les solutions régulières

La linéarisation de ROE se fonde sur la définition

**Définition 7.3.1.1** Une matrice  $A(U, V)$  vérifiant

1.  $A(U, V)$  est diagonalisable dans  $\mathbb{R}$  dans une base complète de vecteurs propres réels,
2. pour tout  $(U, V)$  on a  $f(U) - f(V) = A(U, V)(U - V)$ ,
3. pour tout  $U$  on a  $A(U, U) = A(U)$ ,

est une matrice de Roe associée au système 7.7.

Afin de calculer le flux entre deux cellules voisines  $i$  et  $j$ , l'idée originelle de Ph. Roe dans [21] est de linéariser localement le problème (2.9) entre les deux états respectifs  $U_i$  et  $U_j$ . Ph. Roe cherche donc une interpolation linéaire du flux  $F$  de la forme

$$\tilde{F}(U) = F(U_i) + A_{Roe}(U_i, U_j)(U_i - U_j).$$

Pour cela il faut pouvoir construire une matrice dite “de Roe”  $A_{Roe}(U_i, U_j)$  vérifiant

$$F(U_i) - F(U_j) = A_{Roe}(U_i, U_j)(U_i - U_j). \quad (7.8)$$

Bien qu'une telle matrice existe toujours, il n'est pas toujours aisé de la construire. C'est le cas pour le modèle (2.9) et on a alors recours à une matrice approchée de Roe.

Pour le calcul de la matrice de Roe du modèle à quatre équations il faut résoudre l'équation (7.8) associée au flux  $F$ . Ce calcul étant difficile, les développeurs de Flica4 ont négligé la contribution de l'écart de vitesse  $\vec{u}_r$  dans le flux  $F$ . Ce choix est adapté aux écoulements cocourant où la vitesse relative  $\vec{u}_r$  est négligeable devant la vitesse moyenne  $\vec{u}_m$ .

### 7.3.2 La matrice de Roe dans Flica4

Le flux  $F$  (équation 2.10) peut se décomposer en deux contributions, l'une liée à l'écoulement moyen et l'autre au mouvement relatif :

$$F = F_m + F_r,$$

avec

$$F_m(U) = {}^t \begin{pmatrix} \rho_m {}^t \vec{u}_m \\ \rho_m c_v {}^t \vec{u}_m \\ \rho_m \vec{u}_m \otimes \vec{u}_m + p \mathbb{I}_d \\ \rho_m H_m {}^t \vec{u}_m \end{pmatrix} \quad (7.9)$$

$$F_r(U) = {}^t \begin{pmatrix} 0 \\ \rho_m c_v c_l {}^t \vec{u}_r \\ \rho_m c_v c_l \vec{u}_r \otimes \vec{u}_r \\ \rho_m c_v c_l (H_v - H_l) {}^t \vec{u}_r \end{pmatrix}, \quad (7.10)$$

et

$$\begin{aligned} c_v &= \frac{\alpha_v \rho_v}{\alpha_v \rho_v + \alpha_l \rho_l}, & c_l &= \frac{\alpha_l \rho_l}{\alpha_v \rho_v + \alpha_l \rho_l} \\ \rho_m &= \alpha_v \rho_v + \alpha_l \rho_l \\ \vec{u}_m &= \frac{\alpha_v \rho_v \vec{u}_v + \alpha_l \rho_l \vec{u}_l}{\alpha_v \rho_v + \alpha_l \rho_l} \\ H_m &= \frac{\alpha_v \rho_v H_v + \alpha_l \rho_l H_l}{\alpha_v \rho_v + \alpha_l \rho_l} = E_m + \frac{p}{\rho_m}. \end{aligned}$$

le Logiciel Flica4 néglige la contribution associée à la composante relative  $F_r$  du flux convectif  $F$  et utilise la matrice de Roe  $A_{Roe}^m(U_i, U_j)$  associée à la composante moyenne  $F_m$ . Par définition, cette matrice de Roe doit

- être diagonalisable
- vérifier

$$F_m(U_i) - F_m(U_j) = A_{Roe}^m(U_i, U_j)(U_i - U_j)$$

- vérifier  $A_{Roe}^m(U, U) = \nabla F_m(U)$  ce qui à une dimension d'espace se ramène à

$$A_{Roe}^m(U, U) = \begin{pmatrix} 0 & 0 & 1 & 0 \\ -u_m c_v & u_m & c_v & 0 \\ \chi + (\frac{1}{2}\kappa - 1)u_m^2 & \xi & (2 - \kappa)u_m & \kappa \\ (\chi + \frac{1}{2}\kappa u_m^2 - H)u_m & \xi u_m & H - \kappa u_m^2 & (\kappa + 1)u_m \end{pmatrix}$$

avec  $\chi, \xi$  et  $\kappa$  définis par les relations cités dans [6].

Dans le cas général de deux états  $U_i \neq U_j$ , la matrice de Roe  $A_{Roe}^m(U_i, U_j)$  s'écrit

$$A_{Roe}^m(U_i, U_j) = \begin{pmatrix} 0 & 0 & 1 & 0 \\ -\tilde{u}\tilde{c} & \tilde{u} & \tilde{c} & 0 \\ \tilde{\chi} + (\frac{1}{2}\tilde{\kappa} - 1)\tilde{u}^2 & \tilde{\xi} & (2 - \tilde{\kappa})\tilde{u} & \tilde{\kappa} \\ (\tilde{\chi} + \frac{1}{2}\tilde{\kappa}\tilde{u}^2 - \tilde{H})\tilde{u} & \tilde{\xi}\tilde{u} & \tilde{H} - \tilde{\kappa}\tilde{u}^2 & (\tilde{\kappa} + 1)\tilde{u} \end{pmatrix} \quad (7.11)$$

avec

$$\begin{aligned}\tilde{u} &= \frac{\sqrt{\rho_{mi}}u_{mi} + \sqrt{\rho_{mj}}u_{mj}}{\sqrt{\rho_{mi}} + \sqrt{\rho_{mj}}} \\ \tilde{H} &= \frac{\sqrt{\rho_{mi}}H_{mi} + \sqrt{\rho_{mj}}H_{mj}}{\sqrt{\rho_{mi}} + \sqrt{\rho_{mj}}} \\ \tilde{c} &= \frac{\sqrt{\rho_{mi}}c_{vi} + \sqrt{\rho_{mj}}c_{vj}}{\sqrt{\rho_{mi}} + \sqrt{\rho_{mj}}}.\end{aligned}$$

Le calcul exact de  $\tilde{\chi}$ ,  $\tilde{\xi}$  et  $\tilde{\kappa}$  est beaucoup plus complexe. Nous allons définir les paramètres suivants

$$\begin{aligned}\tilde{\rho}_m &= \sqrt{\rho_{mi}\rho_{mj}} \\ \tilde{T} &= \frac{\sqrt{\rho_{mi}}T_i + \sqrt{\rho_{mj}}T_j}{\sqrt{\rho_{mi}} + \sqrt{\rho_{mj}}} \\ \tilde{p} &= \tilde{\rho}\tilde{H} - \frac{1}{2}\tilde{\rho}\tilde{u}^2 - \tilde{\rho}(\tilde{c}e_v(\tilde{T}) + (1 - \tilde{c})e_l(\tilde{T})) \\ \tilde{\rho}_v &= \frac{\tilde{p} - p_{\infty v}}{(\gamma_v - 1)e_v(\tilde{T})} \\ \tilde{\rho}_l &= \frac{\tilde{p} - p_{\infty l}}{(\gamma_v - 1)e_l(\tilde{T})} \\ \tilde{\alpha}_v &= \frac{\tilde{c}\tilde{\rho}}{\tilde{\rho}_v}, \quad \tilde{\alpha}_l = 1 - \tilde{\alpha}_v\end{aligned}$$

et dans un premier temps nous contenter des approximations suivantes issues des équations ([6])

$$\begin{aligned}\tilde{\chi} &\approx \frac{\frac{\tilde{\alpha}_v\tilde{\rho}_vc_v^0 + \tilde{\alpha}_l\tilde{\rho}_lc_l^0}{\tilde{\rho}_l} - e_l(\tilde{T})\left(\frac{\tilde{\alpha}_vc_v^0}{e_v(\tilde{T})} + \frac{\tilde{\alpha}_lc_l^0}{e_l(\tilde{T})}\right)}{\frac{\tilde{\alpha}_v}{\tilde{p} - p_{\infty v}} + \frac{\tilde{\alpha}_l}{\tilde{p} - p_{\infty l}}} \\ \tilde{\xi} &\approx \frac{(\tilde{\alpha}_v\tilde{\rho}_vc_v^0 + \tilde{\alpha}_l\tilde{\rho}_lc_l^0)\frac{\tilde{\rho}_l - \tilde{\rho}_v}{\tilde{\rho}_v\tilde{\rho}_l} - (e_v(\tilde{T}) - e_l(\tilde{T}))\left(\frac{\tilde{\alpha}_vc_v^0}{e_v}\right)}{\frac{\tilde{\alpha}_v}{\tilde{p} - p_{\infty v}} + \frac{\tilde{\alpha}_l}{\tilde{p} - p_{\infty l}}} \\ \tilde{\kappa} &\approx \frac{\frac{\tilde{\alpha}_vc_v^0}{e_v(\tilde{T})} + \frac{\tilde{\alpha}_lc_l^0}{e_l(\tilde{T})}}{\frac{\tilde{\alpha}_v}{\tilde{p} - p_{\infty v}} + \frac{\tilde{\alpha}_l}{\tilde{p} - p_{\infty l}}}.\end{aligned}$$

### 7.3.3 Spectre de la matrice de Roe

En dimension d'espace 1, la matrice de Roe  $A_{Roe}^m(U_i, U_j)$  (équation 7.11) a quatre valeurs propres

$$\lambda_1 = \tilde{u} - \tilde{a}, \quad \lambda_2 = \tilde{u}, \quad \lambda_3 = \tilde{u}, \quad \lambda_4 = \tilde{u} + \tilde{a},$$

où  $\tilde{a}$  est la vitesse du son du mélange diphasique donnée par

$$\tilde{a}^2 = \tilde{\kappa}(\tilde{H} - \frac{1}{2}\tilde{u}^2) + \tilde{\chi} + \tilde{c}\tilde{\xi}.$$

Les quatre vecteurs propres à gauche et à droite associés sont

$$\begin{aligned} \vec{r}_1 &= \begin{pmatrix} 1 \\ \tilde{c} \\ \tilde{u} - \tilde{a} \\ \tilde{H} - \tilde{u}\tilde{a} \end{pmatrix}, \quad \vec{l}_1 = \frac{1}{2\tilde{a}^2} \begin{pmatrix} \tilde{\chi} + \frac{1}{2}\tilde{\kappa}\tilde{u}^2 + \tilde{a}\tilde{u} \\ \tilde{\xi} \\ -(\tilde{\kappa}\tilde{u} + \tilde{a}) \\ \tilde{\kappa} \end{pmatrix}, \\ \vec{r}_4 &= \begin{pmatrix} 1 \\ \tilde{c} \\ \tilde{u} + \tilde{a} \\ \tilde{H} + \tilde{u}\tilde{a} \end{pmatrix}, \quad \vec{l}_4 = \frac{1}{2\tilde{a}^2} \begin{pmatrix} \tilde{\chi} + \frac{1}{2}\tilde{\kappa}\tilde{u}^2 - \tilde{a}\tilde{u} \\ \tilde{\xi} \\ -(\tilde{\kappa}\tilde{u} - \tilde{a}) \\ \tilde{\kappa} \end{pmatrix}, \\ \vec{r}_2 &= \begin{pmatrix} 0 \\ 1 \\ 0 \\ -\frac{\tilde{\xi}}{\tilde{\kappa}} \end{pmatrix}, \quad \vec{l}_2 = \begin{pmatrix} -\tilde{c} \\ 1 \\ 0 \\ 0 \end{pmatrix}, \\ \vec{r}_3 &= \begin{pmatrix} 1 \\ \tilde{c} \\ \tilde{u} \\ \frac{1}{2}\tilde{u}^2 - \tilde{c}\tilde{\xi} - \tilde{\chi} \end{pmatrix}, \quad \vec{l}_3 = -\frac{1}{\tilde{a}^2} \begin{pmatrix} \tilde{\chi} + \frac{1}{2}\tilde{\kappa}\tilde{u}^2 - \tilde{a}^2 \\ \tilde{\xi} \\ -\tilde{\kappa}\tilde{u} \\ \tilde{\kappa} \end{pmatrix}. \end{aligned}$$

### 7.3.4 Calcul du décentrement

Pour le schéma de Roe ([21]), la matrice de décentrement  $D_{ij}$  (équation 3.9) devrait être égale à  $|A_{Roe}(U_i, U_j)|$ , la valeur absolue de la matrice de Roe associée au flux normal  $F_{\vec{n}_{ij}} = F_{\vec{n}_{ij}}^m + F_{\vec{n}_{ij}}^r$ . Cependant, celle-ci étant difficile à calculer, les concepteurs de Flica4 ont plutôt choisi

$$D_{ij} = |A_{Roe}^m(U_i, U_j)| = \sum_k |\lambda_k| \vec{l}_k \otimes \vec{r}_k, \quad (7.12)$$

où  $A_{Roe}^m(U_i, U_j)$  est la matrice de Roe de mélange (equation 7.11) associée au flux de mélange  $F_{\vec{n}_{ij}}^m$  dont le spectre  $(\lambda_k, \vec{l}_k, \vec{r}_k)$  est donné à la section 7.3.3. Ce choix d'approximation de la matrice de Roe est adapté aux écoulements cocourant où la vitesse relative  $\vec{u}_r$  est négligeable devant la vitesse moyenne  $\vec{u}_m$ . Le calcul du décentrement est effectué dans Flica4 en utilisant l'équation 7.12 et donc la décomposition spectrale de  $A_{Roe}^m(U_i, U_j)$  (annexe 7.3.3). Il est donc par construction limité au cas où les deux phases ont des vitesses



égales.

Dans notre maquette, nous utilisons la technique d'interpolation polynomiale qui ne nécessite pas le calcul des vecteurs propres de  $A_{Roe}^m(U_i, U_j)$ , mais uniquement ses valeurs propres. Nous espérons que ce choix nous permettra plus tard d'employer la matrice de Roe exacte  $D_{ij} = |A_{Roe}(U_i, U_j)|$  pour le calcul du décentrement et donc de mieux traiter les écarts de vitesse non négligeables, en particulier ceux des écoulements à contre-courant.

### 7.3.5 Correction de pression

Nous adoptons ici une approche multidimensionnelle car c'est dans ce contexte que les corrections de pression apportent une amélioration notable de la précision du schéma. Une fois la matrice de Roe approchée calculée le le flux originel de Roe à travers une face de normale  $\vec{n}_{ij}$  s'écrit

$$\tilde{F}(U_i, U_j, \vec{n}_{ij}) = \frac{F(U_i) + F(U_j)}{2} \vec{n}_{ij} - |A_{Roe}^m(U_i, U_j, \vec{n}_{ij})| \frac{U_i - U_j}{2}, \quad (7.13)$$

avec en notant  $\tilde{u}_n = \tilde{\vec{u}} \cdot \vec{n}_{ij}$  la vitesse normale à la face  $ij$

$$|A_{Roe}^m(U_i, U_j, \vec{n}_{ij})| = \left| \begin{pmatrix} 0 & 0 & {}^t\vec{n}_{ij} & 0 \\ -\tilde{u}_n \tilde{c} & \tilde{u}_n & \tilde{c} {}^t\vec{n}_{ij} & 0 \\ (\tilde{\chi} + \frac{1}{2}\tilde{\kappa}|\tilde{\vec{u}}|^2)\vec{n}_{ij} - \tilde{u}_n \tilde{\vec{u}} & \tilde{\xi} \vec{n}_{ij} & \tilde{\vec{u}} \otimes \vec{n}_{ij} + \tilde{u}_n \mathbb{I}_d - \tilde{\kappa} \vec{n}_{ij} \otimes \tilde{\vec{u}} & \tilde{\kappa} \vec{n}_{ij} \\ (\tilde{\chi} + \frac{1}{2}\tilde{\kappa}\tilde{u}^2 - \tilde{H})\tilde{u}_n & \tilde{\xi} \tilde{u}_n & \tilde{H} {}^t\vec{n}_{ij} - \tilde{\kappa} \tilde{u}_n {}^t\tilde{\vec{u}} & (\tilde{\kappa} + 1)\tilde{u}_n \end{pmatrix} \right|$$

Cependant, bien que stable, le schéma de Roe pose des problème de précision à faible nombre de Mach. En effet la pression équivalente à l'interface pour le flux (7.13) est

$$\tilde{p} = \frac{1 + \frac{\tilde{u}}{\tilde{a}}}{2} p_i + \frac{1 - \frac{\tilde{u}}{\tilde{a}}}{2} p_j - \frac{\tilde{a}}{2} (\rho_{mi} \vec{u}_{mi} - \rho_{mj} \vec{u}_{mj}) \cdot \vec{n}_{ij} + O(\|\rho_{mi} \vec{u}_{mi} - \rho_{mj} \vec{u}_{mj}\|^2).$$

Lorsque le nombre de Mach  $\frac{\tilde{u}}{\tilde{a}}$  est proche de zero, la contribution du décentrement  $\frac{\tilde{a}}{2} (\rho_{mi} \vec{u}_{mi} - \rho_{mj} \vec{u}_{mj}) \cdot \vec{n}_{ij}$  reste importante car  $|\tilde{\vec{u}}| \ll \tilde{a}$  et la diffusion numérique trop grande. Les concepteurs de Flica4 on donc introduit des “corrections de pression” pour en augmenter la précision. Le flux corrigé s'écrit

$$\tilde{F}(U_i, U_j, \vec{n}_{ij}) = \frac{F(U_i) + F(U_j)}{2} \vec{n}_{ij} - D_{ij} \frac{U_i - U_j}{2} + \frac{\tilde{a}(\rho_{mi} \vec{u}_{mi} - \rho_{mj} \vec{u}_{mj}) \cdot \vec{n}_{ij}}{2} \begin{pmatrix} 0 \\ 0 \\ \vec{n}_{ij} \\ 0 \end{pmatrix},$$

ce qui revient à rajouter une nouvelle contribution  $D_{ij}^p$  au décentrement  $D_{ij}$  avec

$$D_{ij}^p = \frac{\tilde{a}}{2} \begin{pmatrix} 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & \vec{n}_{ij} \otimes \vec{n}_{ij} & 0 \\ 0 & 0 & 0 & 0 \end{pmatrix}.$$

Pour que le schéma reste stable avec cette correction de pression, il faut se contenter d'une CFL inférieure à 0.5 au lieu de 1.

## 7.4 Capture des régimes stationnaires

Afin de montrer l'importance du décentrement des termes source raides nous considérons le système de lois de bilan stationnaire 1D suivant

$$\partial_x F(U) = S(U, x), \quad (7.14)$$

où  $S$  est un vecteur dont toutes les composantes sont positives, ce qui est le cas par exemple du modèle de dérive de CoreFlows. La Solution  $u$  est donc telle que  $F(U)$  a toutes ses composantes croissantes. Cela entraîne en général une monotonie de  $U$  et l'absence d'oscillations dans la solution.

Cette propriété de monotonie des composantes de  $U$  n'est pas toujours préservé au niveau discret et des oscillations parasites peuvent paraître, surtout lorsque le  $S$  admet des points de discontinuité.

Considérons le schéma classique décentré suivant :

$$F_{i+\frac{1}{2}} = \frac{F_i + F_{i+1}}{2} + \text{signe}(\bar{A}) \frac{F_i - F_{i+1}}{2}, \quad (7.15)$$

où  $\bar{A}$  est une approximation de la matrice jacobienne diagonalisable  $A$  dont le calcul n'est pas toujours exacte.

Dans notre cas,  $\bar{A}$  est la matrice de Roe associée au flux normale  $F_n$  apparaissant dans l'équation (3.1) tels que

$$F_n = F_n^m + F_n^r$$

où :

$F_n^m$  : correspond au flux de mélange [voir annexe 7.3]

$F_n^r$  : correspond au flux lié au mouvement relatif.

Cependant,  $\bar{A}$  étant difficile à calculer, les concepteurs de FLICA4 ont choisi

$$\bar{A} = A_{Roe}^m$$

ce choix est adapté aux écoulements où la vitesse relative  $\mathbf{u}_r$  est négligeable devant la vitesse moyenne  $\mathbf{u}_m$

### 7.4.1 Caractérisation de l'état stationnaire

L'état stationnaire associé au système 3.1 est défini par l'équation suivante :

$$\partial_x F(U^{stat}(x, t)) = S(U, x). \quad (7.16)$$

Pour que le problème stationnaire (7.16) admette une unique solution, on suppose que la matrice  $A$  soit inversible et en particulier  $\text{signe}(\bar{A}) * \text{signe}(\bar{A}) = \mathbb{I}_d$ .

Si le terme source  $S$  est positif *i.e* toutes ses composantes sont positives, l'équation (7.16) implique que toutes les composantes de  $F(U)^{stat}$  croissent, donc ne devraient pas osciller. Si une composante de  $S$  est égale à 0, alors la composante associée de  $F(U)$  devrait être constante.

On veut approcher les solutions de (7.16) numériquement tout en préservant la monotonie des flux au niveau discret. Et pour cela, l'état stationnaire discret doit satisfaire l'équation suivante :

$$\frac{F_{i+1} - F_i}{\Delta x} = S_{i+\frac{1}{2}}, \quad (7.17)$$

où  $S_{i+\frac{1}{2}}$  correspond à  $S(U, x, t)$  et garantit la préservation des régimes stationnaires si  $S_{i+\frac{1}{2}} \geq 0$

L'état stationnaire de (3.12) est caractérisé par :

$$\frac{F_{i+\frac{1}{2}} - F_{i-\frac{1}{2}}}{\Delta x} = S_i, \quad (7.18)$$

De (7.15), (7.17) et (7.18) on en déduit la relation entre  $S_i$  and  $S_{i+\frac{1}{2}}$ .

En remplaçant (7.15) dans (7.18) :

$$\frac{\mathbb{I}_d - \text{signe}(\bar{A})}{2} \frac{F_{i+1} - F_i}{\Delta x} + \frac{\mathbb{I}_d + \text{signe}(\bar{A})}{2} \frac{F_i - F_{i-1}}{\Delta x} = S_i.$$

Et selon (7.17)

$$\frac{\mathbb{I}_d - \text{signe}(\bar{A})}{2} S_{i+\frac{1}{2}} + \frac{\mathbb{I}_d + \text{signe}(\bar{A})}{2} S_{i-\frac{1}{2}} = S_i. \quad (7.19)$$

Or  $\text{signe}(\bar{A}) * \text{signe}(\bar{A}) = \mathbb{I}_d$ , en multipliant (7.19) par  $\frac{\mathbb{I}_d - \text{signe}(\bar{A})}{2}$  on obtient

$$\frac{\mathbb{I}_d - \text{signe}(\bar{A})}{2} S_{i+\frac{1}{2}} = \frac{\mathbb{I}_d - \text{signe}(\bar{A})}{2} S_i. \quad (7.20)$$

En multipliant (7.19) par  $\frac{\mathbb{I}_d + \text{signe}(\bar{A})}{2}$  et en remplaçant  $i$  par  $i + 1$  on obtient

$$\frac{\mathbb{I}_d + \text{signe}(\bar{A})}{2} S_{i+\frac{1}{2}} = \frac{\mathbb{I}_d + \text{signe}(\bar{A})}{2} S_{i+1}. \quad (7.21)$$

De (7.20) et (7.21) on déduit :

$$S_{i+\frac{1}{2}} = \frac{I_d + \text{signe}(\bar{A})}{2} S_{i+1} + \frac{I_d - \text{signe}(\bar{A})}{2} S_i. \quad (7.22)$$

### 7.4.2 Schéma décentré avec un terme source centré

Le schéma décentré classique avec le terme source centré s'écrit :

$$\frac{dU_i}{dt}(t) + \frac{F_{i+\frac{1}{2}} - F_{i-\frac{1}{2}}}{\Delta x} = S(U_i).$$

L'état stationnaire vérifie

$$\frac{F_{i+\frac{1}{2}} - F_{i-\frac{1}{2}}}{\Delta x} = S(U_i).$$

En appliquant la même méthode que le paragraphe précédant 7.4.1 on obtient :

$$\frac{F_{i+1} - F_i}{\Delta x} = \frac{S(U_i) + S(U_{i+1})}{2} + \text{signe}(\bar{A}) \frac{S(U_{i+1}) - S(U_i)}{2}. \quad (7.23)$$

(7.23) est une discrétisation cohérente de l'équation à l'état stationnaire mais ne conserve pas les propriétés de monotonie de la solution quand une ou plusieurs composantes de  $S$  sont négatives ou de signe fixe.

Si  $S$  est presque constant, nous ne sommes pas loin d'une solution monotone. Cependant pour des termes sources raides, le schéma peut fortement enfreindre la monotonie car on ne peut pas contrôler le signe de  $\frac{S(U_i) + S(U_{i+1})}{2} + \text{signe}(\bar{A}) \frac{S(U_{i+1}) - S(U_i)}{2}$ .

### 7.4.3 Schéma entièrement décentré avec un terme source décentré

Le système discret (7.24) découle naturellement de l'analyse du problème de Riemann avec terme source pour un système hyperbolique

$$\frac{dU_i}{dt}(t) + \frac{F_{i+\frac{1}{2}} - F_{i-\frac{1}{2}}}{\Delta x} = \frac{1}{2} (S_{i+\frac{1}{2}} + S_{i-\frac{1}{2}}), \quad (7.24)$$

où

$$\begin{aligned} S_{i-\frac{1}{2}} &= \frac{S(U_{i-1})+S(U_i)}{2} + \text{signe}(\bar{A}) \frac{S(U_{i-1})-S(U_i)}{2} \\ S_{i+\frac{1}{2}} &= \frac{S(U_i)+S(U_{i+1})}{2} + \text{signe}(\bar{A}) \frac{S(U_i)-S(U_{i+1})}{2} \\ S(U_i) &= S(U_i, x_i, t). \end{aligned}$$

Ici, le terme source et le schéma sont tous les deux décentrés. on obtient une discrétisation du terme de source, qui n'est pas triviale. Cette dernière implique le décentrement à l'interface des cellules voisines. Le premier terme de l'égalité (7.24) peut être différent de  $S(U_i)$  si le terme de source est raide.

La solution stationnaire discrète de (7.24) résout :

$$\frac{F_{i+\frac{1}{2}}^{stat} - F_{i-\frac{1}{2}}^{stat}}{\Delta x} = \frac{1}{2} (S_{i+\frac{1}{2}} + S_{i-\frac{1}{2}}). \quad (7.25)$$

En appliquant la même méthodologie que dans le paragraphe 7.4.1, on réécrit (7.25) en utilisant (7.15) comme suit :

$$\begin{aligned} &\frac{\mathbb{I}_d - \text{signe}(\bar{A})}{2} \frac{F_{i+1} - F_i}{\Delta x} + \frac{\mathbb{I}_d + \text{signe}(\bar{A})}{2} \frac{F_i - F_{i-1}}{\Delta x} \\ &= \\ &\frac{\mathbb{I}_d - \text{signe}(\bar{A})}{2} \frac{S(U_i) + S(U_{i+1})}{2} + \frac{\mathbb{I}_d + \text{signe}(\bar{A})}{2} \frac{S(U_{i-1}) + S(U_i)}{2}. \end{aligned}$$

En multipliant (7.26) par  $\frac{\mathbb{I}_d - \text{signe}(\bar{A})}{2}$  on obtient

$$\frac{\mathbb{I}_d - \text{signe}(\bar{A})}{2} \frac{F_{i+1} - F_i}{\Delta x} = \frac{\mathbb{I}_d - \text{signe}(\bar{A})}{2} \frac{S(U_i) + S(U_{i+1})}{2}. \quad (7.26)$$

En multipliant (7.26) par  $\frac{\mathbb{I}_d + \text{signe}(\bar{A})}{2}$  et en remplaçant  $i$  par  $i + 1$  on obtient

$$\frac{\mathbb{I}_d + \text{signe}(\bar{A})}{2} \frac{F_{i+1} - F_i}{\Delta x} = \frac{\mathbb{I}_d + \text{signe}(\bar{A})}{2} \frac{S(U_i) + S(U_{i+1})}{2}. \quad (7.27)$$

De (7.26) et (7.27) on en déduit

$$\frac{F_{i+1} - F_i}{\Delta x} = \frac{S(U_i) + S(U_{i+1})}{2}, \quad (7.28)$$

7.28 est une discrétisation uniforme de l'équation stationnaire (7.16) qui vérifie les mêmes propriétés de monotonie .

On conclut alors, que le décentrement du terme source implique la non oscillation de l'état stationnaire.



# Bibliographie

---

- [1] M. Ishii. Thermo-Fluid Dynamic Theory of Two-Phase Flow., volume 22 of Direction des études et recherches d'électricité de France. Eyrolles, Paris, 1975.
- [2] J. M. Masella, I. Faille and T. Gallouët, "On an Approximate Godunov Scheme", Intl. J. Computational Fluid Dynamics, 1999, Vol. 12, pp. 133-149.
- [3] D.A. Drew and S.L. Passman. Theory of Multicomponent Fluids., volume 135 of Applied Mathematical Sciences. Springer, New York, 1998
- [4] J.M Delhay, M. Giot, and M.L. Riethmuller. Thermohydraulics of Two-Phase Systems for Industrial Design and Nuclear Engineering, volume 5 of A Von Karman Institute Book. Mc Graw Hill Book Company, Hemisphere Publishing Corporation, 1981.
- [5] K. Halaoua. Quelques Solveurs pour les opérateurs de convection et leur application à la mécanique des fluides diphasiques. PhD thesis, ENS Cachan, 1998
- [6] T. Sid Abdelkader. Maquettage et test du code de thermohydraulique diphasique FLICA4, Rapport de PFE, SUP' Galilée -Université Paris 13, 2015
- [7] Angelo Murrone. Modèles bi-fluides à six et sept équations pour les écoulements diphasiques à faible nombre de Mach. Earth Sciences. Université de Provence - Aix-Marseille I, 2003. French
- [8] Projet "Ce sont Des Mathématiques Appliquées à la Thermohydraulique" : <http://cdmath.jimdo.com/>
- [9] Toolbox du projet CDMATH : <https://github.com/PROJECT-CDMATH/CDMATH>
- [10] Petsc 3.4.5 - Parallel storage and manipulation of large sparse matrices : <http://www.mcs.anl.gov/petsc/>
- [11] A. S. Shieh, V. H. Ransom, R. Krishnamurthy, Validation of numerical techniques in RELAP5/MOD3, RELAP5/MOD3 code manual volume 6, October 1994
- [12] <http://www.cas1.gov/COBRA-TF.shtml>
- [13] <http://www.ansys.com/Products/Simulation+Technology/Fluid+Dynamics/Fluid+Dynamics+Products/ANSYS+CFX>
- [14] J. J. Jeong, H. Y. Yoon, I. K. Park, H. K. Cho, J. Kim, A semi-implicit numerical scheme for transient two-phase flows, Nuclear Engineering and Design 236 (2008) 3403-3412

- 
- [15] I. Toumi, A. Bergeron, D. Gallo, E. Royer, D. Caruge, "FLICA-4 : a three-dimensional two-phase flow computer code with advanced numerical methods for nuclear applications," Nuclear Engineering and Design, Volume 200, Issues 1-2, August 2000, Pages 139-155.
- [16] Ph. Fillion, I. Toumi. Flica-4 V.1 Calcul des flux diffusifs dans le cas d'une géométrie non-structurée, Rapport CEA, DMT 95/335
- [17] S. Dellacherie, The Bichteler-Dellacherie Theorem, <https://almostsure.wordpress.com/2011/03/28/the-bichteler-dellacherie-theorem/>
- [18] D. Bestion, "The Physical Closure Laws in The CATHARE Code, Nuclear Engineering and Design," vol. 124, pp. 229-245, 1990.
- [19] N. Méchitoua, M. Boucker, J. Laviéville, J.-M. Hérard, S. Pigny, and G. Serre. An Unstructured Finite Volume Solver for Two-Phase Water/Vapour Flows Modelling Based on an Elliptic Oriented Fractional Step Method. In NURETH 10, International Meeting on Nuclear Reactor Thermal-Hydraulics, Seoul, South Korea, 2003.
- [20] Propriétés thermophysiques des systèmes fluides, <http://webbook.nist.gov/chemistry/fluid/>
- [21] Ph. Roe, Approximate Riemann solvers, parameter vectors, and difference schemes, J. Comput. Phys., 43, 357, 1981
- [22] "Upwind methods for hyperbolic conservation laws with source terms", A. Bermudez, E. Vazquez, Comp. Fluids, vol 23, issue 8, pp. 1049-1071, 1994
- [23] M. Ndjinga, T.-P.-K. Nguyen, C. Chalons, Numerical simulation of an incompressible two-fluid model, Springer Proc. Math. & Stat., Vol. 78, Finite Volumes for Complex Applications FVCA7, 2014
- [24] T.-H. Dao, M. Ndjinga, F. Magoules, Comparaison of Upwind and Centered Schemes for Low Mach Number Flows, Finite Volumes for Complex Applications VI - Problems & Perspectives, Springer Proceedings in Mathematics 4, 2011
- [25] R. J. LeVeque, Finite Volume Methods for Hyperbolic Problems, Cambridge University Press, 2002
- [26] The open-source integration platform for numerical simulation [www.salome-platform.org](http://www.salome-platform.org)
- [27] The open-source, parallel data analysis and visualization application [www.paraview.org](http://www.paraview.org)
- [28] The Visualization ToolKit [www.vtk.org](http://www.vtk.org)
- [29] The open-source integration platform for numerical simulation, : [www.eclipse.org](http://www.eclipse.org)
- [30] The open-source integration platform for numerical simulation, : [www.codeblocks.org](http://www.codeblocks.org)



- 
- [31] Parallel storage and manipulation of large sparse matrices [www.mcs.anl.gov/petsc](http://www.mcs.anl.gov/petsc)
  - [32] Simplified Wrapper and Interface Generator : <http://www.swig.org>
  - [33] HDF5 - Data model and file format of large volume : <http://www.hdfgroup.org/HDF5>
  - [34] DOXYGEN - Generate documentation from source code : <http://www.doxygen.org>
  - [35] CMAKE - Open-source build test and package software : <http://www.cmake.org>
  - [36] [https://fr.wikibooks.org/wiki/Programmation\\_Qt/Qt\\_Designer](https://fr.wikibooks.org/wiki/Programmation_Qt/Qt_Designer)



# Table des figures

---

1.1	Les dix centres du CEA en France . . . . .	2
1.2	Vue du ciel du centre Cea Saclay . . . . .	3
1.3	Organigramme du CEA . . . . .	4
1.4	Schéma de principe d'une centrale à eau pressurisée . . . . .	5
1.5	Schéma d'une cuve de REP 900 . . . . .	6
3.1	Comparaison de schémas numériques : Pression stationnaire $U_{entree} = 1m/s$	21
3.2	Calcul des gradients pour les flux visqueux . . . . .	23
4.1	Hierarchie des classes . . . . .	34
4.2	Environment de Développement de l'interface graphique de COREFLOWS .	35
4.3	Mécanisme de génération de SWIG . . . . .	36
4.4	Exemple d'utilisation de COREFLOWS sur <i>Python</i> . . . . .	37
4.5	Page d'accueil de COREFLOWS . . . . .	38
4.6	Interface de COREFLOWS 1/4 . . . . .	39
4.7	Interface de COREFLOWS 2/4 . . . . .	40
4.8	Interface de COREFLOWS 3/4 . . . . .	41
4.9	Interface de COREFLOWS 4/4 . . . . .	42
5.1	Cavité entraînée par son couvercle : Maillage structuré 2500 mailles . . . .	50
5.2	Cavité entraînée par son couvercle : Maillage non-structuré $\simeq 2600$ mailles	51
5.3	Cavité entraînée par son couvercle : Schéma Upwind - Maillage structuré .	51
5.4	Cavité entraînée par son couvercle : Schéma Upwind - Maillage non-structuré	52
5.5	Cavité entraînée par son couvercle : Schéma LowMach Explicite - Maillage structuré . . . . .	52
5.6	Cavité entraînée par son couvercle : Schéma LowMach Implicite - Maillage structuré . . . . .	53
5.7	Cavité entraînée par son couvercle : Schéma Staggered - Maillage structuré	54
5.8	Cavité entraînée par son couvercle : Schéma Staggered- Maillage non-structuré . . . . .	54
5.9	Mélange de deux lames d'eau de températures $T_1 = 563K$ et $T_2 = 623K$ à $t = 0s$ . . . . .	55
5.10	Mélange de deux lames d'eau de températures différentes, Schéma : LowMach	56
5.11	Mélange de deux lames d'eau de températures différentes, Schéma : Staggered	57
5.12	Mélange de deux lames d'eau de températures différentes, Schéma : Upwind	57

---

5.13	Configuration initiale du calcul . . . . .	58
5.14	Canal avec coloison chauffé partiellement : Vitesse $U_y$ , Schéma LowMach .	59
5.15	Canal avec coloison chauffé partiellement : Vitesse $U_y$ , Schéma LowMach- équilibre . . . . .	59
5.16	Canal avec coloison chauffé partiellement : Vitesse $U_y$ , Schéma Staggered .	60
5.17	Canal avec coloison chauffé partiellement : Vitesse $U_y$ , Schéma Staggered- équilibre . . . . .	60
5.18	Canal avec coloison chauffé partiellement : Vitesse $U_y$ , Schéma Upwind . .	61
5.19	Canal avec coloison chauffé partiellement : Vitesse $U_y$ , Schéma Upwind- équilibre . . . . .	61
5.20	Configuration initiale du canal . . . . .	62
5.21	Canal avec coloison chauffé partiellement : Vitesse $U_y$ , Schéma Upwind- équilibre . . . . .	62
5.22	Canal avec coloison chauffé partiellement : Vitesse $U_y$ , Schéma LowMach .	63
5.23	Canal avec coloison chauffé partiellement : Vitesse $U_y$ , Schéma Upwind- équilibre . . . . .	64
5.24	Canal avec coloison chauffé partiellement : Température, lignes de courant de la vitesse $U_y$ – Schéma Upwind-équilibre . . . . .	65
5.25	Canal avec coloison chauffé partiellement : Concentration, lignes de courant de la vitesse $U_y$ – Schéma Upwind-équilibre . . . . .	65
5.26	Canal avec coloison chauffé partiellement : Vitesse $U_y$ , Schéma Upwind- équilibre . . . . .	66
5.27	Canal avec coloison incliné chauffé partiellement : Concentration, lignes de courant de la vitesse $U_y$ – Schéma Upwind-équilibre . . . . .	67

# Table des matières

---

Résumé	i
Remerciements	iii
Nomenclature	v
<b>1 Introduction générale</b>	<b>1</b>
1.1 Contexte du stage . . . . .	1
1.1.1 Le CEA . . . . .	1
1.1.2 Le centre de Saclay . . . . .	2
1.1.3 Laboratoire de modélisation à l'échelle composante . . . . .	4
1.2 Problématique du stage . . . . .	5
<b>2 Le modèle mathématique</b>	<b>9</b>
2.1 Équations de bilan Euleriennes monophasiques . . . . .	10
2.2 Équations de bilan moyennées . . . . .	10
2.3 Obtention des équations du modèle de dérive . . . . .	12
2.3.1 Le modèle de dérive de <b>FLICA 4</b> . . . . .	12
2.3.2 Le modèle de dérive dans CoreFlows . . . . .	15
<b>3 Méthode numérique</b>	<b>19</b>
3.1 Formulation volumes finis de <b>FLICA 4</b> . . . . .	20
3.1.1 Flux convectifs . . . . .	20
3.1.2 Flux visqueux . . . . .	22
3.1.3 Discrétisation des termes sources . . . . .	23
3.2 Formulation volumes finis de CoreFlows . . . . .	23
3.2.1 Flux convectifs . . . . .	23
3.2.2 Flux visqueux $F_V$ . . . . .	25
3.2.3 Traitement des termes source . . . . .	26
<b>4 Application : CoreFlows</b>	<b>27</b>
4.1 Présentation de COREFLOWS . . . . .	27
4.2 Modèles physiques . . . . .	28
4.2.1 Problèmes linaires scalaires . . . . .	28
4.2.2 Équation de Navier-Stokes . . . . .	29

---

4.2.3	Modèles diphasiques . . . . .	29
4.3	Méthodes numériques . . . . .	32
4.4	Développement . . . . .	33
4.4.1	Architecture logicielle . . . . .	33
4.4.2	Génie Logiciel . . . . .	34
4.4.3	Documentation : Doxygen . . . . .	37
4.5	Exemples d'utilisation de COREFlows . . . . .	39
4.5.1	Interface graphique . . . . .	39
4.5.2	Scripts <i>Python</i> . . . . .	43
<b>5</b>	<b>Résultats numériques</b>	<b>49</b>
5.1	La cavité entraînée . . . . .	49
5.2	Écoulement piloté par la conduction . . . . .	55
5.3	Recirculation entre canaux parallèles . . . . .	58
5.3.1	Cas monophasique . . . . .	58
5.3.2	Cas diphasique . . . . .	63
<b>6</b>	<b>Conclusions</b>	<b>69</b>
<b>7</b>	<b>Annexe</b>	<b>71</b>
7.1	Discretisation en temps . . . . .	71
7.1.1	Schéma explicite . . . . .	71
7.1.2	Schéma implicite . . . . .	71
7.2	Conditions aux limites . . . . .	73
7.3	Annexe : linéarisation de Roe . . . . .	74
7.3.1	Définition de la matrice de ROE . . . . .	74
7.3.2	La matrice de Roe dans Flica4 . . . . .	75
7.3.3	Spectre de la matrice de Roe . . . . .	77
7.3.4	Calcul du décentrement . . . . .	78
7.3.5	Correction de pression . . . . .	79
7.4	Capture des régimes stationnaires . . . . .	80
7.4.1	Caractérisation de l'état stationnaire . . . . .	81
7.4.2	Schéma décentré avec un terme source centré . . . . .	82
7.4.3	Schéma entièrement décentré avec un terme source décentré . . . . .	82
	<b>Bibliographie</b>	<b>85</b>
	<b>Table des figures</b>	<b>90</b>