Adaptation dynamique de maillage pour les écoulements diphasiques en conduites pétrolières

Quang Long Nguyen

UNIVERSITÉ PIERRE ET MARIE CURIE
École Doctorale des Sciences Mathématiques de Paris Centre

Adaptation dynamique de maillage pour les écoulements diphasiques en conduites pétrolières

Application à la simulation des phénomènes de terrain slugging et de severe slugging

THÈSE

pour obtenir le grade de
Docteur en Mathématiques Appliquées

réalisée à
INSTITUT FRANÇAIS DU PÉTROLE

et soutenue par

Quang Long Nguyen

le 12/06/2009
devant le jury composé de

Thierry Gallouët	Université de Provence	Président
Philippe Helluy	Université de Strasbourg	Rapporteur
Kai Schneider	Université de Provence	Rapporteur
Yvon Maday	Université Pierre et Marie Curie	Directeur de thèse
Frédéric Coquel	Université Pierre et Marie Curie	Examinateur
Marie Postel	Université Pierre et Marie Curie	Examinatrice
Quang Huy Tran	Institut Français du Pétrole	Examinateur

Tout d'abord je tiens à remercier Yvon Maday, mon directeur de thèse, qui m'a acceuilli au sein du laboratoire Jaques-Louis Lions.

Je tiens à remercier aussi mes co-directeurs, Frédéric Coquel et Marie Postel, pour leur encadrement pendant ces trois années. Leur compétence technique et leurs aides sont indispensables pour la réussite de mon doctorat.

Je remercie également Roland Masson pour son acceuil chaleureux auprès du département Mathématiques Appliquées de l'IFP.

Mes remerciements s'adressent particulièrement aussi à Quang-Huy Tran, mon responsable à l'IFP. Je lui remercie pour le temps qu'il a consacré pour l'avancement de ma thèse tant sur le développement technique que sur la rédaction de ce rapport.

J'aimerais remercier également les rapporteurs de me donner la possibilité de présenter mes travaux de doctorat.

Table des matières

Liste des figures

Nomenclature

Indice des phase

l	Liquide
g	Gaz

Variables physiques

t	Temps (s)
ρ	Masse volumique du mélange (kg/m^3)
τ	Volume spécifique (m^3/kg)
X	Fraction massique de liquide
Y	Fraction massique de gaz
v	Vitesse massique du mélange (m/s)
v_g	Vitesse réelle du gaz (m/s)
v_l	Vitesse réelle du liquide (m/s)
u_s	Vitesse superficielle du mélange (m/s)
u_g	Vitesse superficielle du gaz (m/s)
u_l	Vitesse superficielle du liquide (m/s)
C_f	Coefficient de frottement
R_g	Fraction surfacique de gaz
R_l	Fraction surfacique de liquide
q_g	Débit de gaz
q_l	Débit de liquide

Caractéristique de la conduite

x	Abscisse le long de la conduite (m) en coordonnée eulérienne
L	Longueur (m) en coordonnée eulérienne
X	Abscisse le long de la conduite (m) en coordonnée lagrangienne
$\mathcal{L}$	Longueur (m) en coordonnée lagrangienne
χ	Coordonnée du référentiel mobile (ALE)
D	Diamètre de la conduite (m)
$\mathcal{T}$	Angle d'inclinaison (degré)

Variables thermodynamiques

p	Pression (Pa ou bar)
ρ_g	Masse volumique gu gaz (kg/m^3)
ρ_l	Masse volumique du liquide (kg/m^3)
a_g	Vitesse du son dans le gaz (m/s)
a_l	Vitesse du son dans le liquide (m/s)
p^0	Pression atmosphérique (Pa ou bar)
ρ_l^0	Masse volumique gu liquide (kg/m^3) à p^0

Variables liées au modèle

S	Source dans le système équilibre
ϕ	Glissement ou l'écart de vitesses entre les phases (m/s)

Variables liées à la discrétisation

N	Nombre de mailles
i	Indice de maille
$i+1/2$	Indice de l'arête entre deux mailles i et $i+1$
L	Gauche (état gauche du problème de Riemann)
R	Droite (état droite du problème de Riemann)
n	Indice d'itération en temps
Δt	Pas de temps
Δx	Pas d'espace

Variables liées à la relaxation

P	Pression apparente
σ	Impulsion apparente
Π	Pression apparente relaxée
Σ	Impulsion apparente relaxée
$(a,b)_{i+1/2}$	Coefficients de relaxation calculés localement sur l'arête $i+1/2$
η	Paramètre de relaxation

Variables vectorielles

$\mathbb{X}$	Variable quelconque
$\delta\mathbb{X}$	Incrément de la variable $\mathbb{X}$ entre deux instants t^n et t^{n+1}
$\mathbb{U}$	Variable conservative du système équilibre (3 composantes : $\rho, \rho Y, \rho v$)
$\mathbb{V}$	Variable primitive du système équilibre (3 composantes : τ, Y, v)
$\mathbb{W}$	Variable conservative du système de relaxation (5 composants : $\rho, \rho v, \rho\Pi, \rho Y, \rho\Sigma$)
$\mathcal{G}$	Flux physique du système de relaxation en eulérien
$\mathbb{G}$	Flux numérique du système de relaxation en eulérien
$\mathcal{F}$	Flux physique du système de relaxation en Lagrange-Projection
$\mathbb{F}$	Flux numérique du système de relaxation en Lagrange-Projection
$\mathbb{T}$	Source du système équilibre en variable conservative
$\mathbb{S}$	Source du système de relaxation en variable conservative

Variables matricielles

$\nabla\mathcal{G}$	Matrice Jacobienne du flux du système de relaxation en eulérien
$\mathcal{A}^{Roe}$	Matrice de Roe pour le schéma eulérien semi-implicite linéarisé
$\mathbb{I}$	Matrice d'identité

Variables et coefficients liés au schéma Lagrange-Projection

J	Taux de dilatation
$\widetilde{\mathbb{X}}$	Solution de la phase Lagrange en variable $\mathbb{X}$
$\widehat{\mathbb{X}}$	Solution de la phase Projection en variable $\mathbb{X}$
θ	Coefficent d'implicitation dans le cas semi-implicite
e_i	Coefficient local de propagation
E^i_j	Coefficient cumulé de propagation
$C^{\pm}$	Changement de variables

Variables et coefficients liés à la méthode Multirésolution

K	Nombre de niveaux de rafinement
N_0	Nombre de mailles sur la grille uniforme la plus grossière (niveau 0)
S^k	Grille uniforme de niveau k
M_i^k	Maille numéro i sur la grille uniforme de niveau k
Δx^k	Pas d'espace sur la grille uniforme de niveau k
$\mathbb{D}_i^{k-1}$	Détail de la maille $2i$ sur la grille uniforme de niveau k
$\mathbb{M}^K$	Représentation multi-échelle de la solution
$\mathbb{U}^k$	Solution au niveau k
ε_k	Paramètre de seuillage au niveau k
Γ_ε^n	Arbre de la Multirésolution au temps t^n
S_ε^n	Grille adaptative au temps t^n

Variables et coefficients liés à la méthode de Pas de Temps Local

$\Delta t_p^{n,k}$	$p+1$ ième micro pas de temps lié à une maille de niveau k
$t_p^{n,k}$	Temps intermédiaire après p micro pas de temps
$\mathbb{U}_{i,p}^{n,k}$	Solution sur la maille i au niveau k au temps intermédiaire $t_p^{n,k}$
$\mathcal{C}^l$	Ensemble des indices de mailles de niveau l pouvant évoluer avec un micro pas de temps $\Delta t_p^{n,l}$
$\bar{\mathcal{C}}^l$	Zone de transision liée à $\mathcal{C}^l$
$\mathbb{F}_{i+1/2,p}^{n,k}$	Flux numérique sur l'arête $i+1/2$ au micro pas de temps $\Delta t_p^{n,k}$ lié une maille de niveau k
$\mathcal{F}_p$	Ensemble des indices des flux à calculer au début du temps intermédiaire $t_p^{n,k}$

Abréviations

U	Uniforme
MR	Multirésolution
PTL/LTS	Pas de Temps Local
L-P	Lagrange-Projection

Introduction

Contexte de travail

Cette thèse s'inscrit dans le cadre des recherches exploratoires sur les Méthodes numériques avancées pour la modélisation. Elle s'intègre également dans une collaboration de plus longue date entre le Département Mathématiques Appliquées et le Laboratoire Jacques-Louis Lions, laquelle collaboration bénéficie du soutien du Ministère de la Recherche au titre d'*Équipe de Recherche Technologique*.

L'application visée est le transport polyphasique d'hydrocarbures à travers les conduites dans lesquelles plusieurs composants peuvent être présents : gaz, huile et eau. Dans le cadre de cette recherche de doctorat, nous nous intéressons uniquement aux écoulements diphasiques (ne contenant que deux phases : gaz et liquide) et isothermes (à température constante).

La simulation numérique de ces écoulements en transitoire constitue en effet l'un des enjeux majeurs dans l'industrie pétrolière. D'une part, les opérateurs tirent des résultats numériques des informations intéressantes pour mieux piloter le transport du mélange, de sorte à éviter certains phénomènes indésirables, notamment les créations de gros bouchons dans le *terrain slugging* ou le *severe slugging*. D'autre part, ces informations permettent aux constructeurs de mieux dimensionner les équipements pour économiser les matériels.

Dans ce contexte de travail, il est essentiel de pouvoir obtenir une assez grande précision sur la capture des ondes de transport, tout en garantissant de la robustesse au calcul. L'IFP développe, dans cette optique et depuis des années, un code de calcul appelé TACITE (Transient Analysis Code IFP Total Elf) permettant actuellement de simuler les écoulements polyphasiques unidimensionnels en régime transitoire [5, 79], avec des lois hydrodynamiques très complexes qui représentent au mieux la réalité. Le modèle mathématique utilisé dans TACITE est connu sous le nom de « Drift Flux Model » ou DFM. Ce modèle provient des lois de conservations (de masse et de quantité de mouvement) fermées par des lois de fermetures thermodynamiques (calcul de pression) et hydrodynamiques (écart de vitesses entre les phases). Le système est composé donc de trois équations aux dérivées partielles non-linéaires dues aux lois physiques. Ces dernières sont parfois fort complexes, ce qui implique un calcul de simulation important.

Motivations et objectifs de la thèse

Développé et commercialisé par la Direction Mécanique Appliquée de l'IFP, le code TACITE est un produit connu et reconnu dans le monde pétrolier pour sa performance : non seulement il fournit des résultats plus précis et plus proches des données expérimentales que les concurrents, mais aussi il permet de simuler des cas d'opération très raides, tels le suivi de bouchons dans le terrain slugging ou le severe slugging [4, 47], sans aucun artifice ad hoc supplémentaire. Il convient toutefois de relativiser cette performance, acquise au prix de beaucoup d'efforts, en mentionnant la dissipation numérique des résultats jugée parfois excessive. D'autre part, la nature du schéma utilisé, de type VFRoe [49, 51, 72],

requiert des calculs d'éléments propres de matrices en grand nombre. En conséquence, bien que les résultats soient très satisfaisants, l'évaluation de ces grandeurs physiques et la résolution du système entier provenant du schéma numérique consomment beaucoup de temps de calcul.

De nombreux travaux ont été réalisés dans le souci d'améliorer la précision, le coût et la robustesse de TACITE. D'abord, en vue de la robustesse, l'IFP a commencé par s'intéresser en collaboration avec F. Coquel (LJLL) à une classe de schémas dite de « relaxation » lors de la thèse de M. Baudin [5,6]. Ensuite, pour accélérer notablement les calculs, en liaison avec M. Postel (LJLL), N. Poussineau [38,81] a exploré la technique de Multirésolution (MR) qui permet de travailler avec un maillage adaptatif en espace, et ce sur un modèle simplifié (glissement nul et pente nulle) utilisant le schéma de relaxation. Inspirée des travaux de Cohen et *al.* [27] sur la décomposition multi-échelle pour les lois de conservation, la méthode « fully adaptive » a donné lieu à des résultats très prometteurs : il est possible de diviser le temps de calcul par un facteur d'au moins 3 pour une même qualité de résultat. Par la suite, N. Andrianov [4] a étendu l'utilisation de la MR au cas d'un glissement quelconque et avec changement de pentes. Les résultats numériques montrent non seulement un gain significatif par rapport au même schéma sur une grille uniforme, mais aussi que la méthode MR est capable de traiter des problèmes géométriques complexes, notamment celui du severe slugging.

Pour aller encore plus loin dans la réduction du temps CPU, tout en maintenant au même niveau, voire améliorant, la précision ainsi que la robustesse, l'étape suivante consiste à mettre au point un algorithme de pas de temps adaptatif. On parle alors de Pas de Temps Local (PTL). L'idée est qu'en ayant un pas de temps adapté à chaque maille, quitte à « synchroniser » la solution au bout d'un certain temps, on peut s'affranchir de la rigidité d'une contrainte de stabilité qui, jusqu'à présent, doit être réglée sur la plus petite maille. En cela, nous sommes encouragés par le succès des premiers travaux universitaires sur le PTL [27, 34, 35, 75].

Comme pour les travaux précédents, la nouvelle méthode sera considérée comme « recevable » si elle passe avec succès les tests de validation référencés en standard, tout en offrant une nette accélération. Ces tests de validation contiennent une vaste gamme de cas d'opération courants, y compris les cas raides d'écoulements à bouchon de type slugging.

Les objectifs de la thèse se composent de quatre étapes naturelles suivantes :

- passer à une loi de glissement non nul et pousser progressivement vers une loi de glissement réaliste de type TACITE,
- passer à l'ordre 2 en espace et en temps, à l'intérieur comme aux limites du domaine,
- réduire le nombre d'appels aux modules de fermeture qui sont déterminants pour le coût, en utilisant la technique de Multirésolution,
- explorer les sous pas de temps locaux PTL.

Revue des travaux antérieurs et en cours

Schémas de relaxation en uniforme

À partir d'idées antérieures de Whitham [98] et Liu [69], Jin et Xin [61] ont proposé pour la première fois en 1995 un schéma de relaxation pour résoudre un système de lois de conservation non-linéaire. La méthode de relaxation utilisée permet de construire un schéma numérique possédant de nombreux avantages grâce au nouveau système quasi-linéaire. L'applicication de cette méthode aux divers systèmes se trouvent dans Chen and Liu [20], Chen et *al.* [21], Serre [86]. Pour une revue complète

de la relaxation, nous renvoyons le lecteur au travail de Bouchut [16] et aux articles de Coquel et Perthame [37], Jin et Slemrod [60], Liu et *al.* [70]. La méthode de relaxation a été aussi appliquée au modèle DFM non linéaire [48, 50] et à ceux avec glissement suivant la stratégie développée par M. Baudin [5, 6]. Cette méthode permet d'une part d'accélérer notablement le temps de calcul, et d'autre part de rendre le système d'EDP plus facile à résoudre.

Rappelons les grandes lignes du travail de Baudin. Dans un premier temps, il a étudié un schéma de relaxation en coordonnées eulériennes en explicite et sur un maillage uniforme. La motivation est de traiter efficacement les non-linéarités provenant des relations de fermetures algébriques complexes. Pour cela, on introduit deux nouvelles variables de relaxation pour rendre le système plus facile à résoudre, sous forme quasi-linéaire. Par ailleurs, en utilisant le développement de Chapman-Enskog au premier ordre, il est possible de déterminer deux coefficients de relaxation afin d'assurer la stabilité du système de relaxation, tout en assurant qu'il s'agit d'une bonne approximation du système initial. L'avantage majeur de cette transformation est que, par construction, le nouveau système est inconditionnellement hyperbolique et tous ses champs sont linéairement dégénérés. Cela permet d'avoir une méthode d'approximation peu coûteuse grâce à la résolution très simple du problème de Riemann associé. De plus, nous pouvons assurer, uniquement pour ce schéma explicite, la positivité de la densité et le principe du maximum sur les fractions massiques des phases.

Muni des résultats en explicite, on cherche ensuite à construire une stratégie d'intégration mixte en temps, dite semi-implicite puisque seul ce type de schéma parvient à remplir le cahier des charges industriel imposé. Par « semi-implicite », on entend un schéma qui est : (i) explicite par rapport aux ondes lentes (représentant le transport de matière qui nous intéresse et pour lequel nous avons besoin de précision) ; (ii) implicite par rapport aux ondes rapides (représentant les phénomènes acoustiques qui ne nous intéressent pas et pour lesquels nous n'avons pas besoin de précision). Nous pouvons ainsi accélérer considérablement le temps de calcul, tout en respectant les conditions CFL assurant la stabilité du système et sans pour autant perdre la précision sur les ondes de transport. Dans l'approche de Baudin, les fonctions flux numériques utilisées sont basées sur une linéarisation de type VFRoe. En conséquence, ce schéma nécessite d'inverser un système linéaire construit sur la linéarisée de Roe.

L'un des désavantages du schéma de relaxation semi-implicite est la taille du problème matriciel à résoudre. Par construction du système relaxé (composé de cinq équations), il s'agit d'une résolution d'un système linéaire par blocs de taille 5×5. Un autre point important sur lequel nous insistons ici est que la garantie de positivité de la densité partielle n'existe qu'en explicite et disparaît en implicite ou semi-implicite, ce qui peut s'avérer gênant sur des cas raides. Or, une stratégie d'intégration semi-implicite en temps est indispensable pour faire passer ces cas réalistes. Nous allons donc montrer dans cette thèse qu'avec une nouvelle formulation du système initial en se basant toujours sur la méthode de relaxation, on peut non seulement réduire la taille du système à résoudre mais aussi avoir la positivité désirée (en semi-implicite).

D'un maillage uniforme à un maillage adaptatif

Alors que les travaux de Baudin sont focalisés sur la robustesse en maillage uniforme, ceux de Poussineau [38, 81] et Andrianov [4], lancés peu de temps après, ont pour objectifs la rapidité en maillage adaptatif.

La méthode adaptative en espace, appelée aussi technique Multirésolution ou Adaptation dynamique de maillage, a été proposée initialement par M. J. Berger et *al.* [13, 14] dont l'application se repose sur la résolution d'un système d'équations aux dérivées partielles, de type hyperbolique. Elle a été ensuite développée dans Harten [54–56] et Müller [73] puis dans Cohen et *al.* [27] dans le contexte de schémas volumes finis pour les systèmes de lois conservation. Formalisée de manière rigoureuse en utilisant la

théorie des ondelettes [25,53,57], cette technique MR a été validée par analyse en terme de convergence et d'estimation d'erreur dans le cas des grilles cartésiennes [27] et dans d'autres applications [28,29]. Cette méthode a été aussi développé pour le cas bidimensionnel sur des maillages cartésiens pour des schémas différences finies [15] et des volumes finis nonstructurés [1]. En multidimensionnelles, d'autres applications ont été faites par Chiavassa et Donat [22–24] (en différences finies) et par Cohen et *al.* [26] (en volumes finis sur des maillages triangulaires).

La méthode Multirésolution que l'on utilise dans ce travail est basée sur celle développée par Cohen [27] suivant l'analyse d'Harten. Plus précisément, la méthode MR repose sur une analyse multi-échelle de la régularité locale de la solution. La solution est représentée sur une hiérarchie de grilles dyadiques de discrétisation sur lesquelles on peut sélectionner localement le niveau de raffinement grâce au processus de codage et décodage et à l'opération de seuillage. Ce dernier permet en effet de négliger certains détails où la solution est considérée comme régulière. La grille adaptative doit se déplacer lorsque la solution évolue afin de bien suivre les singularités de la solution. De plus, la nature hyperbolique du système implique que les vitesses de propagation de la grille soient finies ce qui permet de contrôler l'évolution de la grille adaptative.

Dans nos applications spécifiques où il y a plusieurs ondes (lentes et rapides), il est intéressant de coupler la méthode de relaxation avec une technique de maillage adaptatif. Cela permet de réduire le nombre d'appels aux lois de fermetures, souvent très coûteuses, afin de gagner en temps CPU grâce à la technique de la Multirésolution ci-dessus. Les premiers travaux ont été réalisés dans [38, 81] sur un modèle simple (glissement nul et conduite horizontale) puis dans [4] sur un modèle plus compliqué (glissement non nul et variations des pentes le long de la conduite). Les résultats sont très prometteurs : la technique MR est non seulement performante (gain en temps CPU important) mais aussi capable de traiter les problèmes de terrain slugging et severe slugging. On constate que la méthode MR devient intéressante au niveau du gain en temps CPU dès que la solution peut être représentée sur les mailles grossières dans l'hiérarchie de grilles dyadiques. De plus, la qualité des résultats dépend nettement du paramètre de seuillage. Le choix de ce dernier joue donc un rôle important pour avoir un bon compromis entre le gain CPU et la qualité des résultats.

Par construction, un maillage adaptatif ne dépend pas du type de schéma numérique car il se repose uniquement sur la régularité de la solution. En conséquence, le traitement portant sur la reconstruction de ce maillage peut être considéré comme une sur-couche indépendante du schéma numérique utilisé. Au niveau de l'implémentation informatique, le code correspondant est donc portable et facile à utiliser (voir annexe C).

La technique d'adaptation de maillage en espace présentée ici peut être améliorée pour obtenir un meilleur gain CPU. En effet, le pas de temps dépend de la taille de maille de chaque cellule et de la condition CFL standard. Ce pas de temps est donc pénalisé par la taille de la maille la plus petite (zone de forte variation de la solution), d'où l'idée de calculer un pas de temps adapté à la taille de maille.

Vers le Pas de Temps Local

L'adaptation dynamique de maillage ne se limite pas à la dimension spatiale mais peut s'étendre aussi à la dimension temporelle. En effet, il est possible d'accroître la performance de cette approche en la combinant avec une méthode de Pas de Temps Local (PTL). Les premiers travaux ont été réalisés par Osher et Sanders [78] à partir desquels d'autres travaux ont été développés [30, 40, 89]. Cette technique PTL a aussi développée par Müller et Stiriba [75] dans le cadre des lois de conservations scalaires unidimensionelles et par Lamby et *al.* [63] pour une extension aux équations de Saint-Venant en dimension deux.

La méthode de Pas de Temps Local que nous utilisons ici est basée sur la méthode proposée dans [75]. L'idée principale est que, sur un maillage adaptatif, le pas de temps n'est pas constant mais variable en fonction de la taille de maille locale. Suivant une condition CFL, un macro pas de temps est calculé pour la maille la plus grossière et plus la maille est fine plus le pas de temps est petit. Le rapport entre ce pas de temps est égal donc au rapport de la taille des mailles correspondantes. L'évolution en temps de la solution se fait donc à plusieurs étapes différentes : à un pas de temps intermédiaire donné, seules les mailles au même niveau de synchronisation ou dans des niveaux plus fins sont à mettre à jour. Cette mise à jour est aussi nécessaire pour les zones de « transition », zones contenant les mailles plus grossières qui sont adjacentes aux mailles de niveau courant. Ceci a pour but de synchroniser la solution aux frontières des interfaces de deux niveaux différents. A la fin de chaque macro pas de temps, les solutions discrétisées doivent se retrouver au même niveau et le processus recommence ainsi.

Dans cette approche, on remarque deux points importants : la synchronisation de la solution et le remaillage de la grille lorsque la solution évolue au cours du temps. Pour le premier point, il existe en effet certaines façons pour traiter les zones de transitions afin de bien calculer les flux aux interfaces (voir [75, 78]). Pour le deuxième point, il est démontré dans [75] que, pour que la méthode PTL soit efficace, la prédiction de la grille adaptative en temps et en espace doit être recalculée à chaque pas de temps intermédiaire, et pas uniquement à chaque macro pas de temps.

Dans le cadre de notre travail, la méthode de Pas de Temps Local est appliquée aux problème d'écoulements, dans un premier temps sans conditions aux limites [34, 36] puis avec conditions aux limites [35]. La formulation Lagrange-Projection du schéma utilisé ici oblige à avoir un traitement différent que celui classique pour la synchronisation et le remaillage car il s'agit d'une résolution couplée en deux étapes pour chaque pas de temps intermédiaire. En explicite, une maille dans les zones de transition est suffisante pour une méthode de flux à deux points, tandis qu'en semi-implicite, le nombre de mailles dans ces zones doit être au moins égale à deux. Dans cette application, les résultats numériques sont très encourageants en regard de la convergence de la méthode PTL et son gain CPU beaucoup plus intéressant que celui obtenu avec la méthode MR standard, pour une même qualité de résultats. En effet, une méthode adaptative en temps et en espace permet de réduire encore le nombre d'appels aux lois de fermeture très coûteuses par rapport à la méthode adaptative en espace.

Démarche méthodologique

Les techniques MR/PTL ne peuvent bien fonctionner que si nous disposons d'un bon schéma « de base » en maillage uniforme. Rappelons que le schéma de relaxation semi-implicite développé initialement par Baudin, indispensable nos applications, possède encore certains défauts, à savoir la rapidité, la garantie de la positivité de la densité partielle. Aussi, sommes-nous tentés de recourir à un nouveau schéma semi-implicite reposant sur la décomposition Lagrange-Projection (ou formulation ALE), laquelle est d'ailleurs très classique en mécanique des fluides [42, 52, 83, 91, 97]. L'idée fondamentale revient à traiter la phase Lagrange de manière partiellement implicite, et de laisser explicite la phase Projection (voir par exemple Godlewski et Raviart [52]). Le flux sera donc décomposé en deux parties : une partie acoustique (non linéaire) et une partie transport (linéaire). Par construction, cette stratégie permet de produire un schéma semi-implicite localement conservatif, indispensables pour l'utilisation de la technique PTL, avec de surcroît la possibilité de garantir la positivité de la densité ainsi que le principe du maximum sur les fractions massiques.

Cette nouvelle technique Lagrange-Projection a été testée avec succès, au cours du stage de L. Linise [67], pour un modèle simplifié. La nouveauté théorique que nous souhaitons mettre en avant est que, dans cette technique de semi-implicitation, il est possible d'établir une condition CFL calée uniquement sur les vitesses lentes et permettant de garantir la stabilité du schéma et la positivité des

Fig. 1. Travaux antérieurs et actuels dans le contexte de la thèse : $\phi = 0$ représente le glissement nul, et $\phi \neq 0$ représente le glissement non nul.

densités. À partir de cette condition, on peut déduire un pas de temps convenable, lequel tient compte non seulement des ondes à l'intérieur du domaine mais aussi des variations temporelles des conditions aux limites imposées. À notre connaissance, ce type d'inégalité n'a jamais été mentionné dans la littérature sur Lagrange-Projection. De plus, le schéma L-P est plus rapide que le schéma eulérien, grâce principalement à une taille plus petite du système linéaire à résoudre (2×2 au lieu de 5×5). Quant à la qualité des résultats, elle est tout à fait comparable. Finalement, la méthode Lagrange-Projection que nous avons implémentée est capable de traiter une loi de glissement non triviale, réaliste, de type TACITE, à l'ordre 2 en temps et en espace et ce sur des cas-tests avec une inclinaison non nulle, voir un changement d'inclinaison.

Une fois le schéma de base maîtrisé, le passage à la MR/PTL pour réduire le temps de calcul sera transparent et quasiment automatique, dans la mesure où ces algorithmes sont conçus indépendamment du schéma de base, tant que celui-ci possède une notion de flux local. Nous nous proposons de partir de la méthode *fully adaptive* développée par Müller et Stiriba [75] pour des schémas de volumes finis explicites, puis de chercher à la généraliser au cas d'un schéma semi-implicite.

Plan de ce rapport

Le plan du rapport est le suivant. Dans le premier chapitre, nous considérons le modèle diphasique isotherme utilisé par TACITE, sur lequel est basé notre système. Ensuite, nous rappelons la méthode de relaxation ainsi que le schéma eulérien en explicite, implicite linéarisé et semi-implicite. Puis, les deux chapitres suivants sont consacrés à l'exposition des schémas Lagrange-Projection explicite et semi-implicite. Dans chaque chapitre, nous montrons des résultats numériques associés ainsi que la comparaison entre les deux schémas eulérien et L-P. La complexité des algorithmes de chacun de ces schémas se trouvent dans l'annexe B. Les deux chapitres suivants sont consacrés à décrire les méthodes Multirésolution et Pas de Temps Local. Dans le dernier chapitre, nous détaillons la loi hydrodynamique réaliste dite Synthétique afin de montrer la performance des méthodes MR et PTL. Le rapport sera complété par une conclusion dans laquelle nous évoquons les perspectives de la thèse.

Chapitre 1

Modèle physique

Le modèle retenu par les initiateurs du projet TACITE [80] pour simuler les écoulements diphasiques en conduites pétrolières porte le nom de DFM (Drift-Flux Model). L'origine de cette appellation se trouve dans les propriétés du modèle que nous allons rappeler.

1.1 Géométrie

La géométrie d'une conduite est caractérisée par la donnée de

- sa longueur L,
- son diamètre D,
- son inclinaison $\mathcal{T}$.

Fig. 1.1. Conduite élémentaire.

Sur chaque tronçon rectiligne, le diamètre et l'inclinaison sont constants. Une conduite est, par définition, la succession de plusieurs tronçons de longueur donnée. La variable x représente l'abscisse curviligne le long de la conduite.

1.2 Variables

Les équations sont classiquement écrites dans les variables suivantes (on indicera respectivement par l et g les variables relatives au liquide et au gaz) :

- les masses volumiques ρ_l, ρ_g,
- les fractions surfaciques R_l, R_g,
- les vitesses réelles v_l, v_g,
- la pression p.

Ces variables dépendent de l'abscisse x le long de la conduite et du temps t. Il est par ailleurs commode d'introduire les grandeurs suivantes :

- la masse volumique totale $\rho = \rho_l R_l + \rho_g R_g$,
- la fraction massique du liquide $X = \rho_l R_l / \rho$,
- la fraction massique du gaz $Y = \rho_g R_g / \rho$,
- la vitesse superficielle $u_s = R_l v_l + R_g v_g$,
- la vitesse massique du mélange $v = X v_l + Y v_g$,
- le débit de liquide $q_l = \rho_l R_l v_l$,
- le débit de gaz $q_g = \rho_g R_g v_g$.

Les égalités suivantes sont vérifiées :

$$X + Y = 1, \quad R_l + R_g = 1, \quad \frac{1}{\rho} = \frac{X}{\rho_l} + \frac{Y}{\rho_g}, \tag{1.1}$$

ainsi que les inégalités

$$0 \leq X \leq 1, \quad 0 \leq Y \leq 1, \quad 0 \leq R_l \leq 1, \quad 0 \leq R_g \leq 1. \tag{1.2}$$

1.3 Lois de conservation

Les équations aux dérivées partielles à la base du code TACITE sont obtenues en moyennant les lois de conservation classiques sur une section de la conduite.

1.3.1 Variables naturelles

Les équations de conservation mises en œuvre dans TACITE sont les suivantes :
- la masse du liquide

$$\partial_t(\rho_l R_l) + \partial_x(\rho_l R_l v_l) = 0, \tag{1.3}$$

- la masse du gaz

$$\partial_t(\rho_g R_g) + \partial_x(\rho_g R_g v_g) = 0, \tag{1.4}$$

- la quantité de mouvement totale

$$\partial_t(\rho_l R_l v_l + \rho_g R_g v_g) + \partial_x(\rho_l R_l v_l^2 + \rho_g R_g v_g^2 + p) = S. \tag{1.5}$$

La source S prend en compte les forces de pesanteur et les forces de frottement. Elle est de la forme

$$S = -\rho g \sin \mathcal{T} - \frac{2C_f}{D} \rho v |v|, \tag{1.6}$$

où C_f représente un coefficient de frottement. S ne dépend donc que de ρ et v.

La particularité de TACITE réside dans le fait qu'il n'y a qu'une seule équation de quantité de mouvement au lieu de deux (une pour chaque phase) comme dans les modèles bi-fluides standards [71]. Historiquement, ce choix [80] a été dicté par un souci de simplicité (forme conservative pour l'équation de quantité de mouvement total). Le prix à payer est la nécessité d'une relation supplémentaire, dite loi hydrodynamique, qui exprime le glissement

$$\phi = v_l - v_g, \tag{1.7}$$

en fonction des 3 variables conservatives de base (voir section suivante). Pour se souvenir de la présence d'une telle loi de fermeture, le modèle qui en résulte est appelé Drift-Flux Model (DFM) ou Modèle à Flux de Dérive.

1.3.2 Variables conservatives

Il est très utile de reformuler le système d'équations (1.3)–(1.5) sous une forme plus proche de la dynamique des gaz habituelle. Pour cela, on introduit le triplet $\mathbb{U} = (\rho, \rho Y, \rho v)^T$, en fonction duquel on exprime toutes les autres grandeurs, et ce via les relations thermodynamiques. Bien entendu, le glissement ϕ doit être considéré aussi comme une fonction de $(\rho, \rho Y, \rho v)$. Le système (1.3)–(1.5) se réécrit alors sous la forme

$$\begin{aligned} \partial_t(\rho) \quad &+ \partial_x(\rho v) &&= 0, \\ \partial_t(\rho Y) &+ \partial_x(\rho Y v - \sigma(\mathbb{U})) &&= 0, \\ \partial_t(\rho v) \ &+ \partial_x(\rho v^2 + P(\mathbb{U})) &&= S(\mathbb{U}), \end{aligned} \tag{1.8}$$

où

$$\begin{aligned} \sigma(\mathbb{U}) &= \rho Y(1-Y)\phi(\mathbb{U}) && \text{est l'impulsion apparente,} \\ P(\mathbb{U}) &= p(\mathbb{U}) + \rho Y(1-Y)\phi^2(\mathbb{U}) && \text{est la pression apparente.} \end{aligned} \tag{1.9}$$

Par abus de langage, nous appelons aussi P la pression et σ le glissement. La première équation de (1.8), résultant de la somme de (1.3) et (1.4), représente la conservation de la masse totale, tandis que la deuxième équation n'est autre que la réécriture de l'équation de conservation de la masse de gaz (1.4). Finalement, la troisième équation de (1.8) est le bilan de quantité de mouvement totale (1.5).

Notons que le système (1.8) peut s'écrire encore sous la forme abstraite :

$$\partial_t \mathbb{U} + \partial_x \mathcal{F}(\mathbb{U}) = \mathbb{T}(\mathbb{U}), \tag{1.10}$$

avec

$$\mathcal{F}(\mathbb{U}) = (\rho v, \rho Y v - \sigma(\mathbb{U}), \rho v^2 + P(\mathbb{U}))^T \quad \text{et} \quad \mathbb{T}(\mathbb{U}) = (0, 0, S)^T. \tag{1.11}$$

1.3.3 Variables primitives

Dans le cadre de ce travail, nous utilisons aussi la notion de variable « primitive » définie par

$$\mathbb{V} = (\tau, Y, v)^T, \tag{1.12}$$

où $\tau = 1/\rho$ est le volume spécifique. Cette variable intervient naturellement dans la réécriture du sytème en coordonnées lagrangiennes. Elle nous servira aussi pour la reconstruction des variables dans la montée en ordre du schéma numérique.

1.3.4 Cadre mathématique

L'espace des états admissibles associé au sytème de lois de conservation (1.8) est

$$\Omega = \left\{ \mathbb{U} \in \mathbb{R}^3 ; \rho > 0, Y \in [0,1], v \in \mathbb{R} \right\}. \tag{1.13}$$

On dit que le système (1.10) est *hyperbolique* si, pour tout $\mathbb{U} \in \Omega$, la matrice jacobienne $\nabla \mathcal{F}$ admet des valeurs propres réelles et possède une base de vecteurs propres.

La difficulté avec le système (1.8)–(1.9) est qu'on ne sait rien dire a priori sur son hyperbolicité éventuelle pour des lois de fermeture courantes. Néanmoins, il est observé sur les simulations que « numériquement » le système (1.8)–(1.9) est dans la plupart des cas hyperbolique. De surcroît, les valeurs propres

$$\lambda^- < \lambda_0 < \lambda^+, \tag{1.14}$$

vérifient pour les cas typiques

$$|\lambda_0| \ll |\lambda^-| \approx |\lambda^+|. \tag{1.15}$$

Autrement dit, on assiste à un phénomène de séparation des échelles de vitesse. Les vitesses caractéristiques λ^- et λ^+ se propagent grâce à la compressibilité du fluide et sont associées aux ondes acoustiques, dites *rapides*. Quant à la vitesse caractéristique λ_0 dont le signe est variable, elle se rapporte aux ondes de transport de matière (cinématique), dite *lentes*. Dans le contexte pétrolier, les ingénieurs s'intéressent surtout à la modélisation des ondes lentes car elles correspondent au déplacement de front des bouchons.

1.4 Lois de fermeture

1.4.1 Thermodynamique

La fermeture thermodynamique est donnée par une relation entre ρ_l et ρ_g

$$\frac{\rho(1-Y)}{\rho_l(p)} + \frac{\rho Y}{\rho_g(p)} = 1, \tag{1.16}$$

où $\rho_l = \rho_l(p)$ est la masse volumique du liquide et $\rho_g = \rho_g(p)$ est la masse volumique du gaz.

Dans le cadre de ce rapport, nous supposons que l'écoulement est isotherme (température constante) et que le gaz est parfait. Les équations d'état dans ce cas sont données par

$$\rho_g(p) = \frac{p}{a_g^2} \quad \text{et} \quad \rho_l(p) = \rho_l^0 + \frac{p - p^0}{a_l^2}, \tag{1.17}$$

où a_g est la vitesse du son dans le gaz, p^0 la pression atmosphérique, ρ_l^0 est la masse volumique du liquide à $p = p^0$ et a_l la vitesse du son dans le liquide. Dans la pratique, nous cherchons p en fonction de ρ et Y. La pression est alors donnée par l'inversion de l'égalité $\dfrac{1}{\rho} = \dfrac{X}{\rho_l} + \dfrac{Y}{\rho_g}$.

L'équation (1.17) reste valable pour un mélange d'un gaz parfait et d'un liquide compressible (a_l est fini). Si $a_l \to \infty$, on se trouve dans le cas d'un liquide incompressible c'est-à-dire tel que $\rho_l(p) = \rho_l^0$. La pression s'exprime alors simplement par

$$p(\rho, Y) = a_g^2 \frac{\rho Y}{1 - (1-Y)\frac{\rho}{\rho_l^0}}. \tag{1.18}$$

1.4.2 Hydrodynamique

La fermeture hydrodynamique, donnée par la loi de glissement (1.7), est souvent une propriété expérimentale (donc empirique) caractérisant la configuration d'écoulement considérée. Nous montrons la grande variété des régimes d'écoulement pour des conduites verticales et horizontales dans les Fig. 1.2 et 1.3.

Le glissement ϕ défini plus haut par (1.7) est de signe contraire à la convention habituelle $\phi = v_g - v_l$, mais cela ne change pas fondamentalement les idées. Dans le code TACITE, la fonction ϕ est d'une grande complexité. Elle n'a pas d'expression algébrique simple et peut même être discontinue. Afin de bien suivre les sections suivantes, exprimons tout d'abord v_l et v_g en fonction de ϕ et d'autres variables. Par définition de la vitesse absolue du mélange $v = Xv_l + Yv_g$, et de $\phi = v_l - v_g$, nous obtenons facilement

$$\begin{aligned} v_g &= v - X\phi, \\ v_l &= v + Y\phi. \end{aligned} \tag{1.19}$$

Fig. 1.2. Écoulement dans conduites verticales.

Fig. 1.3. Écoulement dans conduites horizontales.

Dans la pratique, il est utile aussi d'exprimer ces vitesses en fonction de la vitesse superficielle du mélange. Par définition de cette dernière $u_s = u_g + u_l$ où $u_g = R_g v_g$ et $u_l = R_l v_l$ sont respectivement la vitesse superficielle du gaz et du liquide, nous avons

$$u_s = v - (XR_g - YR_l)\phi, \tag{1.20}$$

d'où

$$\begin{aligned} v_g &= u_s - R_l\phi, \\ v_l &= \frac{u_s - u_g}{R_l} = u_s + R_g\phi. \end{aligned} \tag{1.21}$$

Nous appelons des lois de glissement *simples* celles dont le glissement ϕ ainsi que ces dérivées peuvent être exprimés explicitement. Examinons quelques lois de glissement *simples* bien référencées.

1.4.2.1 Glissement dispersé

Les écoulements dispersés sont caractérisés par la présence de phases sous forme d'émulsions. Typiquement dans ce cas on observe des petites bulles de gaz dans le liquide. Le glissement ϕ est donné par

$$\phi = -\frac{\delta}{R_l}, \tag{1.22}$$

où $\delta = 1.53 \sqrt[4]{g\varrho/\rho_l} \sin \mathcal{T}$ est la vitesse de dérive et ϱ la tension superficielle entre le gaz et le liquide.

***Remarque** 1.1.* Cette loi n'est valable que pour R_l proche de 1. En remplaçant (1.22) dans la première équation de (1.21), nous obtenons la vitesse absolue du gaz :

$$v_g = u_s + \delta. \tag{1.23}$$

***Remarque** 1.2.* Dans le cas d'une conduite horizontale ($\mathcal{T} = 0$), le glissement est nul $\phi = 0$ et $v_g = u_s = v$.

1.4.2.2 Glissement Zuber-Findlay CEA

Les lois de glissement de type Zuber-Findlay [95, 96] sont modélisées par la relation

$$u_g = \Psi_0(R_g, P, \mathcal{T})u_s + \Psi_1(R_g, P, \mathcal{T}), \tag{1.24}$$

où Ψ_0 et Ψ_1 sont deux fonctions de R_g, P et $\mathcal{T}$ (indépendantes de u_s).

Un cas particulier correspond à $\Psi_0 = R_g c_0$ et $\Psi_1 = R_g c_1$ où $c_0 = 1 + \mu > 1$ et $c_1 = \nu$, avec μ et ν deux constantes arbitraires. Dans ce cas, nous avons

$$u_g = R_g(c_0 u_s + c_1), \tag{1.25}$$

ce qui donne

$$v_g = c_0 u_s + c_1. \tag{1.26}$$

Les écoulements *intermittents* sont modélisés ainsi par la relation (1.26), appelée loi de Zuber-Findlay CEA.

Calculons maintenant le glissement ϕ pour cette loi. En réinjectant (1.19) dans (1.26), on obtient une loi de glissement non linéaire par rapport à $\mathbb{U}$ qui s'écrit

$$\phi(\mathbb{U}) = \frac{(1 - c_0)v - c_1}{1 - Y + c_0\kappa(\mathbb{U})} = -\frac{\mu v + \nu}{1 - Y + (1 + \mu)\kappa(\mathbb{U})}, \tag{1.27}$$

où

$$\kappa(\mathbb{U}) = (1 - Y)\left(\frac{\rho}{\rho_l(p(\mathbb{U})} - 1\right). \tag{1.28}$$

Dans les simulations numériques, on prend $\mu = 0.07$ et $\nu = 0.2162$.

***Remarque** 1.3.* Notons que cette loi de Zuber-Findlay CEA (1.27), obtenue par expérimentation correspond à un régime d'écoulement intermittent stationnaire et n'est valable que pour $R_g < 1/c_0$. Elle ne peut pas s'appliquer lorsque l'on approche des écoulements monophasiques gaz puisque dans ce cas $u_s = v_g$ donc $v_g = \frac{c_0}{1-c_0}$. C'est une valeur constante et n'a pas de sens physique. Pour remédier à ce problème lors des simulations faisant intervenir le monophasique gaz, l'IFP a mis au point une modification de la loi de Zuber-Findlay CEA.

1.4.2.3 Glissement Zuber-Findlay IFP

Dans la loi Zuber-Findlay IFP, on fait dépendre c_0 et c_1 de R_g et R_l. Plus précisément, afin d'assurer la propriété de consistance $v_g = u_s$ quand $R_g \to 1$ (voir remarque 1.5) on travaillera avec

$$\begin{aligned} c_0 &= 1 + \mu R_l \qquad \text{et} \qquad c_1 = \nu R_l, \\ \Psi_0 &= R_g(1 + \mu R_l) \qquad \text{et} \qquad \Psi_1 = \nu R_g R_l. \end{aligned} \tag{1.29}$$

Fig. 1.4. Loi Zuber-Findlay IFP - Vitesse superficielle du gaz u_g en fonction de R_g à u_S et $\mathcal{T}$ positifs fixés.

Les vitesses superficielle et absolue du gaz sont données ainsi par

$$\begin{aligned} u_g &= R_g(1+\mu R_l)u_s + \nu R_g R_l, \\ v_g &= (1+\mu R_l)u_s + \nu R_l. \end{aligned} \tag{1.30}$$

Après quelques calculs simples, nous obtenons le glissement ϕ pour ce modèle :

$$\phi(\mathbb{U}) = -\frac{\mu v + \nu}{1+\mu\kappa(\mathbb{U})}. \tag{1.31}$$

Dans les simulations numériques, on prend $\mu = 0.2\sin^2\mathcal{T}$ et $\nu = 0.35\sqrt{gD}\sin\mathcal{T}$. On remarque que la situation $\mathcal{T} = 0$ (conduite non inclinée) implique $v_g = u_s = v_l$ et le glissement est alors nul.

***Remarque** 1.4.* Pour cette loi de glissement, il est utile de connaître l'allure de u_g, définie par (1.30), en fonction de R_g. À u_s et $\mathcal{T}$ *positifs* fixés, u_g est un polynôme du second degré en R_g et est représenté sur la Fig. 1.4. Nous remarquons que u_g atteint son maximum pour $R_g^M = \dfrac{(1+\mu)u_s + \nu}{2(\mu u_s + \nu)}$ et il existe deux valeurs $R_g^\flat = \dfrac{u_s}{\mu u_s + \nu}$ et $R_g^\sharp = 1$ pour que u_g soit égale à u_s. Sur ce graphe, nous distinguons donc deux zones : une zone de co-courant ($u_g u_l > 0$) et une zone de contre-courant ($u_g u_l < 0$). La première correspond à $R_g \in [0, R_g^\flat]$ où $0 \leqslant u_g, u_l \leqslant u_s$ et la deuxième correspond à $R_g \in [R_g^\flat, 1]$ où $u_g \geqslant u_s$ et donc $u_l \leqslant 0$.

***Remarque** 1.5.* Les fonctions Ψ_0 et Ψ_1 vérifient les conditions de cohérence avec les états monophasiques $u_g(R_g = 0) = 0$ et $u_g(R_g = 1) = u_s$, lorsque R_g tend vers ses limites :

$$\begin{aligned} &\Psi_0(0, P, \mathcal{T}) = 0, \quad \text{et} \quad \Psi_1(0, P, \mathcal{T}) = 0, \\ &\Psi_0(1, P, \mathcal{T}) = 1, \quad \text{et} \quad \Psi_1(1, P, \mathcal{T}) = 0. \end{aligned} \tag{1.32}$$

***Remarque** 1.6.* Avec le modèle Zuber-Findlay (CEA ou IFP), à u_s fixé, le glissement ne dépend pas de R_g. On peut calculer ϕ et ses dérivées facilement grâce à son expression analytique assez simple.

1.4.2.4 Glissement Synthétique

Chacune des lois de glissement *simples* présentées ci-dessus n'est valide que sur un domaine limité des variables. Aussi, sommes-nous amenés à « recoller les morceaux » afin de pouvoir traiter les simulations « réalistes ». C'est l'objet de la loi de glissement synthétique, qui sera détaillée au chapitre 6.

1.5 Scénario par les conditions aux limites

Nous traiterons dans un premier temps des problèmes de Riemann, pour lesquels il n'y a pas de conditions aux limites. Pour les simulations des écoulements en conduites pétrolières, les unités de contrôle veulent travailler avec des conditions opératoires, à savoir les variations des conditions aux limites.

Plus spécifiquement, à l'entrée de la conduite ($x = 0$), on souhaite imposer les débits massiques du liquide et du gaz en fonction du temps, c'est-à-dire

$$(\rho_\alpha R_\alpha v_\alpha)(x = 0, t) = q_\alpha(t), \quad t \geq 0, \ \alpha = l, g. \tag{1.33}$$

De manière équivalente, cela revient à imposer le débit massique du gaz et le débit massique total

$$\begin{aligned} \rho v(x = 0, t) &= q_T(t), \\ (\rho Y v - \sigma)(x = 0, t) = \rho_g R_g v_g(x = 0, t) &= q_G(t), \end{aligned} \tag{1.34}$$

où $q_G(t) = q_g(t)$ et $q_T(t) = q_g(t) + q_l(t)$.
A la sortie ($x = L$), on souhaite imposer la pression dépendant du temps

$$p(x = L, t) = p(t), \quad t \geq 0. \tag{1.35}$$

Fig. 1.5. Conditions aux limites imposées à l'amont (débit de gaz $q_G(t)$ et débit total $q_T(t)$) et à l'aval (pression $p(t)$).

Il y a plusieurs façons de faire varier les conditions aux limites. On peut par exemple augmenter le débit du gaz à l'amont pendant un certain temps tout en maintenant constant le débit du liquide. On peut aussi annuler complètement ce dernier afin de rendre l'écoulement monophasique gaz. Quant à la pression, elle peut être maintenue constante ou doublée d'un temps à l'autre. Numériquement, ces conditions aux limites doivent être traitées de manière à satisfaire les débits et la pression imposés aux deux extrémités de la conduite. Nous verrons plus tard ces traitements au niveau numérique.

1.6 Donnée initiale

Afin d'« initialiser » les écoulements, nous devons avoir une donnée initiale $\mathbb{U}^0(x) = \mathbb{U}(x, t = 0)$. Cette dernière est l'état stationnaire, calculé à partir de la valeur des conditions aux limites à l'instant $t = 0$. Elle vérifie

$$\begin{aligned} \mathrm{d}_x(\rho^0 v^0) &= 0, \\ \mathrm{d}_x(\rho^0 Y^0 v^0 - \sigma^0) &= 0, \\ \mathrm{d}_x(\rho^0 (v^0)^2 + P^0) &= S^0, \end{aligned} \tag{1.36}$$

avec

$$\begin{aligned} \rho^0 v^0(x = 0) &= q_T(t = 0), \\ (\rho^0 Y^0 v^0 - \sigma^0)(x = 0) &= q_G(t = 0), \\ P^0(x = L) &= P(t = 0). \end{aligned} \tag{1.37}$$

Nous renvoyons le lecteur au travaux de Baudin [5] pour le calcul de cet état stationnaire.

Chapitre 2

Méthode de relaxation en coordonnées eulériennes

Après le modèle physique, nous poursuivons les rappels avec la méthode de relaxation [5, 6, 8, 61] appliquée au modèle DFM selon une approche directe en coordonnées eulériennes. Bien que cette partie relève entièrement de la thèse de Baudin [6], elle est indispensable pour bien comprendre l'apport du schéma Lagrange-Projection au chapitre suivant.

L'idée de la méthode de relaxation consiste à s'affranchir des deux véritables non-linéarités du problème (la pression P et le glissement σ) introduites par les lois de fermeture (thermodynamique et hydrodynamique) en approchant le système de départ par un système plus grand, hyperbolique avec tous les champs linéairement dégénérés. L'avantage de ce nouveau système est la simplicité de la structure d'ondes, ce qui permet d'avoir un solveur de Riemann peu coûteux et robuste. De plus, la stabilité du système de relaxation peut être assurée par le biais des conditions de type Whitham sur les coefficients de relaxation, lesquelles conditions découlent d'une analyse via le développement de Chapman-Enskog.

Au niveau discret, Baudin a étudié un schéma explicite et un schéma semi-implicite. Pour le schéma explicite, on parvient à renforcer les conditions sur les coefficients de relaxation afin de garantir la positivité (de la densité et des fractions). En ce qui concerne le schéma semi-implicite, il s'agit en réalité d'un implicite linéarisé dans lequel une heuristique, due à Gallouët [51], a été appliquée afin de rendre explicite le traitement des ondes lentes. Sa conception repose de façon cruciale sur la possibilité d'exprimer les flux de Godunov du système de relaxation sous la forme d'un flux de Roe.

Les points que nous venons de mentionner constituent dans l'ordre le plan de ce chapitre. Nous reviendrons également sur le calcul du pas de temps ainsi que les conditions aux limites.

2.1 Méthode de relaxation au niveau continu

2.1.1 Du système original au système de relaxation

Considérons notre modèle EDP original sans source ($S(\mathbb{U}) = 0$) (1.8), dit système à l'équilibre en coordonnées eulériennes

$$\begin{aligned} \partial_t(\rho) \quad &+ \partial_x(\rho v) &&= 0, \\ \partial_t(\rho Y) &+ \partial_x(\rho Y v - \sigma(\mathbb{U})) &&= 0, \\ \partial_t(\rho v) \ &+ \partial_x(\rho v^2 + P(\mathbb{U})) &&= 0, \end{aligned} \tag{2.1}$$

où la loi de pression totale P ainsi que la loi de glissement σ sont des fonctions de l'inconnue $\mathbb{U} = (\rho, \rho Y, \rho v)^T$ telles que décrites dans le chapitre 1.

De manière à bien mettre en évidence les véritables non-linéarités du système et à mieux appréhender l'approximation par relaxation que nous proposons pour (2.1), nous commençons par transformer le système (2.1) en coordonnées lagrangiennes (t, X)

$$\begin{aligned} \partial_t(\tau) &- \partial_X(v) &&= 0, \\ \partial_t(v) &+ \partial_X(P(\mathbb{V})) &&= 0, \\ \partial_t(Y) &- \partial_X(\sigma(\mathbb{V})) &&= 0, \end{aligned} \tag{2.2}$$

où $\mathbb{V} = (\tau, Y, v)^T$. Dans ce jeu de variables primitives, on rappelle que $\tau = 1/\rho$ est le volume spécifique et que la variable « spaciale » X est définie par $dX = \rho dx - \rho v dt$.

On constate que seules la loi de pression et la loi de glissement sont à l'origine des non-linéarités dans le système (2.2), d'où l'idée de remplacer la loi de pression $P(\mathbb{V})$ par une nouvelle inconnue notée Π et la loi de glissement $\sigma(\mathbb{V})$ par une nouvelle inconnue notée Σ. Ces deux inconnues sont chacune gouvernée par une équation aux dérivées partielles assujettie à un terme source de relaxation. Le nouveau système que l'on appelle dans la suite *système de relaxation* s'écrit

$$\begin{aligned} \partial_t(\tau) &- \partial_X(v) &&= 0, \\ \partial_t(v) &+ \partial_X(\Pi) &&= 0, \\ \partial_t(\Pi) &+ a^2\, \partial_X(v) &&= \eta(P(\mathbb{V}) - \Pi), \\ \partial_t(Y) &- \partial_X(\Sigma) &&= 0, \\ \partial_t(\Sigma) &- b^2\, \partial_X(Y) &&= \eta(\sigma(\mathbb{V}) - \Sigma), \end{aligned} \tag{2.3}$$

où $\eta > 0$ est le paramètre de relaxation et les coefficients de relaxation $a, b > 0$ sont deux constantes réelles à ajuster. Nous allons voir plus tard que a a la dimension de la vitesse du son et b a la dimension d'une vitesse de transport. En conséquence, on s'attend à ce que a soit beaucoup plus grand que b, à savoir

$$a \gg b > 0. \tag{2.4}$$

Observons que le système du premier ordre (2.3) n'est composé que d'équations aux dérivées partielles linéaires à coefficients variables. Les non-linéarités n'interviennent que dans les termes sources de relaxation. Pour $\eta = 0$, le système de relaxation (2.3) se découple en deux systèmes totalement indépendants en coordonnées lagrangiennes : un bloc acoustique (lié aux ondes rapides $-a$ et a) gouvernant les inconnues (τ, Y, Π) et un bloc cinématique (lié aux ondes lentes $-b$ et b) régissant les inconnues (Y, Σ). Pour $\eta > 0$ les deux systèmes sont couplés à travers le second membre de (2.3).

Dans un deuxième temps, on fait revenir le système (2.3) en coordonnées eulériennes. Le système obtenu

$$\begin{aligned} \partial_t(\rho) &+ \partial_x(\rho v) &&= 0, \\ \partial_t(\rho v) &+ \partial_x(\rho v^2 + \Pi) &&= 0, \\ \partial_t(\rho \Pi) &+ \partial_x(\rho \Pi v + a^2 v) &&= \eta\rho(P(\mathbb{U}) - \Pi), \\ \partial_t(\rho Y) &+ \partial_x(\rho Y v - \Sigma) &&= 0, \\ \partial_t(\rho \Sigma) &+ \partial_x(\rho \Sigma v - b^2 Y) &&= \eta\rho(\sigma(\mathbb{U}) - \Sigma), \end{aligned} \tag{2.5}$$

est déclaré l'« approximation par relaxation » du système de départ (2.1). C'est désormais avec (2.5) que nous travaillons en pratique. Nous notons la variable conservative $\mathbb{W}$ utilisée dans le système de relaxation

$$\mathbb{W} = (\rho, \rho v, \rho\Pi, \rho Y \rho\Sigma)^T. \tag{2.6}$$

Formellement, dans la limite d'un paramètre de relaxation η infini, Π coïncide avec la loi de pression P et Σ avec la loi de glissement σ. De sorte à éviter les instabilités dans la procédure pour les grandes valeurs de η, il est connu (voir par exemple Liu [69], Chen et Levermore [21]) qu'une condition de compatibilité dite condition sous-caractéristique de Whitham entre le système équilibre (2.1) et le système de relaxation (2.3) doit être vérifiée. Cette condition impose aux coefficients a, b d'être suffisamment grands (a doit être choisi plus grand que la vitesse du son). Nous allons maintenant détailler cette condition.

2.1.2 Condition sous-caractéristique de Whitham

On souhaite non seulement garantir la stabilité du système de relaxation (2.5) mais aussi s'assurer qu'il constitue une bonne approximation du système de départ (2.1).

Ainsi que nous l'avons affirmé, la théorie de l'approximation par relaxation impose une condition de compatibilité entre le système équilibre et le système de relaxation. Cette condition de compatibilité, appellée « condition de Whitham » ou aussi « condition sous-caractéristique », est issue d'une condition de dissipativité obtenue au terme d'une analyse de Chapman-Enskog. La démarche est la suivante : on cherche à décrire l'évolution du système de relaxation

$$\partial_t(\mathbb{W}_\eta) + \partial_x \mathcal{G}(\mathbb{W}_\eta) = \eta \mathbb{S}^{Relax}(\mathbb{W}_\eta), \tag{2.7}$$

au voisinage de la variété d'équilibre $\mathbb{W}_{eq}$ en postulant un développement asymptotique formel

$$\mathbb{W}_\eta = \mathbb{W}_{eq} + \frac{1}{\eta}\mathbb{W}_1 + O(\frac{1}{\eta^2}). \tag{2.8}$$

La caractérisation du premier ordre $\mathbb{W}_1$ conduit après élimination des variables de relaxation à une perturbation second ordre du système équilibre dont on exige qu'elle soit dissipative. La matrice de diffusion de cette perturbation dépend des coefficients a et b. Exiger la propriété de dissipativité implique dans notre cadre de travail de choisir ces coefficients sous les conditions

$$a > \max_{\mathbb{U}} \sqrt{-\partial_\tau P(\mathbb{U}) + [\partial_v P(\mathbb{U})]^2}, \tag{2.9a}$$

$$b > \max_{\mathbb{U}} |\partial_Y \sigma(\mathbb{U})|, \tag{2.9b}$$

pour tous états $\mathbb{U}$ considérés. Notons que la dérivée $\partial_\tau P$ (resp. $\partial_v P$) est calculée à v et Y (resp. τ et Y) fixés. Nous renvoyons le lecteur aux travaux de Baudin et *al.* [6] pour les détails de calcul. Rappelons que compte-tenu des valeurs des dérivées respectives de P et σ on a toujours

$$a \gg b > 0. \tag{2.10}$$

Ceci est cohérent avec le fait que a est le paramètre de relaxation associé aux ondes de pressions (ondes rapides) et b celui associé aux ondes de transport (ondes lentes).

Nous verrons par ailleurs qu'il faut définir a et b suffisamment grands de sorte à préserver certaines propriétés physiques, à savoir la positivité de la densité $\rho > 0$ et le principe du maximum sur la fraction massique, c'est-à-dire $Y \in [0, 1]$.

2.1.3 Propriétés mathématiques du système de relaxation

Étudions maintenant les propriétés mathématiques du système de relaxation (2.5) que nous réécrivons sous la forme condensée

$$\partial_t \mathbb{W} + \partial_x \mathcal{G}(\mathbb{W}) = \eta \mathbb{S}^{Relax}(\mathbb{W}), \quad t > 0, x \in \mathbb{R}, \tag{2.11}$$

où le terme source de relaxation est défini par

$$\mathbb{S}^{Relax}(\mathbb{W}) = (0, 0, \rho(P(\mathbb{U}) - \Pi), 0, \rho(\sigma(\mathbb{U}) - \Sigma))^T, \tag{2.12}$$

et le flux correspondant

$$\mathcal{G}(\mathbb{W}) = (\rho v, \rho v^2 + \Pi, \rho \Pi v + a^2 v, \rho Y v - \Sigma, \rho \Sigma v - b^2 Y)^T. \tag{2.13}$$

L'espace des états associé à (2.11) est défini par

$$\Omega = \{\mathbb{W} \in \mathbb{R}^5; \rho > 0, v \in \mathbb{R}, \Pi \in \mathbb{R}, Y \in [0,1], \Sigma \in \mathbb{R}\}. \tag{2.14}$$

La matrice jacobienne du système de relaxation (2.11)

$$\nabla_{\mathbb{W}} \mathcal{G}(\mathbb{W}) = \begin{bmatrix} 0 & 1 & 0 & 0 & 0 \\ -v^2 - \Pi\tau & 2v & \tau & 0 & 0 \\ -v\Pi + a^2 v\tau & \Pi + a^2\tau & v & 0 & 0 \\ -vY + \Sigma\tau & Y & 0 & v & -\tau \\ -v\Sigma + b^2 Y\tau & \Sigma & 0 & -b^2\tau & v \end{bmatrix} \tag{2.15}$$

est diagonalisable pour tout état $\mathbb{W}$ de Ω avec cinq valeurs propres réelles distinctes naturellement ordonnées (voir [6, 7]).

$$\lambda_1(\mathbb{W}) = v - a\tau, \quad \lambda_2(\mathbb{W}) = v - b\tau, \quad \lambda_3(\mathbb{W}) = v, \quad \lambda_4(\mathbb{W}) = v + b\tau, \quad \lambda_5(\mathbb{W}) = v + a\tau, \tag{2.16}$$

et cinq vecteurs propres à droite correspondants

$$\begin{gathered} \overrightarrow{r_1} = (1, v - a\tau, \Pi + a^2\tau, Y, \Sigma)^T, \quad \overrightarrow{r_2} = (0, 0, 0, 1, b)^T, \quad \overrightarrow{r_3} = (1, v, \Pi, Y, \Sigma)^T, \\ \overrightarrow{r_4} = (0, 0, 0, 1, -b)^T, \quad \overrightarrow{r_5} = (1, v + a\tau, \Pi + a^2\tau, Y, \Sigma)^T. \end{gathered} \tag{2.17}$$

Étant donnée la signification physique des coefficients a et b, les valeurs propres λ_1 et λ_5 correspondent aux ondes acoustiques et les valeurs propres $\lambda_2, \lambda_3, \lambda_4$ correspondent aux ondes cinématiques. Dans l'application que nous allons traiter, les vitesses acoustiques sont beaucoup plus grandes que les ondes cinématiques, les ondes acoustiques sont donc les ondes rapides, et les ondes cinématiques sont les ondes lentes.

On peut vérifier que les valeurs propres $\lambda_{i=1,\ldots,5}(\mathbb{W})$ sont associées à des champs linéairement dégénérés et chacune de ces i-ondes admet un invariant de Riemann fort $\Psi_i(\mathbb{W})$ vérifiant, en l'absence de terme source de relaxation, l'équation aux dérivées partielles suivante

$$\partial_t \Psi_i(\mathbb{W}) + \lambda_i(\mathbb{W}) \partial_x \Psi_i(\mathbb{W}) = 0, \tag{2.18}$$

pour toute solution régulière $\mathbb{W}$ de (2.11). Rappelons qu'un i-invariant de Riemann fort est un invariant de Riemann faible pour toutes les autres ondes, c'est-à-dire qu'il reste constant à travers toutes les j-ondes ($j = 1, \ldots, 5$) avec $j \neq i$. La liste des invariants forts est donnée comme suit

$$\Psi_1(\mathbb{W}) = \Pi - av, \quad \Psi_2(\mathbb{W}) = \Sigma + bY, \quad \Psi_3(\mathbb{W}) = \Pi + a\tau^2, \quad \Psi_4(\mathbb{W}) = \Sigma - bY. \quad \Psi_5(\mathbb{W}) = \Pi + av, \tag{2.19}$$

Fig. 2.1. La structure du problème de Riemann pour le système de relaxation (2.5). Les ω_i sont les vitesses de propagation des discontinuités de contact séparant 6 états intermédiaires $(\mathbb{W}_j)_{j=1,\dots,6}$ du problème de Riemann.

2.1.4 Problème de Riemann pour le système de relaxation

On souhaite résoudre le problème de Riemann associé au système de relaxation (2.5) sans terme source muni de la condition initiale

$$\mathbb{W}_0(x) = \begin{cases} \mathbb{W}_L & \text{si } x < 0, \\ \mathbb{W}_R & \text{si } x > 0, \end{cases} \tag{2.20}$$

avec $\mathbb{W}_L, \mathbb{W}_R$ deux états donnés dans Ω.

Avant de définir la solution du problème de Riemann associé à (2.20), nous introduisons des notations suivantes avec a et b strictement positifs :

$$\begin{aligned} \tau_L^* &= \tau_L + \frac{v_R - v_L}{2a} - \frac{\Pi_R - \Pi_L}{2a^2}, & \tau_R^* &= \tau_R + \frac{v_R - v_L}{2a} + \frac{\Pi_R - \Pi_L}{2a^2}, \\ v^* &= \frac{v_R + v_L}{2} - \frac{\Pi_R - \Pi_L}{2a}, & \Pi^* &= \frac{\Pi_R + \Pi_L}{2} - a\frac{v_R - v_L}{2}, \\ Y^* &= \frac{Y_L + Y_R}{2} + \frac{\Sigma_R - \Sigma_L}{2b}, & \Sigma^* &= \frac{\Sigma_L + \Sigma_R}{2} + b\frac{Y_R - Y_L}{2}. \end{aligned} \tag{2.21}$$

D'après la caractérisation spectrale du système homogène (2.5), il est attendu que la solution auto-semblable $\mathbb{W}(\frac{x}{t}, \mathbb{W}_L, \mathbb{W}_R)$ du problème de Riemann considéré soit constituée d'au plus 6 états $(\mathbb{W}_j)_{j=1,\dots,6}$ systématiquement séparés par des discontinuités de contact se propageant à la vitesse $(\omega_i)_{i=1,\dots,5}$. La structure de la solution du problème de Riemann associé (2.20) est présentée dans la Fig. 2.1 où $\mathbb{W}_1 = \mathbb{W}_L$, $\mathbb{W}_6 = \mathbb{W}_R$. L'existence d'une famille complète d'invariants de Riemann forts donnée en (2.19) permet la détermination aisée des quatre états intermédiaires, à savoir

$$\mathbb{W}_2 = \begin{bmatrix} \rho_L^* \\ \rho_L^*\Pi^* \\ \rho_L^* v^* \\ \rho_L^* Y_L \\ \rho_L^* \Sigma_L \end{bmatrix}, \quad \mathbb{W}_3 = \begin{bmatrix} \rho_L^* \\ \rho_L^*\Pi^* \\ \rho_L^* v^* \\ \rho_L^* Y^* \\ \rho_L^* \Sigma^* \end{bmatrix}, \quad \mathbb{W}_4 = \begin{bmatrix} \rho_R^* \\ \rho_R^*\Pi^* \\ \rho_R^* v^* \\ \rho_R^* Y^* \\ \rho_R^* \Sigma^* \end{bmatrix}, \quad \mathbb{W}_5 = \begin{bmatrix} \rho_R^* \\ \rho_R^*\Pi^* \\ \rho_R^* v^* \\ \rho_R^* Y_R \\ \rho_R^* \Sigma_R \end{bmatrix}. \tag{2.22}$$

Les vitesses $(\omega_i)_{i=1..5}$ sont définies par

$$\begin{aligned}
\omega_1 &= \lambda_1(\mathbb{W}_1) = \lambda_1(\mathbb{W}_2),\\
\omega_2 &= \lambda_2(\mathbb{W}_2) = \lambda_2(\mathbb{W}_3),\\
\omega_3 &= \lambda_3(\mathbb{W}_3) = \lambda_3(\mathbb{W}_4),\\
\omega_4 &= \lambda_4(\mathbb{W}_4) = \lambda_4(\mathbb{W}_5),\\
\omega_5 &= \lambda_5(\mathbb{W}_5) = \lambda_5(\mathbb{W}_6),
\end{aligned} \tag{2.23}$$

où les $(\lambda_i)_{i=1..5}$ trouvent les définitions dans (2.16), d'où

$$\omega_1 = v_L - a\tau_L, \quad \omega_2 = v^* - b\tau_L^*, \quad \omega_3 = v^*, \quad \omega_4 = v^* + b\tau_R^*, \quad \omega_5 = v_R + a\tau_R. \tag{2.24}$$

On remarque que grâce à la propriété de dégénérescence linéaire des λ_i, on peut vérifier facilement $v_L - a\tau_L = v^* - a\tau_L^*$ et $v_R + a\tau_R = v^* + a\tau_R^*$. Dans la pratique, nous préférons exprimer les solutions intermédiaires en variable primitive $\mathbb{V} = (\tau, Y, v, \Pi, \Sigma)^T$ avec

$$\mathbb{V}_1 = \begin{bmatrix} \tau_L \\ Y_L \\ v_L \\ \Pi_L \\ \Sigma_L \end{bmatrix}, \quad \mathbb{V}_2 = \begin{bmatrix} \tau_L^* \\ Y_L \\ v^* \\ \Pi^* \\ \Sigma_L \end{bmatrix}, \quad \mathbb{V}_3 = \begin{bmatrix} \tau_L^* \\ Y^* \\ v^* \\ \Pi^* \\ \Sigma^* \end{bmatrix}, \quad \mathbb{V}_4 = \begin{bmatrix} \tau_R^* \\ Y^* \\ v^* \\ \Pi^* \\ \Sigma^* \end{bmatrix}, \quad \mathbb{V}_5 = \begin{bmatrix} \tau_R^* \\ Y_R \\ v^* \\ \Pi^* \\ \Sigma_R \end{bmatrix}, \quad \mathbb{V}_6 = \begin{bmatrix} \tau_R \\ Y_R \\ v_R \\ \Pi_R \\ \Sigma_R \end{bmatrix}. \tag{2.25}$$

Notons que la structure d'ondes présentée dans la Fig. 2.1 est uniquement valide pour les vitesses $(\omega_i)_{i=1..5}$ ordonnées de manière croissante, la seule configuration qui nous intéresse. Or, contrairement à (2.16) où les $(\lambda_i)_{i=1,...,5}$ sont naturellement ordonnées, la condition (2.10), *i.e.* $a \gg b > 0$, est insuffisante pour assurer l'ordre des $(\omega_i)_{i=1..5}$. Il est démontré (voir [7]) que l'on a

$$v_L - a\tau_L < v^* - b\tau_L^* < v^* < v^* + b\tau_R < v_R + a\tau_R \tag{2.26}$$

si et seulement si

$$\tau_L^* > 0 \quad \text{et} \quad \tau_R^* > 0. \tag{2.27}$$

Pour assurer la positivité des volumes spécifiques intermédiaires (2.27), le coefficient de relaxation a doit vérifier la condition suivante (voir [7])

$$a > \frac{-(v_R - v_L) + \sqrt{(v_R - v_L)^2 + 8\min(\tau_L, \tau_R)\,|\Pi_R - \Pi_L|}}{4\min(\tau_L, \tau_R)}. \tag{2.28}$$

Notons que la condition (2.28) n'a pas de rapport direct avec la condition de Whitham (2.9a) requise pour assurer la stabilité du système de relaxation. Dans la pratique, a est choisi de manière à satisfaire les deux conditions (2.9a) et (2.28).

En plus de la positivité de τ_L^* et τ_R^*, nous demandons aussi le principe du maximum de l'état intermédiaire Y^*, c'est-à-dire $Y^* \in [0, 1]$. Dans [7], il est établi qu'une condition suffisante pour assurer cette propriété est

$$b > \max\left(|\rho_L \phi(\mathbb{U}_L)|, |\rho_R \phi(\mathbb{U}_R)|\right). \tag{2.29}$$

Comme pour a, la condition (2.29) sur b est à imposer en supplément de la condition de Whitham (2.9b).

2.2 Méthode de relaxation en explicite

Mettant à profit les idées qui viennent d'être expliquées au niveau continu, nous allons construire un schéma de relaxation en coordonnées eulériennes totalement discret explicite en temps.

2.2.1 Discrétisation et notations

Nous considérons une conduite de longueur L que nous découpons en N mailles notées M_i pour $i \in \{1, \ldots, N\}$. Soit x_i le centre de la maille M_i, Δx sa longueur et $x_{i+1/2}$ l'interface séparant deux mailles M_i et M_{i+1}. Soit $\mathbb{U}(x,t)$ la solution exacte du système hyperbolique (2.1). Dans l'esprit d'un schéma volumes finis, l'approximation de $\mathbb{U}_i^n$ à l'instant t^n représente la moyenne de la fonction $x \mapsto \mathbb{U}(x,t^n)$ sur l'intervalle M_i, c'est-à-dire

$$\mathbb{U}_i^n = \frac{1}{\Delta x}\int_{x_{i-1/2}}^{x_{i+1/2}} \mathbb{U}(x,t^n)dx. \tag{2.30}$$

Notons par $\mathbb{U}_{\Delta x}(x,t^n)$ la solution discrète du système équilibre.

2.2.2 Schéma eulérien explicite premier ordre

On souhaite résoudre le système de relaxation (2.5). On suppose connue la solution discrète à l'équilibre $\mathbb{W}_{\Delta x}(x,t^n)$ à l'instant t^n de manière à mettre à jour la solution discrète à l'instant ultérieur $t^{n+1} = t^n + \Delta t$. Ici, $\mathbb{W}_{\Delta x}$ est définit à la date t^n par

$$\mathbb{W}_{\Delta x}(x,t^n) = [\rho(x,t^n), (\rho v)(x,t^n), (\rho\Pi)(x,t^n), (\rho Y)(x,t^n), (\rho\Sigma)(x,t^n)]^T, \tag{2.31}$$

avec

$$\begin{aligned} (\rho\Pi)(x,t^n) &= \rho(x,t^n)P(\mathbb{U}_{\Delta x}(x,t^n)), \\ (\rho\Sigma)(x,t^n) &= \rho(x,t^n)\sigma(\mathbb{U}_{\Delta x}(x,t^n)). \end{aligned} \tag{2.32}$$

Suivant le principe de la relaxation, la résolution de (2.5) se décompose en deux étapes

1. **Étape d'évolution** : on résout le problème de Cauchy pour le système de relaxation (2.5) (en présence du terme source S) en prenant le paramètre de relaxation $\eta = 0$, c'est-à-dire

$$\begin{aligned} \partial_t(\rho) &+ \partial_x(\rho v) &&= 0, \\ \partial_t(\rho v) &+ \partial_x(\rho v^2 + \Pi) &&= S(\mathbb{W}), \\ \partial_t(\rho\Pi) &+ \partial_x(\rho\Pi v + a^2 v) &&= 0, \\ \partial_t(\rho Y) &+ \partial_x(\rho Y v - \Sigma) &&= 0, \\ \partial_t(\rho\Sigma) &+ \partial_x(\rho\Sigma v - b^2 Y) &&= 0. \end{aligned} \tag{2.33}$$

   Sous forme abstraite, ce système s'écrit

$$\partial_t\mathbb{W} + \partial_x\mathcal{G}(\mathbb{W}) = \mathbb{S}(\mathbb{W}), \tag{2.34}$$

   avec $\mathbb{S}(\mathbb{W}) = (0, S(\mathbb{W}), 0, 0, 0)^T$. En discrétisant (2.34) nous avons

$$\mathbb{W}_i^{n+1} = \mathbb{W}_i^n - \frac{\Delta t}{\Delta x}(\mathbb{G}_{i+1/2}^n - \mathbb{G}_{i-1/2}^n) + \Delta t\mathbb{S}(\mathbb{W}_i^n), \tag{2.35}$$

   où $\mathbb{G}_{i+1/2}^n$ est le flux numérique de Godunov définit à partir de la solution du problème de Riemann à l'interface $i+1/2$ à l'instant t^n (voir 2.1.4) par

$$\mathbb{G}_{i+1/2}^n = \mathbb{G}^{Godunov}(\mathbb{W}_i^n, \mathbb{W}_{i+1}^n) = \mathcal{G}(\mathbb{W}_{\Re}(0^+, \mathbb{W}_i^n, \mathbb{W}_{i+1}^n)). \tag{2.36}$$

2. **Étape de la mise à l'équilibre** : on résout le problème d'EDO suivant en laissant le paramètre de relaxation η tendre vers $+\infty$

$$\begin{aligned} \partial_t(\rho) &= 0, \\ \partial_t(\rho v) &= 0, \\ \partial_t(\rho\Pi) &= \eta\rho(P(\mathbb{U}) - \Pi), \\ \partial_t(\rho Y) &= 0, \\ \partial_t(\rho\Sigma) &= \eta\rho(\sigma(\mathbb{U}) - \Sigma). \end{aligned} \tag{2.37}$$

Dans la limite $\eta \to +\infty$, cette étape revient tout simplement à mettre les variables de relaxation à l'équilibre, à savoir pour tout $i \in \mathbb{Z}$

$$\Pi_i^{n+1} = P(\mathbb{U}_i^{n+1}) \qquad \text{et} \qquad \Sigma_i^{n+1} = \sigma(\mathbb{U}_i^{n+1}). \tag{2.38}$$

Remarquons que grâce à cette opération, au début du pas de temps suivant, l'équilibre (2.32) est toujours valable pour $n \leftarrow n+1$, c'est-à-dire

$$\begin{aligned} (\rho\Pi)(x, t^{n+1}) &= \rho(x, t^{n+1}) P(\mathbb{U}_{\Delta x}(x, t^{n+1})), \\ (\rho\Sigma)(x, t^{n+1}) &= \rho(x, t^{n+1}) \sigma(\mathbb{U}_{\Delta x}(x, t^{n+1})). \end{aligned} \tag{2.39}$$

2.2.3 Condition de positivité et principe du maximum

Nous aimerions assurer non seulement la stabilité du système de relaxation mais aussi la positivité de la densité et le principe du maximum de la fraction massique du gaz pour le schéma eulérien explicite premier ordre. Supposons $\rho_i^n > 0$ et $Y_i^n \in [0,1]$, nous souhaitons avoir aussi $\rho_i^{n+1} > 0$ et $Y_i^{n+1} \in [0,1]$.

***Théorème** 2.1. Si les coefficients de relaxation $a_{i+1/2}^n$ et $b_{i+1/2}^n$ vérifient les conditions* (2.28) *et* (2.29)*, c'est-à-dire*

$$a_{i+1/2}^n > a_{i+1/2}^\sharp(\mathbb{U}_i^n, \mathbb{U}_{i+1}^n) \quad et \quad b_{i+1/2}^n > b_{i+1/2}^\sharp(\mathbb{U}_i^n, \mathbb{U}_{i+1}^n), \tag{2.40}$$

avec

$$\begin{aligned} a_{i+1/2}^\sharp(\mathbb{U}_i^n, \mathbb{U}_{i+1}^n) &= \frac{-(v_{i+1}^n - v_i^n) + \sqrt{(v_{i+1}^n - v_i^n)^2 + 8\min(\tau_i^n, \tau_{i+1}^n)|P(\mathbb{U}_{i+1}^n) - P(\mathbb{U}_i^n)|}}{4\min(\tau_i^n, \tau_{i+1}^n)} \\ b_{i+1/2}^\sharp(\mathbb{U}_i^n, \mathbb{U}_{i+1}^n) &= \max(|\rho_i^n \phi(\mathbb{U}_i^n)|, |\rho_{i+1}^n \phi(\mathbb{U}_{i+1}^n)|), \end{aligned} \tag{2.41}$$

et sous la condition CFL

$$\frac{\Delta t}{\Delta x} \max_i \max \left(|v^* - a\tau_L^*|_{i+1/2}^n, |v^* + a\tau_R^*|_{i+1/2}^n \right) < \frac{1}{2}, \tag{2.42}$$

alors nous avons pour le schéma de relaxation explicite premier ordre

$$\rho_i^{n+1} > 0 \quad et \quad Y_i^{n+1} \in [0,1]. \tag{2.43}$$

Démonstration La démonstration se trouve dans [6, 7]. □

Notons que les coefficients a et b définis ci-dessus sont considérés comme deux coefficients dépendant du temps mais variables en espace. Ils sont calculés localement afin de diminuer la diffusion numérique. Dans la pratique nous couplons la condition (2.40) avec la condition de Whitham au niveau discret, à savoir

$$a_{i+1/2}^n = \max \left(a_{i+1/2}^\flat(\mathbb{U}_i^n, \mathbb{U}_{i+1}^n), a_{i+1/2}^\sharp(\mathbb{U}_i^n, \mathbb{U}_{i+1}^n) \right), \tag{2.44}$$

et

$$b^n_{i+1/2} = \max\left(b^\flat_{i+1/2}(\mathbb{U}^n_i, \mathbb{U}^n_{i+1}), b^\sharp_{i+1/2}(\mathbb{U}^n_i, \mathbb{U}^n_{i+1})\right), \tag{2.45}$$

avec

$$a^\flat_{i+1/2}(\mathbb{U}^n_i, \mathbb{U}^n_{i+1}) = \max\left(\sqrt{-\partial_\tau P(\mathbb{U}^n_i) + \left[\partial_v P(\mathbb{U}^n_i)\right]^2}, \sqrt{-\partial_\tau P(\mathbb{U}^n_{i+1}) + \left[\partial_v P(\mathbb{U}^n_{i+1})\right]^2}\right), \tag{2.46}$$

et

$$b^\flat_{i+1/2}(\mathbb{U}^n_i, \mathbb{U}^n_{i+1}) = \max(|\partial_Y \sigma(\mathbb{U}^n_i)|, |\partial_Y \sigma(\mathbb{U}^n_{i+1})|), \tag{2.47}$$

2.3 Méthode de relaxation en semi-implicite

La conception du schéma de relaxation explicite achevée, l'étape suivante consiste à en élaborer une version semi-implicite. Dans le contexte des écoulements pétroliers, les industriels s'intéressent principalement aux ondes lentes (transport de matière). Ils souhaitent, en effet, simuler le comportement des bouchons dans un terrain slugging, c'est-à-dire des discontinuités sur l'onde cinématique, et non sur les ondes acoustiques. Or, les phénomènes acoustiques possèdent des vitesses caractéristiques qui peuvent être jusqu'à 30 fois plus grandes que les vitesses lentes (phénomènes de transport). Par conséquent, un schéma d'intégration en temps explicite comme celui présenté dans la section précédente est inadapté à notre exigence de précision à cause de la condition CFL dépendant des ondes rapides. En fait, cette condition CFL explicite, indispensable pour assurer la stabilité du schéma, impose de très petits pas de temps et entraîne une diffusion trop importante pour les ondes lentes. Par ailleurs un schéma totalement implicite manquerait de précision sur les ondes lentes.

Il est donc nécessaire d'adopter une stratégie d'intégration en temps implicite par rapport aux ondes rapides et explicite par rapport aux ondes lentes sous une condition CFL basée sur les vitesses lentes, comme c'est le cas dans TACITE. On obtiendra un pas de temps adapté (plus grand donc moins de temps de calcul) et une bonne précision pour les ondes lentes.

Le schéma de relaxation semi-implicite que nous présentons dans cette partie a été mise en application dans le cadre de la thèse de Baudin [6]. Dans un premier temps on construit un schéma implicite linéarisé puis semi-implicite linéaire en se basant sur une linéarisation du flux de Roe. Une version d'implicitation de la projection a été aussi mise en œuvre afin de rendre explicite les ondes lentes.

2.3.1 Schéma implicite linéarisé

Considérons, pour commencer, le schéma totalement implicite

$$\mathbb{W}^{n+1}_i = \mathbb{W}^n_i - \frac{\Delta t}{\Delta x}\left[\mathbb{G}(\mathbb{W}^{n+1}_i, \mathbb{W}^{n+1}_{i+1}) - \mathbb{G}(\mathbb{W}^{n+1}_{i-1}, \mathbb{W}^{n+1}_i)\right] + \Delta t \mathbb{S}(\mathbb{W}^{n+1}_i). \tag{2.48}$$

Le second membre dépend de la solution à l'instant t^{n+1}. La résolution de (2.48) nécessite donc d'inverser un problème non-linéaire et le coût CPU est élevé vu la taille du problème. À l'instar du schéma VFRoe [49,51,72], on se contente d'un schéma implicite linéarisé. Les flux $\mathbb{G}$ et le terme source $\mathbb{S}$ sont approchés par leur développement de Taylor au premier ordre

$$\begin{aligned} \mathbb{G}(\mathbb{W}^{n+1}_i, \mathbb{W}^{n+1}_{i+1}) &\simeq \mathbb{G}(\mathbb{W}^n_i, \mathbb{W}^n_{i+1}) + \nabla_L \mathbb{G}(\mathbb{W}^n_i, \mathbb{W}^n_{i+1})\delta\mathbb{W}_i + \nabla_R \mathbb{G}(\mathbb{W}^n_i, \mathbb{W}^n_{i+1})\delta\mathbb{W}_{i+1}, \\ \mathbb{S}(\mathbb{W}^{n+1}_i) &\simeq \mathbb{S}(\mathbb{W}^n_i) + \nabla\mathbb{S}(\mathbb{W}^n_i)\delta\mathbb{W}_i \end{aligned} \tag{2.49}$$

avec $\nabla_L \mathbb{G}(\mathbb{W}_L, \mathbb{W}_R)$ la matrice jacobienne de $\mathbb{G}$ par rapport à $\mathbb{W}_L$ et $\nabla_R \mathbb{G}(\mathbb{W}_L, \mathbb{W}_R)$ la matrice jacobienne par rapport à $\mathbb{W}_R$. L'incrément $\delta\mathbb{W}_i$ est tout simplement $\delta\mathbb{W}_i = \mathbb{W}^{n+1}_i - \mathbb{W}^n_i$.

Introduisons

$$\mathbf{C}_i = -\frac{\Delta t}{\Delta x}\nabla_L \mathbb{G}(\mathbb{W}_i^n, \mathbb{W}_{i+1}^n), \quad \mathbf{E}_{i+1} = \frac{\Delta t}{\Delta x}\nabla_R \mathbb{G}(\mathbb{W}_i^n, \mathbb{W}_{i+1}^n), \tag{2.50a}$$

$$\mathbf{D}_i = \mathbb{I} - \mathbf{C}_i - \mathbf{E}_i - \Delta t \nabla \mathbb{S}_i, \tag{2.50b}$$

avec $\mathbb{I}$ matrice d'identité 5×5. Alors, le schéma implicite linéarisé s'écrit

$$\begin{aligned}\mathbb{W}_i^{n+1} = \mathbb{W}_i^n - \frac{\Delta t}{\Delta x}\left[\mathbb{G}(\mathbb{W}_i^n, \mathbb{W}_{i+1}^n) - \mathbb{G}(\mathbb{W}_{i-1}^n, \mathbb{W}_i^n)\right] + \Delta t \mathbb{S}(\mathbb{W}_i^n) \\ - \mathbf{C}_{i-1}\delta\mathbb{W}_{i-1} - (\mathbf{D}_i - \mathbb{I})\delta\mathbb{W}_i - \mathbf{E}_{i+1}\delta\mathbb{W}_{i+1},\end{aligned} \tag{2.51}$$

ce qui revient à ne faire qu'une seule itération de Newton pour (2.48). La résolution de (2.51) se fait donc en quatre étapes

- Étape physique où on calcule la solution « explicite » avec le terme explicite connu dans le second-membre

$$\mathbb{W}_i^{n\#} = \mathbb{W}_i^n - \frac{\Delta t}{\Delta x}\left[\mathbb{G}(\mathbb{W}_i^n, \mathbb{W}_{i+1}^n) - \mathbb{G}(\mathbb{W}_{i-1}^n, \mathbb{W}_i^n)\right] + \Delta t \mathbb{S}(\mathbb{W}_i^n). \tag{2.52}$$

- Étape mathématique où on doit résoudre un système linéaire des inconnues $\delta\mathbb{W}_i$

$$\mathbf{C}_{i-1}\delta\mathbb{W}_{i-1} + \mathbf{D}_i\delta\mathbb{W}_i + \mathbf{E}_{i+1}\delta\mathbb{W}_{i+1} = \mathbb{W}_i^{n\#} - \mathbb{W}_i^n. \tag{2.53}$$

- Étape de mise à jour de la solution conservative

$$\mathbb{W}_i^{n+1} = \mathbb{W}_i^n + \delta\mathbb{W}_i. \tag{2.54}$$

- Étape de projection à l'équilibre des variables de relaxation

$$\begin{aligned}(\rho\Sigma)_i^{n+1} &= \rho_i^{n+1}\sigma(\mathbb{W}_i^{n+1}), \\ (\rho\Pi)_i^{n+1} &= \rho_i^{n+1}P(\mathbb{W}_i^{n+1}).\end{aligned} \tag{2.55}$$

La première étape coïncide avec le schéma explicite que l'on sait résoudre. La difficulté réside dans la résolution du système (2.53) qui peut s'écrire sous la forme matricielle par bloc

$$\begin{bmatrix} \mathbf{D}_0 & \mathbf{E}_1 & 0 & \cdots & \cdots & \cdots & \cdots & 0 \\ \mathbf{C}_0 & \mathbf{D}_1 & \mathbf{E}_2 & \ddots & & & & \vdots \\ 0 & \mathbf{C}_1 & \mathbf{D}_2 & \mathbf{E}_3 & \ddots & & & \vdots \\ \vdots & \ddots & \ddots & \ddots & \ddots & \ddots & & \vdots \\ \vdots & & \ddots & \ddots & \ddots & \ddots & \ddots & \vdots \\ \vdots & & & \ddots & \mathbf{C}_{N-2} & \mathbf{D}_{N-1} & \mathbf{E}_N & 0 \\ \vdots & & & & \ddots & \mathbf{C}_{N-1} & \mathbf{D}_N & \mathbf{E}_{N+1} \\ 0 & \cdots & \cdots & \cdots & \cdots & 0 & \mathbf{C}_N & \mathbf{D}_{N+1} \end{bmatrix} \cdot \begin{bmatrix} \delta\mathbb{W}_0 \\ \delta\mathbb{W}_1 \\ \delta\mathbb{W}_2 \\ \cdots \\ \cdots \\ \delta\mathbb{W}_{N-1} \\ \delta\mathbb{W}_N \\ \delta\mathbb{W}_{N+1} \end{bmatrix} = \begin{bmatrix} \mathcal{R}_0 \\ \mathcal{R}_1 \\ \mathcal{R}_2 \\ \cdots \\ \cdots \\ \mathcal{R}_{N-1} \\ \mathcal{R}_N \\ \mathcal{R}_{N+1} \end{bmatrix}, \tag{2.56}$$

avec les résidus explicites

$$\begin{aligned}&\mathcal{R}_i = -\frac{\Delta t}{\Delta x}\left[\mathbb{G}(\mathbb{W}_i^n, \mathbb{W}_{i+1}^n) - \mathbb{G}(\mathbb{W}_{i-1}^n, \mathbb{W}_i^n)\right] + \Delta t \mathbb{S}(\mathbb{W}_i^n), \quad i \in \{1, \ldots, N\}, \\ &\mathcal{R}_0 = 0 \quad \text{et} \quad \mathcal{R}_{N+1} = 0.\end{aligned} \tag{2.57}$$

Notons que pour le traitement des conditions limites en implicite, dans l'étape mathématique les états fictifs sont inchangés. Ceci revient à prendre $\delta\mathbb{W}_0 = \delta\mathbb{W}_{N+1} = 0$ et par conséquent il suffit de prendre $\mathbf{D}_0 = \mathbf{D}_{N+1} = \mathbf{I}$ et $\mathbf{E}_1 = \mathbf{C}_N = 0$. Pour construire les matrices $\mathbf{C}_i$ et $\mathbf{E}_i$ il faut définir la fonction flux. Il est facile d'expliciter ces matrices dès que l'on a un schéma de type Roe. Nous verrons dans la section suivante comment exprimer les flux de relaxation sous forme des flux de Roe et nous allons donner les détails de ces matrices.

2.3.2 Flux numérique de type Roe pour le schéma implicite linéarisé

L'évaluation des matrices $\nabla_L\mathbb{G}$ et $\nabla_R\mathbb{G}$ pose a priori un problème théorique sérieux, car il est bien connu que le flux de Godunov n'est pas différentiable. C'est pourquoi nous nous intéressons à la réécriture du flux de Godunov défini par (2.36) sous la forme d'un flux de Roe, ce qui permet d'isoler la partie non-différentiable des $\nabla_L\mathbb{G}$ et $\nabla_R\mathbb{G}$. Au niveau mathématique, une telle reformulation est possible grâce à la dégénérescence linéaire des champs.

Nous rappelons qu'un flux numérique $\mathbb{G}(\mathbb{W}_i, \mathbb{W}_{i+1})$ est dit de Roe s'il peut se mettre sous la forme

$$\mathbb{G}^{Roe}(\mathbb{W}_i, \mathbb{W}_{i+1}) = \frac{1}{2}\left[\mathcal{G}(\mathbb{W}_i) + \mathcal{G}(\mathbb{W}_{i+1})\right] - \frac{1}{2}|\mathcal{A}(\mathbb{W}_i, \mathbb{W}_{i+1})|(\mathbb{W}_{i+1} - \mathbb{W}_i), \tag{2.58}$$

où $\mathcal{A}(\mathbb{W}_i, \mathbb{W}_{i+1}) : \Omega \times \Omega \to \mathrm{Mat}(\mathbb{R}^5)$, appelée matrice de Roe, doit vérifier les propriétés suivantes :

1. $\mathcal{A}(\mathbb{W}, \mathbb{W}) = \nabla\mathcal{G}(\mathbb{W})$,
2. $\mathcal{A}(\mathbb{W}_i, \mathbb{W}_{i+1})$ est diagonalisable,
3. $\mathcal{A}(\mathbb{W}_i, \mathbb{W}_{i+1})(\mathbb{W}_{i+1} - \mathbb{W}_i) = \mathcal{G}(\mathbb{W}_{i+1}) - \mathcal{G}(\mathbb{W}_i)$,

pour tous les états $\mathbb{W}_i, \mathbb{W}_{i+1}$ dans Ω. Ici, nous avons supprimé l'indice en temps n pour abréger les notations.

Nous tentons maintenant de chercher une matrice $\mathcal{A}(\mathbb{W}_i, \mathbb{W}_{i+1})$ qui permet d'écrire le flux de Godunov $\mathbb{G}^{Godunov}(\mathbb{W}_i, \mathbb{W}_{i+1})$ sous forme de (2.58). Commençons par un résultat technique.

***Lemme** 2.2. Si les coefficients de relaxation locaux $a_{i+1/2}$ et $b_{i+1/2}$ vérifient la condition* (2.40) *alors la matrice des vecteurs propres à droite associés au problème de Riemann* (2.20) *pour le système de relaxation* (2.5)*, donnée par*

$$R(\mathbb{W}_i, \mathbb{W}_{i+1}) = \begin{bmatrix} 1 & 0 & 1 & 0 & 1 \\ v_i - a_{i+1/2}\tau_i & 0 & v^*_{i+1/2} & 0 & v_{i+1} + a_{i+1/2}\tau_{i+1} \\ \Pi_i + (a_{i+1/2})^2\tau_i & 0 & \Pi^*_{i+1/2} & 0 & \Pi_{i+1} + (a_{i+1/2})^2\tau_{i+1} \\ Y_i & 1 & Y^*_{i+1/2} & 1 & Y_{i+1} \\ \Sigma_i & b_{i+1/2} & \Sigma^*_{i+1/2} & -b_{i+1/2} & \Sigma_{i+1} \end{bmatrix}, \tag{2.59}$$

avec

$$\tau^*_i = \tau_i + \frac{v_{i+1} - v_i}{2a_{i+1/2}} - \frac{\Pi_{i+1} - \Pi_i}{2(a_{i+1/2})^2}, \qquad \tau^*_{i+1} = \tau_{i+1} + \frac{v_{i+1} - v_i}{2a_{i+1/2}} + \frac{\Pi_{i+1} - \Pi_i}{2(a_{i+1/2})^2},$$

$$Y^*_{i+1/2} = \frac{Y_i + Y_{i+1}}{2} + \frac{\Sigma_{i+1} - \Sigma_i}{2b_{i+1/2}}, \qquad \Pi^*_{i+1/2} = \frac{\Pi_{i+1} + \Pi_i}{2} - a_{i+1/2}\frac{v_{i+1} - v_i}{2}, \tag{2.60}$$

$$v^*_{i+1/2} = \frac{v_{i+1} + v_i}{2} - \frac{\Pi_{i+1} - \Pi_i}{2a_{i+1/2}}, \qquad \Sigma^*_{i+1/2} = \frac{\Sigma_i + \Sigma_{i+1}}{2} + b_{i+1/2}\frac{Y_{i+1} - Y_i}{2},$$

est inversible.

Démonstration La démonstration de ce lemme se trouve dans [7]. □

À partir de cette matrice R inversible, nous pouvons exhiber une matrice de Roe répondant au désir de réécriture (2.58).

***Théorème** 2.3. Si $a_{i+1/2} > b_{i+1/2} > 0$ et vérifient la condition (2.41) pour tout $\mathbb{W}_i, \mathbb{W}_{i+1}$ dans Ω, alors il existe une matrice de Roe $\mathcal{A}^{Roe}(\mathbb{W}_i, \mathbb{W}_{i+1})$ telle que*

$$\mathbb{G}^{Godunov}(\mathbb{W}_i, \mathbb{W}_{i+1}) = \mathbb{G}^{Roe}(\mathbb{W}_i, \mathbb{W}_{i+1}) = \frac{1}{2}\left[\mathcal{G}(\mathbb{W}_i) + \mathcal{G}(\mathbb{W}_{i+1})\right] - \frac{1}{2}|\mathcal{A}^{Roe}(\mathbb{W}_i, \mathbb{W}_{i+1})|(\mathbb{W}_{i+1} - \mathbb{W}_i). \tag{2.61}$$

Cette matrice de Roe est définie par

$$\mathcal{A}^{Roe}(\mathbb{W}_i, \mathbb{W}_{i+1}) = R(\mathbb{W}_i, \mathbb{W}_{i+1})\Lambda(\mathbb{W}_i, \mathbb{W}_{i+1})R^{-1}(\mathbb{W}_i, \mathbb{W}_{i+1}), \tag{2.62}$$

avec Λ la matrice diagonale des valeurs propres $(\omega_j)_{j=1,\dots,5}$ définies en (2.24)

$$\Lambda(\mathbb{W}_i, \mathbb{W}_{i+1}) = Diag(v_i - a_{i+1/2}\tau_i, v^*_{i+1/2} - b_{i+1/2}\tau^*_i, v^*_{i+1/2}, v^*_{i+1/2} + b_{i+1/2}\tau^*_{i+1}, v_{i+1} + a_{i+1/2}\tau_{i+1}), \tag{2.63}$$

et $R(\mathbb{W}_i, \mathbb{W}_{i+1})$ retrouve sa définition dans (2.59).

Démonstration Les lecteurs peuvent voir la démonstration dans [6, 7]. □

Grâce à l'existence d'une telle matrice de Roe, au lieu de travailler avec le système initial (2.34) on travaillera avec le système linéarisé

$$\partial_t \mathbb{W} + \mathcal{A}^{Roe}(\mathbb{W}_i, \mathbb{W}_{i+1})\partial_x \mathbb{W} = 0 \tag{2.64}$$

à chaque interface $i+1/2$. Les flux numériques correspondants sont donnés par (2.61). De plus, comme déjà mentionné, le fait que l'on puisse exprimer le flux de Godunov comme flux de Roe nous permet d'isoler la partie non-différentiable des $\nabla_L\mathbb{G}$ et $\nabla_R\mathbb{G}$ dans la valeur absolue $|\mathcal{A}^{Roe}|$. Nous optons alors pour une approximation supplémentaire, qui consiste à « geler » cette partie dans les dérivées. Ceci conduit à

$$\begin{aligned} \nabla_L\mathbb{G}(\mathbb{W}_i, \mathbb{W}_{i+1}) &\simeq \frac{1}{2}\Big[\nabla\mathcal{G}(\mathbb{W}_i) \quad + |\mathcal{A}^{Roe}(\mathbb{W}_i, \mathbb{W}_{i+1})|\Big], \\ \nabla_R\mathbb{G}(\mathbb{W}_i, \mathbb{W}_{i+1}) &\simeq \frac{1}{2}\Big[\nabla\mathcal{G}(\mathbb{W}_{i+1}) - |\mathcal{A}^{Roe}(\mathbb{W}_i, \mathbb{W}_{i+1})|\Big]. \end{aligned} \tag{2.65}$$

Nous pouvons ainsi expliciter les matrices $\mathbf{C}_i$ et $\mathbf{E}_{i+1}$ définies par (2.50a), à savoir

$$\mathbf{C}_i = -\frac{\Delta t}{2\Delta x}\Big[\nabla\mathcal{G}(\mathbb{W}_i) + |\mathcal{A}^{Roe}(\mathbb{W}_i, \mathbb{W}_{i+1})|\Big] \quad \text{et} \quad \mathbf{E}_{i+1} = \frac{\Delta t}{2\Delta x}\Big[\nabla\mathcal{G}(\mathbb{W}_{i+1}) - |\mathcal{A}^{Roe}(\mathbb{W}_i, \mathbb{W}_{i+1})|\Big]. \tag{2.66}$$

2.3.3 Schéma relaxation semi-implicite linéaire

On souhaite construire un schéma semi-implicite dont le traitement est explicite sur les ondes lentes v et $v \pm b\tau$ et implicite sur les ondes rapides $v \pm a\tau$. En conséquence, lors de la construction du système linéaire par blocs dans la phase implicite nous ne gardons que les champs associés aux ondes rapides. Concrètement, on va éliminer la contribution des ondes lentes dans les matrices issues de la linéarisation, dans le même esprit de ce qui avait été proposé par Gallouët et *al.* [51, 72].

Définissons tout d'abord l'opération $\sim$ appliquée à une matrice diagonalisable $\mathcal{X}$. Cette opération consiste à diagonaliser la matrice $\mathcal{X}$

$$\mathcal{X} = R\Lambda R^{-1}, \tag{2.67}$$

avec

$$\Lambda = Diag(\lambda_1, \lambda_2, \lambda_3, \lambda_4, \lambda_5), \tag{2.68}$$

puis à annuler les valeurs propres correspondant aux ondes lentes

$$\widetilde{\Lambda} = Diag(\lambda_1, 0, 0, 0, \lambda_5), \tag{2.69}$$

et enfin à revenir à la base initiale pour obtenir $\widetilde{\mathcal{X}}$

$$\widetilde{\mathcal{X}} = R\widetilde{\Lambda}R^{-1}. \tag{2.70}$$

Le schéma de relaxation semi-implicite linéaire est obtenu en remplaçant

- $\nabla\mathcal{G}(\mathbb{W}_i)$ par $\widetilde{\nabla\mathcal{G}}(\mathbb{W}_i)$,
- $\nabla\mathcal{G}(\mathbb{W}_{i+1})$ par $\widetilde{\nabla\mathcal{G}}(\mathbb{W}_{i+1})$,
- $\mathcal{A}^{Roe}(\mathbb{W}_i, \mathbb{W}_{i+1})$ par $\widetilde{\mathcal{A}}^{Roe}(\mathbb{W}_i, \mathbb{W}_{i+1})$.

Les approximations (2.65) pour les matrices $\nabla_L\mathbb{G}$ et $\nabla_R\mathbb{G}$ deviennent alors

$$\begin{aligned}
\widetilde{\nabla_L\mathbb{G}}(\mathbb{W}_i, \mathbb{W}_{i+1}) &\simeq \frac{1}{2}\Big[\widetilde{\nabla\mathcal{G}}(\mathbb{W}_i) \quad + |\widetilde{\mathcal{A}}^{Roe}(\mathbb{W}_i, \mathbb{W}_{i+1})|\Big], \\
\widetilde{\nabla_R\mathbb{G}}(\mathbb{W}_i, \mathbb{W}_{i+1}) &\simeq \frac{1}{2}\Big[\widetilde{\nabla\mathcal{G}}(\mathbb{W}_{i+1}) - |\widetilde{\mathcal{A}}^{Roe}(\mathbb{W}_i, \mathbb{W}_{i+1})|\Big].
\end{aligned} \tag{2.71}$$

Les nouvelles matrices de $\mathbf{C}_i$, $\mathbf{E}_{i+1}$ et $\mathbf{D}_{i+1}$ définies dans (2.50a) et (2.50b) sont maintenant

$$\widetilde{\mathbf{C}}_i = -\frac{\Delta t}{\Delta x}\widetilde{\nabla_L\mathbb{G}}(\mathbb{W}_i^n, \mathbb{W}_{i+1}^n), \quad \widetilde{\mathbf{E}}_{i+1} = \frac{\Delta t}{\Delta x}\widetilde{\nabla_R\mathbb{G}}(\mathbb{W}_i^n, \mathbb{W}_{i+1}^n), \tag{2.72}$$

et

$$\widetilde{\mathbf{D}}_i = \mathbb{I} - \widetilde{\mathbf{C}}_i - \widetilde{\mathbf{E}}_i - \Delta t\nabla\mathbb{S}_i. \tag{2.73}$$

On remarque que cette stratégie est exacte pour un système hyperbolique linéaire. En non-linéaire, cette modification pour rendre explicite le transport de matière n'est qu'une heuristique puisque $\nabla\mathcal{G}(\mathbb{W}_i)$, $\nabla\mathcal{G}(\mathbb{W}_{i+1})$ et $\mathcal{A}^{Roe}(\mathbb{W}_i, \mathbb{W}_{i+1})$ ne sont pas diagonalisables dans la même base.

2.3.4 Implicitation de la projection

Un ingrédient supplémentaire dans la conception des schémas de relaxation en implicite est ce que nous appelons la projection des variables de relaxation (2.55). Dans Chalons [19] il est démontré que l'état stationnaire obtenu avec cette méthode n'est pas précis. Un remède consiste à lier l'évolution des variables de relaxation associées au bloc implicite $(\rho\Pi)$ à l'évolution des variables d'équilibre $(\rho, \rho Y, \rho v)$. Cette implicitation de la projection des variables de relaxation à l'équilibre se traduit par

$$\delta(\rho\Pi)_i = \partial_\rho(\rho P)(\mathbb{W}_i)\delta\rho_i + \partial_{\rho Y}(\rho P)(\mathbb{W}_i)\delta(\rho Y)_i + \partial_{\rho v}(\rho P)(\mathbb{W}_i)\delta(\rho v)_i. \tag{2.74}$$

Cette méthode permet non seulement d'avoir un état stationnaire plus précis mais aussi de diminuer la taille du système linéaire à résoudre (2.56). En effet, en remplaçant l'incrément $\delta(\rho\Pi)_i$ par sa linéarisation (2.74), on peut éliminer la troisième ligne des matrices $\mathbf{C}_i, \mathbf{D}_i$ et $\mathbf{E}_i$ tout en rajoutant la contribution de $\delta\rho_i, \delta(\rho Y)_i$ et $\delta(\rho v)_i$ provenant de cette implicitation à la première, deuxième et quatrième lignes de ces matrices. En conséquence, au lieu d'inverser un système linéaire par blocs de tailles 5×5, on ne travaille qu'avec des blocs de taille 4×4, ce qui permet de gagner en temps CPU lors de la résolution.

2.3.5 Choix de coefficients de relaxation et positivité du schéma implicite et semi-implicite

Nous rappelons qu'en explicite nous savons calculer les coefficients de relaxation a et b suffisamment grands de manière à assurer :

- la stabilité du système de relaxation dans la limite d'infini du paramètre de relaxation η,
- les propriétés physiques de la solution discrétisée, à savoir la positivité de la densité et le principe du maximum sur la fraction massique du gaz.

Ces coefficients sont choisis localement sur chaque interface afin de diminuer la diffusion du schéma explicite

$$a^n_{i+1/2} = \max\left(a^\flat_{i+1/2}(\mathbb{W}^n_i, \mathbb{W}^n_{i+1}), a^\sharp_{i+1/2}(\mathbb{W}^n_i, \mathbb{W}^n_{i+1})\right), \tag{2.75a}$$

$$b^n_{i+1/2} = \max\left(b^\flat_{i+1/2}(\mathbb{W}^n_i, \mathbb{W}^n_{i+1}), b^\sharp_{i+1/2}(\mathbb{W}^n_i, \mathbb{W}^n_{i+1})\right), \tag{2.75b}$$

où $a^\flat_{i+1/2}, a^\sharp_{i+1/2}, b^\flat_{i+1/2}$ et $b^\sharp_{i+1/2}$ sont définis par (2.41),(2.46) et (2.47).

En implicite ou semi-implicite, le choix de ces coefficients dépend de la méthode utilisée. En effet, l'expérience montre que :

- pour le schéma implicite linéarisé (2.51) où tous les champs sont implicités, les coefficients doivent être constants ce qui donne une diffusion importante sur toutes les ondes.

$$a^n = \max_i(a^n_{i+1/2}), \tag{2.76a}$$

$$b^n = \max_i(b^n_{i+1/2}). \tag{2.76b}$$

- pour le schéma semi-implicite où on n'implicite que les ondes rapides $v \pm a\tau$, le coefficient b peut être choisi localement suivant (2.75b). Si on utilise la méthode d'implicitation de la projection (2.74) le coefficient a peut être choisi local comme dans (2.75a), sinon il doit être pris constant suivant (2.76a).

En pratique, on utilise le schéma de relaxation semi-implicite avec implicitation de la projection où les calculs sont simplifiés grâce à la structure très simple de la matrice diagonale creuse $\widetilde{\Lambda}$. Dans ce cas, le fait de choisir les coefficients a et b localement sur les interfaces conduit à une diffusion raisonable sur les ondes rapides non pertinentes pour notre application, tandis que les ondes lentes ou ondes de transport qui nous intéressent restent toujours précises.

Nous insistons dans le fait que contrairement au cas explicite, quel que soit le choix de a et b le schéma semi-implicite ne garantit ni la positivité de la densité ni le principe du maximum sur la fraction massique du gaz. Le calcul du pas de temps Δt est basé sur les ondes lentes et rapides pour éviter les interactions entre les ondes, ce qui ne donne aucune information sur la positivité du schéma. Nous verrons dans le chapitre suivant un autre schéma dans lequel l'estimation a priori du pas de temps permet de garantir ces propriétés.

2.3.6 Calcul du pas de temps

En semi-implicite, nous calculons un pas de temps lié aux ondes lentes v et $v \pm b\tau$ sous une CFL explicite et un pas de temps lié aux ondes rapides $v \pm a\tau$ sous une CFL implicite. À chaque interface $i+1/2$ on calcule deux pas de temps

- Pas de temps sous contrainte explicite

$$\frac{\Delta t^{exp}}{\Delta x} \max_i \max \left(|v^* - b\tau_L^*|_{i+1/2}^n, |v_{i+1/2}^*|, |v^* + b\tau_R^*|_{i+1/2}^n \right) < \mathrm{CFL}^{exp}. \tag{2.77}$$

- Pas de temps sous contrainte implicite

$$\frac{\Delta t^{imp}}{\Delta x} \max_i \max \left(|v^* - a\tau_L^*|_{i+1/2}^n, |v^* + a\tau_R^*|_{i+1/2}^n \right) < \mathrm{CFL}^{imp}, \tag{2.78}$$

où les conditions CFL sont : $\mathrm{CFL}^{exp} = 0.5$ et $\mathrm{CFL}^{imp} = 20$.

Dans (2.77) et (2.78), les coeffcients a et b sont choisis localement ou globalement dépendant de la méthode numérique choisie. Le pas de temps final est le minimum des ces deux pas de temps calculé sur toutes les interfaces

$$\Delta t = \min(\Delta t^{exp}, \Delta t^{imp}). \tag{2.79}$$

***Remarque** 2.4.* La $\mathrm{CFL}^{exp} = 0.5$ est classique pour éviter les interactions entres les ondes dans une cellule. En revanche, la $\mathrm{CFL}^{imp} = 20$ est déterminée expérimentalement pour avoir un bon compromis entre un grand pas de temps et une diffusion numérique acceptable des ondes rapides.

2.4 Traitement des conditions aux limites

Les conditions aux limites constituent un enjeu important dans la résolution des problèmes d'écoulements qui nous préoccupent : c'est en effet par les conditions aux limites que l'information est injectée dans le système. Il se peut qu'il n'y ait pas de solution si ces conditions sont mal posées. La première question est donc (i) est-ce que les conditions limites que l'on souhaite imposer sont bien posées au niveau continu? (ii) comment les prescrire au niveau discret? Pour répondre à ces questions, il est nécessaire de revisiter quelques notions de base concernant les conditions aux limites pour les systèmes hyperboliques en général, après quoi on pourra les appliquer au système de relaxation.

2.4.1 Conditions aux limites pour un système hyperbolique général

Considérons le système hyperbolique suivant

$$\partial_t \mathbb{U} + \partial_x \mathcal{F}(\mathbb{U}) = 0, \qquad \mathbb{U} \in \Omega \subset \mathbb{R}^p, x > 0, t > 0, \tag{2.80}$$

où p est un entier positif et $\mathcal{F} : \Omega \subset \mathbb{R}^p \longrightarrow \mathbb{R}^p$ est non-linéaire. On suppose que les champs caractéristiques sont soit vraiment non-linéaires soit linéairement dégénérés.

2.4.1.1 Cas des conditions complètes

Le système (2.80) est muni d'une condition initiale

$$\mathbb{U}(x, t = 0) = \mathbb{U}^0(x), \qquad x > 0, \tag{2.81}$$

et des conditions aux limites « virtuelles »

$$\mathbb{U}(x = 0, t) = \mathbb{U}_b(t), \qquad t > 0. \tag{2.82}$$

L'épithète « virtuel » est à comprendre dans le sens suivant. Il se peut que l'égalité (2.82) soit impossible à satisfaire complètement, car elle entre en conflit avec l'équation intérieure (2.80). Dans ce cas, on est obligé de l'affaiblir d'une façon ou d'une autre. En la matière, il y a différentes stratégies, notamment celle de Benabdallah et Serre [12] basée sur une formulation entropique et celle de

Dubois-LeFloch [46] basée sur les demi-problèmes de Riemann. C'est ce deuxième choix que nous allons développer.

Pour tout état $\mathbb{U}_b \in \Omega$, on introduit

$$\mathcal{V}(\mathbb{U}_b) = \{\mathbb{W} \in \Omega \text{ t.q. } \mathbb{W} = \mathbb{W}_{\mathfrak{R}}(0^+; \mathbb{U}_b, \mathbb{V}), \mathbb{V} \in \Omega\}, \tag{2.83}$$

où $\mathbb{W}_{\mathfrak{R}}$ désigne la solution du problème de Riemann (supposée existante). C'est l'ensemble des solutions en $x/t = 0^+$ de tous les problèmes de Riemann associés à (2.80) et admettant $\mathbb{U}_b$ comme donnée initiale gauche. Alors, dans la philosophie du Dubois-LeFloche, on remplace le problème (2.80)–(2.82) par

$$\begin{aligned} &\partial_t \mathbb{U} + \partial_x \mathcal{F}(\mathbb{U}) = 0, && \mathbb{U} \in \Omega \subset \mathbb{R}^p, x > 0, t > 0, \\ &\mathbb{U}(x, t = 0) = \mathbb{U}^0(x), && x > 0, \\ &\mathbb{U}(0^+, t) \in \mathcal{V}(\mathbb{U}_b(t)), && t > 0. \end{aligned} \tag{2.84}$$

Autrement dit, le modélisateur propose $\mathbb{U}_b(t)$, mais c'est le système hyperbolique (2.80) qui dispose, par l'intermédiaire de $\mathcal{V}$.

2.4.1.2 Cas des conditions incomplètes

Au lieu de vouloir imposer tout le vecteur $\mathbb{U}_b(t)$ en $x = 0$, on cherche maintenant à y prescrire

$$\Phi(\mathbb{U}(0, t), t) = 0, \quad \text{avec } \Phi : \Omega \times (0, +\infty) \to \mathbb{R}^q. \tag{2.85}$$

Ce cas de figure est dit incomplet, car $q \leq p$.

Plaçons-nous d'abord dans le cas complètement linéaire. Ceci signifie que :

1. Le flux du système est linéaire, *i.e.* $\mathcal{F}(\mathbb{U}) = \mathbf{A}\mathbb{U}$.
   Soient $\lambda_1 < \lambda_2 < \cdots < \lambda_p$ les valeurs propres (constantes) de $\mathbf{A}$. On suppose qu'il y a exactement q valeurs propres entrantes et $p - q$ valeurs propres sortantes. Ainsi nous avons

$$\begin{aligned} \lambda_1 < \lambda_2 < \cdots < \lambda_{p-q} < 0, \\ 0 < \lambda_{p-q+1} < \cdots < \lambda_{p-1} < \lambda_p. \end{aligned} \tag{2.86}$$

   Pour abréger, on notera

$$\begin{aligned} M &= \{1, 2, \ldots, p - q\}, \\ I &= \{p - q + 1, \ldots, p - 1, p\}, \end{aligned} \tag{2.87}$$

   les ensembles d'indices correspondants.

2. Les conditions qu'on souhaite imposer sont linéaires, c'est-à-dire

$$\Phi(\mathbb{U}) = E\mathbb{U} - f, \tag{2.88}$$

   où E est une matrice $p \times q$.

De la linéarité du flux, il découle que pour tout $\mathbb{U}_b \in \Omega$, l'ensemble $\mathcal{V}(\mathbb{U}_b)$ défini en (2.83) n'est autre que

$$\mathcal{V}(\mathbb{U}_b) = \left\{\mathbb{U} \in \Omega \text{ t.q. } L^I \mathbb{U} = L^I \mathbb{U}_b\right\}, \tag{2.89}$$

où L^I est la matrice $q \times p$ des vecteurs propres à gauche correspondant aux $\{\lambda_i\}_{i\in I}$. Ceci est dû à ce que les grandeurs

$$\alpha_i = l_i.\mathbb{U}, \tag{2.90}$$

(l_i étant le vecteur propre à gauche de λ_i) sont les coefficients de la décomposition

$$\mathbb{U} = \sum_{i\in I} \alpha_i r_i + \sum_{m\in M} \alpha_m r_m, \tag{2.91}$$

et sont régis par l'équation de transport

$$\partial_t \alpha_i + \lambda_i \partial_x \alpha_i = 0. \tag{2.92}$$

Autrement dit, quand bien même on se donne la condition complète $\mathbb{U}_b$, tout ce qui compte sera finalement l'ensemble des α_i entrants, à savoir $L^I\mathbb{U}_b$. La valeur $L^M\mathbb{U}_b$ des α_m sortants n'a pas d'importance et peut être prescrite arbitrairement.

La question qui se pose est de savoir si, à partir des données incomplètes $E\mathbb{U} = f$, on est capable de déterminer entièrement $L^I\mathbb{U}$. Par multiplication de (2.91) par E, on obtient

$$\begin{aligned} f &= \sum_{i\in I} \alpha_i E r_i + \sum_{m\in M} \alpha_m E r_m \\ &= (ER^I)(L^I\mathbb{U}) + (ER^M)(L^M\mathbb{U}) \end{aligned} \tag{2.93}$$

où l'on a noté R^I (resp. R^M) la matrice des vecteurs propres à droite correspondant aux $\{\lambda_i\}_{i\in I}$ (resp. $\{\lambda_m\}_{i\in M}$). Les composantes de $L^M\mathbb{U}$ sont arbitrairement données. Il est clair alors qu'une condition nécessaire et suffisante pour que $L^I\mathbb{U}$ soit bien déterminé est que la matrice carrée ER^I, de taille $q\times q$, soit inversible.

Revenons à présent au cas non-linéaire. On peut appliquer le raisonnement précédent au système linéarisé et obtenir la condition de Kreiss [62]

$$\det \nabla_{\mathbb{U}}\Phi(\mathbb{U}) R^I(\mathbb{U}) \neq 0. \tag{2.94}$$

Les détails sont dans [52]. La différence toutefois avec le cas linéaire est que comme le signe des valeurs propres $\lambda_i(\mathbb{U})$ peut dépendre de $\mathbb{U}$, on n'est pas certain d'avoir toujours exactement q valeurs propres entrantes. Cela étant, on procède de la manière suivante :

- On se donne un ensemble d'indices

$$M \subset \{1, 2, \ldots, p\}, \tag{2.95}$$

avec

$$\mathrm{card} M = p - q, \tag{2.96}$$

et on décide de « tuer » les ondes correspondantes à $\{\lambda_m\}_{m\in M}$. Par cette expression, on entend qu'à traversée de l'onde λ_m dans le problème de Riemann $x = 0$, on décide que le saut d'amplitude est nul. Dans le cas linéaire, cela revient à choisir les composantes sortantes $L^M\mathbb{U}_b$ égales à leurs homologues intérieures.

- L'ensemble complémentaire

$$I = \{1, 2, \ldots, p\} \setminus M, \tag{2.97}$$

a pour cardinal q. Les champs $\{\lambda_i\}_{i\in I}$ ne sont pas nécessairement tous entrants, mais on décide que ce sont eux qui porteront les q informations qu'on souhaite imposer. Bien entendu, on doit vérifier la condition d'inversibilité

$$\det \nabla_{\mathbb{U}}\Phi(\mathbb{U}) R^I(\mathbb{U}) \neq 0, \tag{2.98}$$

et cela implique que tous les choix de M (donc de I) ne sont pas acceptables.

2.4.2 Application au système de relaxation diphasique

Dans les simulations TACITE, on souhaite imposer aux limites certaines grandeurs physiques en fonction du temps. À l'amont ($x = 0$) on veut imposer le débit du gaz et le débit total, ce qui implique les flux entrant dans le domaine. À l'aval ($x = L$) on veut imposer la pression et une condition de non-retour du liquide. Cela fait, à chaque extremité, deux informations ($q = 2$) pour cinq inconnues ($p = 5$).

Fig. 2.2. Les ondes dans le cas du schéma de relaxation en eulérien - Structure d'ondes.

Nous rappelons que pour le modèle de relaxation, nous avons cinq ondes $\lambda_1 = v - a\tau, \lambda_2 = v - b\tau, \lambda_3 = v, \lambda_4 = v + b\tau$ et $\lambda_5 = v + a\tau$. Le coefficient a est très grand, on a donc naturellement $v - a\tau < 0$ et $v + a\tau > 0$. En revanche, on ne connaît pas les signes de $v - b\tau, v$ et $v + b\tau$.

2.4.2.1 À l'amont

À l'amont, puisque l'onde $v + a\tau$ est une onde entrante ($v + a\tau > 0$) il est naturel que cette onde porte une information que l'on impose à $x = 0$, à savoir le débit total du mélange. Comme nous avons deux conditions aux limites à imposer à cet endroit, il faut donc choisir une autre onde qui apporte la deuxième information (sur le débit du gaz) et « tuer » celles qui restent. Le choix est critique car il faut bien les choisir de manière à satisfaire la condition de Kreiss (2.98). Examinons deux cas de traitement des conditions aux limites à l'amont.

1. Le premier cas correspond à un « bon » choix des ondes à tuer et à garder. Prenons les ensembles M et I comme suivant

$$\begin{aligned} M &= \{1, 2, 3\}, \\ I &= \{4, 5\}. \end{aligned} \tag{2.99}$$

Fig. 2.3. À l'amont, supprimer les ondes représentées par les lignes en pointillé.

Nous allons alors enlever les trois ondes à gauche et garder les deux ondes à droite à l'interface $x = 0$.
En variables conservatives $\mathbb{W} = (\rho, \rho v, \rho\pi, \rho Y, \rho\Sigma)^T$, les deux conditions imposées à l'amont s'écrivent

$$\Phi^{Amont}(\mathbb{W}) = \begin{pmatrix} \rho v - \bar{q}_T \\ \rho Y v - \Sigma - \bar{q}_G \end{pmatrix} = 0, \tag{2.100}$$

où $\bar{q}_G$ et $\bar{q}_T$ sont respectivement le débit du gaz et le débit total que l'on impose à l'amont. La dérivée de Φ^{Amont} et la matrice des vecteurs propres à droites associés aux ondes $\lambda_4 = v + b\tau$ et $\lambda_5 = v + a\tau$ sont données par

$$\nabla_{\mathbb{W}}\Phi^{Amont}(\mathbb{W}) = \begin{bmatrix} 0 & 1 & 0 & 0 & 0 \\ -Yv & Y & 0 & v & 0 \end{bmatrix} \quad \text{et} \quad R^I(\mathbb{W}) = \begin{bmatrix} 0 & 1 \\ 0 & v + a\tau \\ 0 & \Pi + a\tau^2 \\ 1 & Y \\ -b & \Sigma \end{bmatrix}. \tag{2.101}$$

Le déterminant à tester, donné par

$$\det\left[\nabla_{\mathbb{W}}\Phi^{Amont}(\mathbb{W})R^I(\mathbb{W})\right] = \det\begin{bmatrix} 0 & v + a\tau \\ v & Y(v + a\tau) \end{bmatrix} = -v(v + a\tau) \tag{2.102}$$

est non nul parce que $v \neq 0$ et $v \neq -a\tau$ (a étant très grand devant $\rho|v|$). La condition de Kreiss est ainsi vérifiée.

2. Le deuxième cas correspond à un « mauvais » choix des ondes pour apporter les informations à l'amont. Considérons les ensembles

$$\begin{aligned} M &= \{1, 2, 4\}, \\ I &= \{3, 5\}. \end{aligned} \tag{2.103}$$

Alors on va supprimer les ondes $\lambda_1 = v - a\tau$, $\lambda_2 = v - b\tau$ et $\lambda_4 = v + b\tau$. La matrice des vecteurs propres à droite $R^I(\mathbb{W})$ associés aux ondes à garder $\lambda_3 = v$ et $\lambda_5 = v + a\tau$ ainsi que le produit

Fig. 2.4. À l'amont, supprimer les mauvaises ondes représentées par les lignes en pointillé.

matriciel $\nabla_{\mathbb{W}}\Phi^{Amont}(\mathbb{W})R^I(\mathbb{W})$ sont donnés par

$$R^I(\mathbb{W}) = \begin{bmatrix} 1 & 1 \\ v & v+a\tau \\ \Pi & \Pi + a\tau^2 \\ Y & Y \\ \Sigma & \Sigma \end{bmatrix} \quad \text{et} \quad \nabla_{\mathbb{W}}\Phi^{Amont}(\mathbb{W})R^I(\mathbb{W}) = \begin{bmatrix} v & v+a\tau \\ Yv & Y(v+a\tau) \end{bmatrix}. \tag{2.104}$$

On voit bien que le déterminant de $\nabla_{\mathbb{W}}\Phi^{Amont}(\mathbb{W})R^I(\mathbb{W})$ est nul pour tout état $\mathbb{W} \in \Omega$. La condition de Kreiss n'est pas satisfaite. Par conséquent, on ne peut pas utiliser cette méthode de suppression d'ondes.

2.4.2.2 À l'aval

Contrairement à ce que l'on a vu à l'amont, les ondes entrantes dans le domaine à l'aval ($x = L$) sont celles dont la vitesse est négative, et les ondes sortantes sont celles dont la vitesse est positive. Puisque a est très grand, il est donc naturel que l'onde $\lambda_1 = v - a\tau$ est une onde entrante et on la garde comme une onde physique qui porte une information sur les conditions à imposer. Nous devons alors trouver une deuxième onde à garder pour remplir les conditions aux limites. Présentons deux choix différents.

1. Le premier choix consiste à prendre

$$\begin{aligned} M &= \{3,4,5\}, \\ I &= \{1,2\}. \end{aligned} \tag{2.105}$$

Nous allons voir qu'il s'agit d'un bon choix pour respecter la conditions d'inversibilité de Kreiss.

Fig. 2.5. À l'aval, supprimer les ondes représentées par les lignes en pointillé.

À l'aval, les deux conditions aux limites imposées s'écrivent

$$\Phi^{Aval}(\mathbb{W}) = \begin{pmatrix} Y - \bar{Y} \\ \Pi - \bar{p} + \Sigma\phi(\bar{p}, \bar{Y}, v) \end{pmatrix} = 0, \tag{2.106}$$

où $\bar{p}$ et $\bar{Y}$ sont respectivement la pression et la fraction massique imposées à l'aval. La fonction ϕ est le glissement ou l'écart entre les vitesses de chaque phase, défini par (1.7). La dérivée de cette condition aux limites par rapport à la variable conservative $\mathbb{W}$ est

$$\nabla_{\mathbb{W}}\Phi^{Aval}(\mathbb{W}) = \begin{bmatrix} -\tau Y & 0 & 0 & \tau & 0 \\ \Phi_{(\rho)} & \Phi_{(\rho v)} & \Phi_{(\rho\Pi)} & \Phi_{(\rho Y)} & \Phi_{(\rho\Sigma)} \end{bmatrix}, \tag{2.107}$$

avec

$$\begin{aligned}
\Phi_{(\rho)} &= -\tau\Pi + \tau\Sigma\phi(\bar{\rho}, \rho\bar{Y}, \bar{\rho}v) + \Sigma\tau v\bar{\rho}\phi_{(\rho v)}, \\
\Phi_{(\rho v)} &= -\Sigma\frac{\bar{\rho}}{\rho}\phi_{(\rho v)}, \\
\Phi_{(\rho\Pi)} &= \tau, \\
\Phi_{(\rho Y)} &= 0, \\
\Phi_{(\rho\Sigma)} &= -\tau\phi(\bar{p}, \bar{Y}, v_{N+1}).
\end{aligned} \quad . \tag{2.108}$$

Dans (2.108), $\bar{\rho}$ est la densité du mélange calculée à partir de la pression et de la fraction massique du gaz imposées, *i.e.* $\bar{\rho} = \rho(\bar{Y}, \bar{p})$. La matrice des vecteurs propres à droites correspondant aux deux ondes « entrantes » et le produit matriciel $\nabla_{\mathbb{W}}\Phi^{Aval}(\mathbb{W})R^I(\mathbb{W})$ sont donnés par

$$R^I(\mathbb{W}) = \begin{bmatrix} 1 & 0 \\ v - a\tau & 0 \\ \Pi + a\tau^2 & 0 \\ Y & 1 \\ \Sigma & b \end{bmatrix} \quad \text{et} \quad \nabla_{\mathbb{W}}\Phi^{Aval}(\mathbb{W})R^I(\mathbb{W}) = \begin{bmatrix} a^2\tau^2 + a\tau^2\Sigma\bar{\rho}\phi_{(\rho v)} & * \\ 0 & \tau \end{bmatrix}, \tag{2.109}$$

où le terme $*$ est un terme qui nous intéresse pas. Le déterminant à tester est $a^2\tau^3 + a\tau^3\Sigma\bar{\rho}\phi_{(\rho v)}$. Il est non nul dès que l'on choisit a suffisamment grand ce qui est le cas. La condition de Kreiss est alors vérifiée.

2. Regardons un deuxième cas où on choisit des mauvaises ondes à garder pour satisfaire les conditions aux limites. Nous supprimons maintenant les trois ondes $\lambda_2 = v - b\tau, \lambda_4 = v + b\tau$ et $\lambda_5 = v + a\tau$. Alors les ensembles M et I sont

$$\begin{aligned}
M &= \{2, 4, 5\}, \\
I &= \{1, 3\}.
\end{aligned} \tag{2.110}$$

La matrice des vecteurs propres à droite des ondes gardées ainsi que le produit entre cette matrice et la jacobienne des conditions aux limites sont

$$R^I(\mathbb{W}) = \begin{bmatrix} 1 & 1 \\ v - a\tau & v \\ \Pi + a\tau^2 & \Pi \\ Y & Y \\ \Sigma & \Sigma \end{bmatrix} \quad \text{et} \quad \nabla_{\mathbb{W}}\Phi^{Aval}(\mathbb{W})R^I(\mathbb{W}) = \begin{bmatrix} 0 & 0 \\ * & * \end{bmatrix}. \tag{2.111}$$

Pour tout $\mathbb{W} \in \Omega$, la matrice $\nabla_{\mathbb{W}}\Phi^{Aval}(\mathbb{W})R^I(\mathbb{W})$ est singulière à cause de son déterminant nul. La condition de Kreiss n'est pas vérifiée. C'est donc un mauvais choix pour supprimer les ondes à l'aval.

***Remarque** 2.5.* Les choix pour annihiler les bonnes ondes à l'amont (2.99) et à l'aval (2.105) correspondent à la méthode préconisée par F. Dubois [45]. En effet, d'après F. Dubois, il faut tuer les ondes en partant de celle qui sépare l'état fictif du premier état intermédiaire du problème de Riemann, puis

Fig. 2.6. À l'aval, supprimer les mauvaises ondes représentées par les lignes en pointillé.

de les prendre dans l'ordre (et non arbitrairement), la dernière étant celle séparant le dernier état intermédiaire du premier état intérieur du domaine. Dans ce cas, nous devons supprimer trois ondes de chaque côté dans l'ordre indiqué.

2.4.3 Conditions aux limites au niveau discret

Passons maintenant au traitement des conditions aux limites pour le système de relaxation diphasique au niveau discret. Notons $\mathbb{W}_0 = (\rho, \rho v, \rho\Pi, \rho Y, \rho\Sigma)_0$ l'état fictif amont et $\mathbb{W}_{N+1} = (\rho, \rho v, \rho\Pi, \rho Y, \rho\Sigma)_{N+1}$ l'état fictif aval.

Fig. 2.7. Les invariants de Riemann forts du problème de Riemann pour le système de relaxation (3.2).

Pour chaque état fictif constitué de cinq composantes, il faut cinq équations pour le déterminer complètement. Pour cela, nous avons vu qu'il faut « tuer » certaines ondes à l'amont et à l'aval, c'est-à-dire annuler leur amplitude. Cela revient à rendre égaux les invariants forts de Riemann associés à ces ondes entre l'état fictif et le premier état intérieur. Dans la suite, nous allons montrer les équations correspondant à cette méthode de suppression d'ondes, ainsi que les conditions aux limites à imposer. Les calculs précis des états fictifs sont détaillés dans le chapitre suivant où nous présentons le schéma de relaxation en Lagrange-Projection.

2.4.3.1 À l'amont

Nous avons vu qu'à l'amont, tuer les ondes λ_1, λ_2 et λ_3 est un bon choix. Dans ce cas, nous avons trois relations à écrire. Les deux conditions aux limites imposées permettent d'écrire deux autres équations ce qui complète le système pour calculer entièrement l'état fictif $\mathbb{W}_0$.

- Pour l'ensemble $M = \{1, 2, 3\}$

$$\begin{aligned} \Pi_0 - av_0 &= \Pi_1 - av_1, \\ \Sigma_0 + bY_0 &= \Sigma_1 + bY_1, \\ \Pi_0 + a^2\tau_0 &= \Pi_1 + a^2\tau_1. \end{aligned} \tag{2.112}$$

- Pour l'ensemble $S = \{4, 5\}$

$$\begin{aligned} \rho_0 v_0 &= \bar{q}_T, \\ Y_0 q_T - \Sigma_0 &= \bar{q}_G. \end{aligned} \tag{2.113}$$

L'état $\mathbb{W}_1$ étant connu, nous avons système composé de cinq équations avec cinq inconues qui peut être résolu facilement.

2.4.3.2 À l'aval

À l'aval, comme déjà vu, nous devons tuer les ondes λ_3, λ_4 et λ_5 et garder les ondes λ_1, λ_2. Ici, nous imposons la pression p et la fraction massique du gaz. Les cinq équations à écire pour déterminer l'état fictif $\mathbb{W}_{N+1}$ sont

- Pour l'ensemble $M = \{3, 4, 5\}$

$$\begin{aligned} \Pi_{N+1} + av_{N+1} &= \Pi_N + av_N, \\ \Sigma_{N+1} - bY_{N+1} &= \Sigma_N - bY_N, \\ \Pi_{N+1} + a^2\tau_{N+1} &= \Pi_N + a^2\tau_N. \end{aligned} \tag{2.114}$$

- Pour l'ensemble $S = \{1, 2, \}$

$$\begin{aligned} \Pi_{N+1} &= \bar{p} + \rho_{N+1} Y_{N+1}(1 - Y_{N+1})\phi^2(\bar{p}, Y_{N+1}, v_{N+1}), \\ Y_{N+1} &= Y_N. \end{aligned} \tag{2.115}$$

Dans (2.115), ϕ est le glissement calculé à la pression imposée.

Chapitre 3

Méthode de relaxation en Lagrange-Projection

Ce chapitre, à partir duquel nous abordons les contributions propres à la thèse, est consacré à la première nouveauté que nous aimerions mettre en avant, à savoir la méthode Lagrange-Projection en utilisation conjointe avec la technique de relaxation. La présentation que nous allons donner correspond au modèle complet avec glissement quelconque.

La méthode que nous préconisons possède plusieurs atouts. Tout d'abord, le formalisme ALE (Arbitrary Lagrangian-Eulérien) [44, 59] entraîne une décomposition en deux phases de la résolution. Cette décomposition fait apparaître une séparation fort naturelle entre les ondes rapides et les ondes lentes, que l'on peut exploiter pour mettre au point un nouveau schéma semi-implicite.

Le deuxième avantage de la méthode Lagrange-Projection est la positivité du nouveau schéma semi-implicite ainsi construit. Rappelons que dans le chapitre précédent, nous avons prouvé qu'en eulérien le schéma explicite peut garantir la positivité de la densité et le principe du maximum de la fraction massique du gaz. En revanche, rien n'est garanti pour le schéma implicite. Ici l'utilisation de la méthode Lagrange-Projection nous permet d'obtenir un schéma volumes finis conservatif qui préserve ces propriétés physiques exigées par nos applications. Cette garantie est soumise, en explicite comme en implicite au premier ordre, à une certaine condition CFL. Concrètement, exiger la positivité du schéma Lagrange-Projection conduit à établir une condition CFL explicitement calculable permettant de choisir le pas de temps. En explicite, cette condition est facilement satisfaite car toutes les valeurs sont connues. En implicite, cela est plus difficile car il faut faire une estimation a priori pour imposer un pas de temps adéquat. Bien évidemment, cette estimation prend en compte non seulement la contribution des données à l'intérieur du domaine mais aussi les conditions aux limites imposées. À notre connaissance, ce type d'estimation n'existe pas à l'heure actuelle dans la théorie de l'hyperbolique.

Deux autres avantages du schéma Lagrange-Projection sont la rapidité et la robustesse du schéma semi-implicite. En effet, au lieu de travailler avec un système linéaire par blocs de taille 5×5 comme en eulérien, notre système à résoudre est constitué simplement des matrices de taille 2×2 ce qui réduit considérablement le temps CPU. D'autre part, nous allons voir que ce nouveau schéma est capable de passer des cas-tests réalistes d'écoulements diphasiques.

Comme pour l'ancienne méthode de relaxation, nous allons présenter la nouvelle méthode tout d'abord au niveau continu, ensuite en discret explicite, puis en semi-implicite. Centrale à ce dernier cas est l'estimation a priori du pas de temps. Nous accordons aussi une importance particulière au traitement des conditions aux limites, du terme source ainsi que la montée en ordre du schéma. Le chapitre sera complété par l'illustration de quelques résultats numériques et suivi d'une conclusion.

3.1 Méthode de relaxation-Lagrange-Projection au niveau continu

Nous avons vu au chapitre précédent que le système DFM de départ

$$\begin{aligned} \partial_t(\rho) \quad &+ \partial_x(\rho v) &&= 0, \\ \partial_t(\rho Y) &+ \partial_x(\rho Y v - \sigma(\mathbb{U})) &&= 0, \\ \partial_t(\rho v) \quad &+ \partial_x(\rho v^2 + P(\mathbb{U})) &&= 0, \end{aligned} \tag{3.1}$$

peut être approché par le système de relaxation

$$\begin{aligned} \partial_t(\rho) \quad &+ \partial_x(\rho v) &&= 0, \\ \partial_t(\rho v) \quad &+ \partial_x(\rho v^2 + \Pi) &&= 0, \\ \partial_t(\rho \Pi) &+ \partial_x(\rho \Pi v + a^2 v) &&= \eta\rho(P(\mathbb{U}) - \Pi), \\ \partial_t(\rho Y) &+ \partial_x(\rho Y v - \Sigma) &&= 0, \\ \partial_t(\rho \Sigma) &+ \partial_x(\rho \Sigma v - b^2 Y) &&= \eta\rho(\sigma(\mathbb{U}) - \Sigma). \end{aligned} \tag{3.2}$$

Au lieu de discrétiser directement (3.2) comme avant, on va réécrire ce système dans un autre référentiel afin de mieux mettre en relief sa structure.

3.1.1 Principes de la décomposition ALE

Désignons par χ la coordonnée d'un référentiel mobile qui n'est ni la coordonnée eulérienne habituelle x (grille spatiale) ni la coordonnée lagrangienne ou matérielle X (grille en masse), u est la vitesse de déplacement imposée au repère mobile par rapport au laboratoire. Dans ce cas dans le repère mobile ALE, la vitesse des particules qui se déplacent à une vitesse v par rapport au laboratoire est $v - u$. La correspondance entre le laboratoire et le repère mobile nous donne $\partial_t x_{|\chi} = v - u$. La Fig. 3.1 montre la relation entre ces trois repères.

Fig. 3.1. Trois repères.

Notons $J = \partial_\chi x_{|t}$ le taux de dilatation. Après quelques changements de variables [44, 59] on parvient à un nouveau système équivalent à (3.2)

$$\begin{array}{llllll}
\partial_t(J) & + \partial_\chi(u) & - & \partial_\chi(v) & = 0, \\
\partial_t(\rho J) & + \partial_\chi(\rho u) & & & = 0, \\
\partial_t(\rho v J) & + \partial_\chi(\rho v u) & + & \partial_\chi(\Pi) & = 0, \\
\partial_t(\rho \Pi J) & + \partial_\chi(\rho \Pi u) & + & \partial_\chi(a^2 v) & = \eta \rho J(P - \Pi), \\
\partial_t(\rho Y J) & + \partial_\chi(\rho Y u) & - & \partial_\chi(\Sigma) & = 0, \\
\partial_t(\rho \Sigma J) & + \underbrace{\partial_\chi(\rho \Sigma u)}_{\textbf{Convective}} & - & \underbrace{\partial_\chi(b^2 Y)}_{\textbf{Lagrangienne}} & = \eta \rho J(\sigma - \Sigma).
\end{array} \tag{3.3}$$

L'idée majeure de la méthode ALE est de décomposer le système précédent en deux parties : une partie lagrangienne et une partie transport. À chaque itération en temps, nous avons à résoudre

1. Une phase lagrangienne

$$\begin{array}{llll}
\partial_t(J) & - & \partial_\chi(v) & = 0, \quad (3.4a) \\
\partial_t(\rho J) & & & = 0, \quad (3.4b) \\
\partial_t(\rho v J) & + & \partial_\chi(\Pi) & = 0, \quad (3.4c) \\
\partial_t(\rho \Pi J) & + a^2 & \partial_\chi(v) & = \eta \rho J(P - \Pi), \quad (3.4d) \\
\partial_t(\rho Y J) & - & \partial_\chi(\Sigma) & = 0, \quad (3.4e) \\
\partial_t(\rho \Sigma J) & - b^2 & \partial_\chi(Y) & = \eta \rho J(\sigma - \Sigma). \quad (3.4f)
\end{array}$$

   Dans cette phase, on constate que pour $\eta = 0$ le système (3.4) se décompose en deux sous systèmes : le premier associé aux ondes rapides

$$\begin{array}{llll}
\partial_t(J) & - & \partial_\chi(v) & = 0, \\
\partial_t(\rho J) & & & = 0, \\
\partial_t(\rho v J) & + & \partial_\chi(\Pi) & = 0, \\
\partial_t(\rho \Pi J) & + a^2 & \partial_\chi(v) & = 0,
\end{array} \tag{3.5}$$

   et le deuxième associé aux ondes lentes

$$\begin{array}{llll}
\partial_t(\rho Y J) & - & \partial_\chi(\Sigma) & = 0, \\
\partial_t(\rho \Sigma J) & - b^2 & \partial_\chi(Y) & = 0.
\end{array} \tag{3.6}$$

2. Une phase convective, phase durant laquelle on ne tient compte que du déplacement de matière de vitesse u

$$\begin{array}{llll}
\partial_t(J) & + \partial_\chi(u) & = 0, \quad (3.7a) \\
\partial_t(\rho J) & + \partial_\chi(\rho u) & = 0, \quad (3.7b) \\
\partial_t(\rho v J) & + \partial_\chi(\rho v u) & = 0, \quad (3.7c) \\
\partial_t(\rho \Pi J) & + \partial_\chi(\rho \Pi u) & = 0, \quad (3.7d) \\
\partial_t(\rho Y J) & + \partial_\chi(\rho Y u) & = 0, \quad (3.7e) \\
\partial_t(\rho \Sigma J) & + \partial_\chi(\rho \Sigma u) & = 0. \quad (3.7f)
\end{array}$$

En écrivant les cinq dernières équations de (3.7) sous forme abstraite, la phase convective s'écrit

$$\begin{aligned} \partial_t(J) \quad &+ \partial_\chi(u) \quad = 0, \\ \partial_t(\mathbb{W}J) &+ \partial_\chi(\mathbb{W}u) = 0, \end{aligned} \tag{3.8}$$

d'où

$$\partial_t(\mathbb{W}) + \frac{u}{J}\partial_\chi(\mathbb{W}) = 0. \tag{3.9}$$

Ceci montre que la deuxième phase correspond en réalité à un transport à la vitesse $\vartheta = \dfrac{u}{J}$. C'est pourquoi on l'appelle aussi phase de « remaillage ».

3.1.2 Méthode de Lagrange-Projection

Historiquement, la décomposition ALE a été introduite [59] dans les années soixante pour traiter les problèmes à frontière mobile. Ici, elle sert à « séparer » les ondes en maillage fixe, ce qui implique qu'on n'aurait a priori besoin que d'un choix particulier pour u.

À l'issue des deux phases lagrangienne et convective on souhaite retomber sur le repère eulérien. Cela revient à faire coïncider χ et x après chaque itération en temps, c'est-à-dire

$$J^n = 1 \quad \forall n \in \mathbb{N}, \tag{3.10}$$

$$(J^n = 1, \mathbb{W}^n) \overset{\textbf{Lagrange}}{\longrightarrow} (\widetilde{J}^n, \widetilde{\mathbb{W}}^n) \overset{\textbf{Projection}}{\longrightarrow} (J^{n+1} = 1, \mathbb{W}^{n+1}). \tag{3.11}$$

Afin de réaliser (3.11) nous voyons à partir de (3.4a) et (3.7a) qu'il faut prendre $u = v$. Au niveau discret, nous verrons ce qu'il convient à faire. La méthode Lagrange-Projection se décompose en deux phases

1. La phase lagrangienne où la masse corrigée $m = \rho J$ est constante d'après (3.4b), ce qui revient à résoudre

$$\begin{aligned} m\partial_t(\tau) \; &- \partial_\chi(v) \quad = 0, \\ m\partial_t(v) \; &+ \partial_\chi(\Pi) \quad = 0, \\ m\partial_t(\Pi) &+ \partial_\chi(a^2 v) \; = \eta m(P - \Pi), \\ m\partial_t(Y) \; &- \partial_\chi(\Sigma) \quad = 0, \\ m\partial_t(\Sigma) \; &- \partial_\chi(b^2 Y) = \eta m(\sigma - \Sigma). \end{aligned} \tag{3.12}$$

   Ce système est muni d'une condition initiale au début de chaque pas de temps t^n

$$(m^n, \mathbb{V}^n) = (\rho^n J^n, \mathbb{V}^n) = (\rho^n, \mathbb{V}^n), \quad \mathbb{V} = (\tau, Y, v, \Pi, \Sigma)^T, \tag{3.13}$$

   et des conditions aux limites de type

$$\begin{aligned} (\rho v)^n(\chi = 0) &= q_T(t^n), \qquad \Pi^n(\chi = \mathcal{L}) = p(t^n), \\ (\rho Y v)^n(\chi = 0) &= q_G(t^n), \qquad Y^n(\chi = \mathcal{L}) = Y(t^n), \end{aligned} \tag{3.14}$$

   où $\mathcal{L}$ désigne la sortie de la conduite dans la coordonnée lagrangienne.

Dans cette phase, étant fixée à l'instant t^n la masse $m^n = \rho^n J^n = \rho^n$ et $\chi = x$, le système à résoudre est donc

$$\begin{aligned}
\rho^n \partial_t(\tau) \; - \partial_x(v) \quad &= 0, \\
\rho^n \partial_t(v) \; + \partial_x(\Pi) \quad &= 0, \\
\rho^n \partial_t(\Pi) + \partial_x(a^2 v) \; &= \eta \rho^n (P - \Pi), \\
\rho^n \partial_t(Y) \; - \partial_x(\Sigma) \quad &= 0, \\
\rho^n \partial_t(\Sigma) \; - \partial_x(b^2 Y) \; &= \eta \rho^n (\sigma - \Sigma).
\end{aligned} \tag{3.15}$$

2. La phase projection représentée par un système composé de cinq équations de transport où la vitesse de transport ϑ est bien choisie. On peut écrire ce système sous la forme condensée suivante

$$\partial_t(\mathbb{W}) + \vartheta \partial_x(\mathbb{W}) = 0. \tag{3.16}$$

Ce système est muni de la donnée initiale $\widetilde{\mathbb{W}^n}$ (solution de la phase lagrangienne) et des conditions aux limites en 0 et en L (en coordonnée eulérienne). Nous allons voir plus tard au niveau discret que ces conditions aux limites nécessitent un traitement particulier afin d'avoir un bon flux aux bords du domaine.

3.2 Méthode de relaxation-Lagrange-Projection en explicite premier ordre

En préparation au schéma semi-implicite, nous présentons dans cette section le schéma Lagrange-Projection explicite en temps et au premier ordre. Cela nous aidera à mieux comprendre le mécanisme du schéma Lagrange-Projection, à savoir la condition de positivité et la forme de flux conservative du schéma, le traitement des conditions aux limites ainsi que la montée en ordre du schéma.

Notons tout d'abord que tout comme le schéma eulérien explicite, en Lagrange-Projection explicite les coefficients de relaxation a et b sont aussi des valeurs locales afin de diminuer la diffusion du schéma. Ils sont donc calculés localement aux interfaces du problème de Riemann associé. Rappelons les conditions dites de Whitham que ces coefficients doivent respecter pour la stabilité lors de l'utilisation de la méthode de relaxation :

$$a_{i+1/2} = a^\flat_{i+1/2}(\mathbb{W}_i, \mathbb{W}_{i+1}) \quad \text{et} \quad b_{i+1/2} = b^\flat_{i+1/2}(\mathbb{W}_i, \mathbb{W}_{i+1}), \tag{3.17}$$

avec

$$a^\flat_{i+1/2}(\mathbb{W}_i, \mathbb{W}_{i+1}) = \max\left(\sqrt{-\partial_\tau P(\mathbb{W}_i) + [\partial_v P(\mathbb{W}_i)]^2}, \sqrt{-\partial_\tau P(\mathbb{W}_{i+1}) + [\partial_v P(\mathbb{W}_{i+1})]^2}\right), \tag{3.18}$$

et

$$b^\flat_{i+1/2}(\mathbb{W}_i, \mathbb{W}_{i+1}) = \max\left(|\partial_Y \sigma(\mathbb{W}_i)|, |\partial_Y \sigma(\mathbb{W}_{i+1})|\right). \tag{3.19}$$

Ici, contrairement au schéma eulérien explicite, les conditions sur a et b vis-à-vis des bornes $a^\sharp_{i+1/2}$ et $b^\sharp_{i+1/2}$ (définies dans (2.41)) ne sont pas nécessaires pour assurer la positivité de la densité et la fraction massique de gaz. Nous allons voir qu'il est possible d'établir une CFL explicite calée sur le pas de temps pour garantir ces propriétés. C'est pour cela que seules les conditions de Whitham (3.18) et (3.19) imposées sur a et b sont requises à ce stade.

3.2.1 Formule de mise à jour

Le schéma explicite est obtenu facilement en discrétisant les cinq équations du système (3.15) pour obtenir la solution $\widetilde{\mathbb{W}}$ de la phase Lagrange, puis les cinq équations découplées du système (3.16) pour obtenir la solution $\widehat{\mathbb{W}}$ de la phase Projection.

Pour la phase Lagrange, le système de discrétisation s'écrit, pour tout $i \in \mathbb{Z}$:

$$\begin{aligned}
\rho_i^n \frac{\widetilde{\tau}_i^n - \tau_i^n}{\Delta t} &- \frac{v_{i+1/2}^{n,*} - v_{i-1/2}^{n,*}}{\Delta x} = 0,\\
\rho_i^n \frac{\widetilde{v}_i^n - v_i^n}{\Delta t} &+ \frac{\Pi_{i+1/2}^{n,*} - \Pi_{i-1/2}^{n,*}}{\Delta x} = 0,\\
\rho_i^n \frac{\widetilde{\Pi}_i^n - \Pi_i^n}{\Delta t} &+ a^2 \frac{v_{i+1/2}^{n,*} - v_{i-1/2}^{n,*}}{\Delta x} = 0,\\
\rho_i^n \frac{\widetilde{Y}_i^n - Y_i^n}{\Delta t} &- \frac{\Sigma_{i+1/2}^{n,*} - \Sigma_{i-1/2}^{n,*}}{\Delta x} = 0,\\
\rho_i^n \frac{\widetilde{\Sigma}_i^n - \Sigma_i^n}{\Delta t} &- b^2 \frac{Y_{i+1/2}^{n,*} - Y_{i-1/2}^{n,*}}{\Delta x} = 0,
\end{aligned} \tag{3.20}$$

où $v_{i+1/2}^{n,*}$, $\Pi_{i+1/2}^{n,*}$ et $\Sigma_{i+1/2}^{n,*}$ sont les grandeurs intermédiaires qui interviennent dans le problème de Riemann à l'interface $i+1/2$

$$\begin{aligned}
v_{i+1/2}^{n,*} &= \frac{v_i^n + v_{i+1}^n}{2} - \frac{\Pi_{i+1}^n - \Pi_i^n}{2a_{i+1/2}^n},\\
\Pi_{i+1/2}^{n,*} &= \frac{\Pi_i^n + \Pi_{i+1}^n}{2} - a_{i+1/2}^n \frac{v_{i+1}^n - v_i^n}{2},\\
\Sigma_{i+1/2}^{n,*} &= \frac{\Sigma_i^n + \Sigma_{i+1}^n}{2} + b_{i+1/2}^n \frac{Y_{i+1}^n - Y_i^n}{2}.
\end{aligned} \tag{3.21}$$

Dans (3.21) les variables de relaxation Π_i^n et Σ_i^n sont définies à l'équilibre par

$$\Pi_i^n = P(\mathbb{U}_i^n), \qquad \text{et} \qquad \Sigma_i^n = \sigma(\mathbb{U}_i^n). \tag{3.22}$$

Notons que dans la pratique, les formules de mise à jour de $\widetilde{\Sigma}_i^n$ et de $\widetilde{\Pi}_i^n$ ne sont pas implémentées. En effet, la procédure de la relaxation exige que les variables de relaxation Σ et Π coïncident avec les variétés d'équilibre à la fin de chaque pas de temps comme indiqué dans (3.22). Par conséquent, désormais nous ne prenons pas en compte la discrétisation de Σ et Π mais imposons (3.22) au début de chaque pas de temps t^n.

On obtient donc la solution intermédiaire de la phase Lagrange

$$\begin{aligned}
\widetilde{\tau}_i^n &= \tau_i^n + \frac{\Delta t}{\rho_i^n \Delta x}(v_{i+1/2}^{n,*} - v_{i-1/2}^{n,*}),\\
\widetilde{Y}_i^n &= Y_i^n + \frac{\Delta t}{\rho_i^n \Delta x}(\Sigma_{i+1/2}^{n,*} - \Sigma_{i-1/2}^{n,*}),\\
\widetilde{v}_i^n &= v_i^n - \frac{\Delta t}{\rho_i^n \Delta x}(\Pi_{i+1/2}^{n,*} - \Pi_{i-1/2}^{n,*}).
\end{aligned} \tag{3.23}$$

Avant d'effectuer la phase Projection, il faut définir, comme mentionné dans la section précédente, une vitesse de transport ϑ convenable pour que l'on retombe sur la grille eulérienne à la fin de cette étape, c'est-à-dire $J_i^n = J_i^{n+1} = 1$ pour tout $i \in \mathbb{Z}$, $n \in \mathbb{N}$.

***Proposition** 3.1. Une condition suffisante pour retomber, après la phase Projection, sur la grille eulérienne est que*

$$u^{n,*}_{i+1/2} = v^{n,*}_{i+1/2}, \quad \forall i \in \mathbb{Z} \text{ et } n \in \mathbb{N}. \tag{3.24}$$

Démonstration En discrétisant la première équation de (3.4) et de (3.7) nous avons

$$\begin{aligned} &\text{phase Lagrange :} \qquad \widetilde{J}^n_i = J^n_i + \frac{\Delta t}{\Delta x}(v^{n,*}_{i+1/2} - v^{n,*}_{i-1/2}), \\ &\text{phase Projection : } J^{n+1}_i = \widetilde{J}^n_i - \frac{\Delta t}{\Delta x}(u^{n,*}_{i+1/2} - u^{n,*}_{i-1/2}), \end{aligned} \tag{3.25}$$

d'où

$$J^{n+1}_i = J^n_i + \frac{\Delta t}{\Delta x}(v^{n,*}_{i+1/2} - v^{n,*}_{i-1/2}) - \frac{\Delta t}{\Delta x}(u^{n,*}_{i+1/2} - u^{n,*}_{i-1/2}). \tag{3.26}$$

En conséquence, pour que $J^{n+1}_i = J^n_i = 1$ pour tout $i \in \mathbb{Z}$, $n \in \mathbb{N}$, il suffit que $u^{n,*}_{i+1/2}$ soit égale à $v^{n,*}_{i+1/2}$. □

La vitesse de transport ϑ dans la phase Projection est alors celle provenant du problème de Riemann à l'étape Lagrange :

$$\vartheta = v^{n,*}_{i+1/2}. \tag{3.27}$$

La solution de la phase Projection est obtenue en discrétisant toutes les équations de (3.16)

$$\begin{aligned}
(\rho_i)^{n+1} &= (\widetilde{\rho})^n_i - \frac{\Delta t}{\Delta x}\Big[(v^{n,*}_{i-1/2})^+(\widetilde{\rho}^n_i - \widetilde{\rho}^n_{i-1}) + (v^{n,*}_{i+1/2})^-(\widetilde{\rho}^n_{i+1} - \widetilde{\rho}^n_i)\Big], \\
(\rho v)^{n+1}_i &= (\widetilde{\rho v})^n_i - \frac{\Delta t}{\Delta x}\Big[(v^{n,*}_{i-1/2})^+((\widetilde{\rho v})^n_i - (\widetilde{\rho v})^n_{i-1}) + (v^{n,*}_{i+1/2})^-((\widetilde{\rho v})^n_{i+1} - (\widetilde{\rho v})^n_i)\Big], \\
(\rho \Pi)^{n+1}_i &= (\widetilde{\rho \Pi})^n_i - \frac{\Delta t}{\Delta x}\Big[(v^{n,*}_{i-1/2})^+((\widetilde{\rho \Pi})^n_i - (\widetilde{\rho \Pi})^n_{i-1}) + (v^{n,*}_{i+1/2})^-((\widetilde{\rho \Pi})^n_{i+1} - (\widetilde{\rho \Pi})^n_i)\Big], \\
(\rho Y)^{n+1}_i &= (\widetilde{\rho Y})^n_i - \frac{\Delta t}{\Delta x}\Big[(v^{n,*}_{i-1/2})^+((\widetilde{\rho Y})^n_i - (\widetilde{\rho Y})^n_{i-1}) + (v^{n,*}_{i+1/2})^-((\widetilde{\rho Y})^n_{i+1} - (\widetilde{\rho Y})^n_i)\Big], \\
(\rho \Sigma)^{n+1}_i &= (\widetilde{\rho \Sigma})^n_i - \frac{\Delta t}{\Delta x}\Big[(v^{n,*}_{i-1/2})^+((\widetilde{\rho \Sigma})^n_i - (\widetilde{\rho \Sigma})^n_{i-1}) + (v^{n,*}_{i+1/2})^-((\widetilde{\rho \Sigma})^n_{i+1} - (\widetilde{\rho \Sigma})^n_i)\Big],
\end{aligned} \tag{3.28}$$

avec $(x)^+ = \max(x, 0)$ et $(x)^- = \min(x, 0)$, $x \in \mathbb{R}$.

De la même façon que dans la phase Lagrange, nous ne nous sommes intéressés ici qu'à la discrétisation de ρ, Y et v puisque les variables de relaxation seront mises à l'équilibre à la fin de cette étape par

$$\Pi^{n+1}_i = P(\mathbb{U}^{n+1}_i), \qquad \text{et} \qquad \Sigma^{n+1}_i = \sigma(\mathbb{U}^{n+1}_i). \tag{3.29}$$

Le système de discrétisation pour la phase Projection peut donc s'écrire sous la forme générale

$$\mathbb{U}^{n+1}_i = \widetilde{\mathbb{U}}^n_i - \frac{\Delta t}{\Delta x}\left[(v^{n,*}_{i-1/2})^+(\widetilde{\mathbb{U}}^n_i - \widetilde{\mathbb{U}}^n_{i-1}) + (v^{n,*}_{i+1/2})^-(\widetilde{\mathbb{U}}^n_{i+1} - \widetilde{\mathbb{U}}^n_i)\right]. \tag{3.30}$$

3.2.2 Identité de Piola et forme conservative avec flux local

La première équation du système (3.23) nous permet de donner une relation entre $\widetilde{\tau}^n_i$ et τ^n_i. En l'occurrence, ce sont les variables physiques $\widetilde{\rho}^n_i$ et ρ^n_i qui nous intéressent pour la phase suivante. Nous cherchons donc une relation entre ces deux variables. On sait que

$$\rho^n_i(\widetilde{\tau}^n_i - \tau^n_i) = \frac{\Delta t}{\Delta x}(v^{n,*}_{i+1/2} - v^{n,*}_{i-1/2}), \tag{3.31}$$

à partir de quoi on obtient l'identité de Piola

$$\rho_i^n = \widetilde{\rho}_i^n \Big[1 + \frac{\Delta t}{\Delta x}(v_{i+1/2}^{n,*} - v_{i-1/2}^{n,*})\Big]. \tag{3.32}$$

Cette identité de Piola est autant utile pour vérifier la consistance (y compris en semi-implicite) que pour optimiser l'implémentation du code. En effet, nous rappelons que l'une des motivations de ce travail est d'utiliser la méthode de pas de temps local (voir plus tard dans chapter 5) il est donc important de vérifier que le schéma Lagrange-Projection proposé contient une notion de flux et que ce schéma soit conservatif. Pour cela, nous allons énoncer la proposition suivante.

***Proposition** 3.2. Le schéma volumes finis utilisant la décomposition à deux pas Lagrange-Projection est un schéma conservatif qui peut s'écrire sous forme*

$$\mathbb{U}_i^{n+1} = \mathbb{U}_i^n - \frac{\Delta t}{\Delta x}(\mathbb{F}_{i+1/2}^n - \mathbb{F}_{i-1/2}^n), \tag{3.33}$$

où le flux $\mathbb{F}_{i+1/2}^n$ est défini par

$$\mathbb{F}_{i+1/2}^n = (v_{i+1/2}^{n,*})^+ \widetilde{\mathbb{U}}_i^n + (v_{i+1/2}^{n,*})^- \widetilde{\mathbb{U}}_{i+1}^n + \mathbb{H}_{i+1/2}^{n,*}, \tag{3.34}$$

avec $\mathbb{H}_{i+1/2}^{n,*} = (0, -\Sigma_{i+1/2}^{n,*}, \Pi_{i+1/2}^{n,*})^T$.

Démonstration Pour démontrer cette proposition, nous allons exprimer l'état $\mathbb{U}^{n+1}$ à l'instant t^{n+1} en fonction de l'état $\mathbb{U}^n$ à l'instant t^n en proposant une expression explicite pour le flux. Pour cela, nous devons combiner les deux phases Lagrange et Projection en passant par l'identité de Piola.

Tout d'abord on introduit la variable $\mathbb{X} = (1, Y, v)^T$. Alors $\mathbb{U} = \rho\mathbb{X}$ et (3.30) se réécrit

$$(\rho\mathbb{X})_i^{n+1} = (\widetilde{\rho\mathbb{X}})_i^n - \frac{\Delta t}{\Delta x}\Big[(v_{i-1/2}^{n,*})^+((\widetilde{\rho\mathbb{X}})_i^n - (\widetilde{\rho\mathbb{X}})_{i-1}^n) + (v_{i+1/2}^{n,*})^-((\widetilde{\rho\mathbb{X}})_{i+1}^n - (\widetilde{\rho\mathbb{X}})_i^n)\Big]. \tag{3.35}$$

En multipliant l'identité de Piola (3.32) par $\widetilde{\mathbb{X}}_i^n$, on obtient

$$\begin{aligned}(\widetilde{\rho\mathbb{X}})_i^n &= \widetilde{\rho}_i^n \widetilde{\mathbb{X}}_i^n \\ &= \rho_i^n \widetilde{\mathbb{X}}_i^n - \widetilde{\rho}_i^n \widetilde{\mathbb{X}}_i^n \frac{\Delta t}{\Delta x}(v_{i+1/2}^{n,*} - v_{i-1/2}^{n,*}).\end{aligned} \tag{3.36}$$

En remplaçant cette relation (3.36) dans (3.35), on arrive à

$$\begin{aligned}(\rho\mathbb{X})_i^{n+1} = \rho_i^n \widetilde{\mathbb{X}}_i^n &- \frac{\Delta t}{\Delta x}\Big[(v_{i+1/2}^{n,*})^+(\widetilde{\rho\mathbb{X}})_i^n \quad + (v_{i+1/2}^{n,*})^-(\widetilde{\rho\mathbb{X}})_{i+1}^n\Big] \\ &+ \frac{\Delta t}{\Delta x}\Big[(v_{i-1/2}^{n,*})^+(\widetilde{\rho\mathbb{X}})_{i-1}^n + (v_{i-1/2}^{n,*})^-(\widetilde{\rho\mathbb{X}})_i^n\Big].\end{aligned} \tag{3.37}$$

De plus les deux dernières équations de (3.23) nous donnent

$$\rho_i^n(\widetilde{\mathbb{X}}_i^n - \mathbb{X}_i^n) = -\frac{\Delta t}{\Delta x}(\mathbb{H}_{i+1/2}^{n,*} - \mathbb{H}_{i-1/2}^{n,*}) \tag{3.38}$$

$$\Longrightarrow \rho_i^n \widetilde{\mathbb{X}}_i^n = \rho_i^n \mathbb{X}_i^n - \frac{\Delta t}{\Delta x}(\mathbb{H}_{i+1/2}^{n,*} - \mathbb{H}_{i-1/2}^{n,*}). \tag{3.39}$$

On remplace (3.39) dans (3.37) et on a

$$\begin{aligned}(\rho\mathbb{X})_i^{n+1} &= (\rho\mathbb{X})_i^n - \frac{\Delta t}{\Delta x}\left[(v_{i+1/2}^{n,*})^+(\widetilde{\rho\mathbb{X}})_i^n + (v_{i+1/2}^{n,*})^-(\widetilde{\rho\mathbb{X}})_{i+1}^n + \mathbb{H}_{i+1/2}^{n,*}\right] \\ &\quad + \frac{\Delta t}{\Delta x}\left[(v_{i-1/2}^{n,*})^+(\widetilde{\rho\mathbb{X}})_{i-1}^n + (v_{i-1/2}^{n,*})^-(\widetilde{\rho\mathbb{X}})_i^n + \mathbb{H}_{i-1/2}^{n,*}\right] \\ \Longleftrightarrow \qquad \mathbb{U}_i^{n+1} &= \mathbb{U}_i^n - \frac{\Delta t}{\Delta x}\left[(v_{i+1/2}^{n,*})^+\widetilde{\mathbb{U}}_i^n + (v_{i+1/2}^{n,*})^-\widetilde{\mathbb{U}}_{i+1}^n + \mathbb{H}_{i+1/2}^{n,*}\right] \\ &\quad + \frac{\Delta t}{\Delta x}\left[(v_{i-1/2}^{n,*})^+\widetilde{\mathbb{U}}_{i-1}^n + (v_{i-1/2}^{n,*})^-\widetilde{\mathbb{U}}_i^n + \mathbb{H}_{i-1/2}^{n,*}\right].\end{aligned} \tag{3.40}$$

Le schéma Lagrange-Projectionn proposé peut bien donc se mettre sous forme conservative avec le flux $\mathbb{F}_{i+1/2}^n = (v_{i+1/2}^{n,*})^+\widetilde{\mathbb{U}}_i^n + (v_{i+1/2}^{n,*})^-\widetilde{\mathbb{U}}_{i+1}^n + \mathbb{H}_{i+1/2}^{n,*}$ sur l'arête $i+1/2$. □

3.2.3 Condition de positivité et principe du maximum

Comme pour le schéma de relaxation explicite d'ordre 1 en coordonnées eulériennes, nous avons une garantie de positivité pour la densité ρ et la fraction massique Y. Le théorème qui suit est l'équivalent du Théorème 2.1.

***Théorème** 3.1. Sous la CFL*

$$\frac{\Delta t}{\Delta x}\max_i \max\left(|v_{i+1/2}^{n,*}|, b_{i+1/2}^n\right) < \frac{1}{2}, \tag{3.41}$$

nous avons pour le schéma Lagrange-Projection explicite

$$\rho_i^{n+1} > 0 \quad et \quad Y_i^{n+1} \in [0,1]. \tag{3.42}$$

Démonstration Tout d'abord, nous demandons la positivité de la densité ρ à l'issue de chaque phase de la méthode. Dans la phase Lagrange, en utilisant l'identité de Piola (3.32), nous avons $\widetilde{\rho}_i^n > 0$ dès que

$$1 + \frac{\Delta t}{\Delta x}(v_{i+1/2}^{n,*} - v_{i-1/2}^{n,*}) > 0, \quad i \in \mathbb{Z}. \tag{3.43}$$

Dans la phase de Projection, on constate que l'étape (3.30) peut encore s'écrire

$$\mathbb{U}_i^{n+1} = \widetilde{\mathbb{U}}_{i-1}^n\left[\frac{\Delta t}{\Delta x}(v_{i-1/2}^{n,*})^+\right] + \widetilde{\mathbb{U}}_i^n\left[1 - \frac{\Delta t}{\Delta x}((v_{i-1/2}^{n,*})^+ - (v_{i+1/2}^{n,*})^-)\right] + \widetilde{\mathbb{U}}_{i+1}^n\left[-\frac{\Delta t}{\Delta x}(v_{i+1/2}^{n,*})^-\right]. \tag{3.44}$$

On remarque que les coefficients de $\widetilde{\mathbb{U}}_{i-1}^n$ et $\widetilde{\mathbb{U}}_{i+1}^n$ sont positifs, la combinaison (3.44) est convexe si le coefficient de $\widetilde{\mathbb{U}}_i^n$ est aussi positif, c'est-à-dire si

$$1 - \frac{\Delta t}{\Delta x}\left[(v_{i-1/2}^{n,*})^+ - (v_{i+1/2}^{n,*})^-\right] > 0, \quad i \in \mathbb{Z}. \tag{3.45}$$

Pour assurer la positivité de ρ, le pas de temps Δt doit donc satisfaire les conditions (3.43) et (3.45). Il est évident que (3.45) est plus contraignante que (3.43), il suffit donc de satisfaire (3.45). De plus, en majorant $(v_{i-1/2}^{n,*})^+$ et $-(v_{i+1/2}^{n,*})^-$ par la valeur absolue de la vitesse en considération, nous obtenons la condition suffisante suivante

$$\frac{\Delta t}{\Delta x}|v_{i+1/2}^{n,*}| < \frac{1}{2}, \quad i \in \mathbb{Z}. \tag{3.46}$$

Pour avoir le principe du maximum sur la fraction massique du gaz Y dans la phase Lagrange, le pas de temps est contraint sous la CFL suivante pour éviter l'interaction entre les ondes

$$\frac{\Delta t}{\Delta x}b_{i+1/2}^n < \frac{1}{2}, \quad i \in \mathbb{Z}. \tag{3.47}$$

Sous la condition (3.46) et (3.47), nous pouvons conserver les propriétés physiques sur la positivité de la densité et de la fraction massique du gaz. Plus précisément :

$$\begin{aligned} (\rho)^n > 0 \Longrightarrow \quad & (\rho)^{n+1} > 0, \\ (\rho Y)^n > 0 \Longrightarrow \quad & (\rho Y)^{n+1} > 0 \Longrightarrow Y^{n+1} > 0, \\ (\rho^n - (\rho Y)^n) > 0 \Longrightarrow & (\rho^{n+1} - (\rho Y)^{n+1}) > 0 \Longrightarrow Y^{n+1} < 1. \ \square \end{aligned} \tag{3.48}$$

En Lagrange-Projection explicite, les valeurs de $(v^{n,*}_{i+1/2})^-$ et $(v^{n,*}_{i-1/2})^+$ sont connues. Il est donc facile de calculer le pas de temps convenable.

3.2.4 Traitements aux limites

Nous avons vu dans la section 2.4.2 du chapitre précédent le traitement des conditions aux limites pour le système de relaxation en eulérien. En Lagrange-Projection, ce traitement est encore plus « naturel » grâce à la structure très simples des ondes.

En effet, le système de relaxation en coordonnées lagrangiennes admet, pour un problème de Riemann, cinq ondes $\lambda_1 = -a, \lambda_2 = -b, \lambda_3 = 0, \lambda_4 = +b$ et $\lambda_5 = +a$. Puisque a et b sont positifs, il est évident que λ_4 et λ_5 (resp. λ_1 et λ_2) sont les ondes entrantes dans le domaine à l'amont (resp. à l'aval). Possédant deux conditions aux limites à imposer à chaque extrémité de la conduite, il suffira de considérer ces ondes entrantes comme ondes physiques portant l'information dans le domaine. Les autres ondes n'ont pas d'importance et seront donc « tuées » suivant la méthode présentée dans 2.4.2. Notons que cette façon de supprimer les ondes correspond à la méthode proposée par F. Dubois [45], et elle satisfait la condition d'inversibilité de Kreiss [62].

Nous rappelons que, par le vocable « tuer » les ondes, on entend l'annulation de l'amplitude de ces ondes en égalisant les invariants de Riemann fort (présentés dans la Fig. 3.2) associés aux ondes à tuer (voir 2.4.3). Nous montrons sur la Fig. 2.7 ces invariants du problème de Riemann pour le système de relaxation en lagrangien.

Fig. 3.2. Les invariants de Riemann forts du problème de Riemann avec état gauche $\mathbb{V}_L$ et droit $\mathbb{V}_R$ pour le système de relaxation (3.2).

3.2.4.1 À l'amont

Nous allons voir comment traiter les conditions aux limites pour la phase Lagrange, puis pour la phase Projection.

1. Phase Lagrange.
   Comme vu dans le chapitre précédemment, à l'amont nous allons supprimer trois ondes qui transportent $\Pi - av$, $\Sigma + bY$ et $\Pi + a\tau^2$.

Fig. 3.3. Les lignes en pointillé représentent les ondes à annuler à l'amont.

Les deux conditions aux limites imposées à l'amont reposent sur le débit total $q_T(t^n)$ et le débit de gaz $q_G(t^n)$ à l'instant t^n. Ces quantités varient en fonction du temps mais nous les notons dans la suite q_T et q_G. De plus, nous supprimons aussi l'indice en temps n sur toutes les variables pour abréger les formules. Nous avons

$$\begin{aligned} \rho_0 v_0 &= q_T, \\ Y_0 q_T - \Sigma_0 &= q_G. \end{aligned} \tag{3.49}$$

Supprimer les trois ondes λ_1, λ_2 et λ_3 revient à faire

$$\begin{aligned} \Pi_0 - a v_0 &= \Pi_1 - a v_1, \\ \Sigma_0 + b Y_0 &= \Sigma_1 + b Y_1, \\ \Pi_0 + a^2 \tau_0 &= \Pi_1 + a^2 \tau_1. \end{aligned} \tag{3.50}$$

De (3.49) et (3.50) nous avons cinq équations pour déterminer toutes les inconnues de l'état fictif à l'amont. On constate que l'on peut découpler ces équations en deux sous-systèmes indépendants. Le premier sous-système porte sur (Y_0, Σ_0)

$$\begin{aligned} Y_0 q_T - \Sigma_0 &= q_G, \\ b Y_0 + \Sigma_0 &= b Y_1 + \Sigma_1, \end{aligned} \tag{3.51}$$

et le deuxième sous-système sur (τ_0, v_0, Π_0)

$$\begin{aligned} v_0 &= q_T \tau_0, \\ \Pi_0 - a v_0 &= \Pi_1 - a v_1, \\ \Pi_0 + a^2 \tau_0 &= \Pi_1 + a^2 \tau_1. \end{aligned} \tag{3.52}$$

Après quelques calculs, l'état fictif à l'amont est finalement donné par

$$\begin{aligned} \tau_0 &= \frac{a\tau_1 + v_1}{a + q_T}, \\ Y_0 &= \frac{q_G + b Y_1 + \Sigma_1}{q_T + b}, \\ v_0 &= q_T \frac{a\tau_1 + v_1}{a + q_T}, \end{aligned} \quad \text{et} \quad \begin{aligned} \Pi_0 &= \Pi_1 + a^2 \frac{q_T \tau_1 - v_1}{a + q_T}, \\ \Sigma_0 &= \frac{b(q_T Y_1 - q_G) + q_T \Sigma_1}{q_T + b}. \end{aligned} \tag{3.53}$$

D'un point de vue purement théorique, nous pouvons vérifier que le fait d'imposer deux débits à l'entrée est compatible avec la condition de Kreiss [62]. Cette condition demande que le produit matriciel entre la matrice jacobienne des conditions limites imposées (3.49) et la matrice des vecteurs propres à droite correspondant aux ondes non-supprimées soit inversible. La démonstration se trouve dans la section 2.4.1.2.

2. Phase Projection.
En Lagrange-Projection explicite, l'état fictif pour la phase Projection est obtenue en remontant l'état fictif de la phase Lagrange, *i.e.*

$$\widetilde{\mathbb{V}}_0 = \mathbb{V}_0. \tag{3.54}$$

Faisons ainsi, on peut démontrer qu'à l'ordre 1 le flux numérique à l'entrée est bien respecté à l'interface 1/2.

3.2.4.2 À l'aval

1. Phase Lagrange.
À l'aval nous annulons trois ondes à droite à l'interface $N+1/2$, *i.e.* celles qui transporte $\Pi + a\tau^2, \Sigma - bY$ et $\Pi + a\tau$. Nous avons alors

$$\begin{aligned} \Pi_{N+1} + a v_{N+1} &= \Pi_N + a v_N, \\ \Sigma_{N+1} - b Y_{N+1} &= \Sigma_N - b Y_N, \\ \Pi_{N+1} + a^2 \tau_{N+1} &= \Pi_N + a^2 \tau_N, \end{aligned} \tag{3.55}$$

Fig. 3.4. Les lignes en pointillé représentent les ondes à annuler à l'aval.

À l'aval, la variable hors équilibre Π, relaxant la pression à l'état fictif est donnée par

$$\begin{aligned} \Pi_{N+1} &= p + \rho_{N+1} Y_{N+1}(1 - Y_{N+1})\phi^2(p, Y_{N+1}, v_{N+1}) \\ &= p + \Sigma_{N+1}\phi(p, Y_{N+1}, v_{N+1}). \end{aligned} \tag{3.56}$$

où p est une fonction $p(t)$ du temps imposée sur la maille fictive $N+1$.
De plus, on impose que la fraction massique du gaz Y soit égale à la fraction massique du premier état intérieur, *i.e.* $Y_{N+1} = Y_N$.

En remplaçant (3.56) dans la première équation de (3.55), nous avons

$$\begin{aligned} p + \Sigma_{N+1}\phi(p, Y_{N+1}, v_{N+1}) + a v_{N+1} &= \Pi_N + a v_N, \\ \Sigma_{N+1} &= \Sigma_N - b Y_N + b Y_{N+1}. \end{aligned} \tag{3.57}$$

La valeur de Σ_{N+1} est connue puisque l'on traite Σ séparément et explicitement. La complexité de résolution de l'équation (3.57) est comparable à celle de la loi hydrodynamique utilisée. En conséquence, la difficulté provient de la non-linéarité de la loi de glissement ϕ par rapport à v. Pour la loi Zuber-Findlay ou Zuber-Findlay-IFP, on aura une équation linéaire de ϕ par rapport à v, la résolution devient plus simple. En revanche, dans le cas général (loi de glissement quelconque), nous devons utiliser une méthode (peut-être itérative) pour trouver v_{N+1}. Une fois v_{N+1} connu, on peut déduire Π_{N+1} et τ_{N+1} par

$$\begin{aligned} \Pi_{N+1} &= \Pi_N + a(v_N - v_{N+1}), \\ \tau_{N+1} &= \tau_N + \frac{\Pi_N - \Pi_{N+1}}{a^2}. \end{aligned} \tag{3.58}$$

La condition de Kreiss (2.98) légitimant cette technique de suppression d'ondes peut être vérifiée facilement (voir section 2.4.2.2).

2. Phase Projection.
La mise à jour de l'état fictif aval pour la phase Projection est très simple. Il suffit de le remonter de l'instant précédent, c'est-à-dire
$$\widetilde{\mathbb{V}}_{N+1} = \mathbb{V}_{N+1}. \tag{3.59}$$
Comme à l'amont, nous ne mettons pas l'état fictif à l'équilibre ce qui nous permet de réduire le coût de calcul.

3.3 Montée en ordre

Nous allons traiter dans cette partie la montée en ordre du schéma Lagrange-Projection explicite. Bien qu'elles soient d'une complexité algorithmique plus importante, les méthodes numériques d'ordre supérieur à 1 restent intéressantes par rapport à celles d'ordre 1 car elles permettent, entre autre, de diminuer la diffusion numérique et d'avoir des fronts de chocs plus accentués. Dans l'ordre de la présentation, nous traitons d'abord la montée en ordre pour la phase Lagrange, puis pour la phase Projection.

3.3.1 Montée en ordre en espace

3.3.1.1 Phase Lagrange

En espace, la montée en ordre du schéma explicite est obtenue par la technique MUSCL (Monotonic Upstream Scheme for Conservation Laws [94]). Plus concrètement, au lieu de prendre une solution approchée constante par maille, on reconstruit une approximation linéaire par morceaux, comme illustré sur la Fig. 4.3. En effet, pour chaque maille i, on calcule deux pentes $\wp_{i-1}$ et $\wp_i$ avec

$$\wp_i = \mathbb{X}_{i+1} - \mathbb{X}_i, \tag{3.60}$$

où $\mathbb{X}$ peut être la variable conservative ou primitive. La pente choisie sur chaque maille i notée $\mathcal{P}_i$ pour la reconstruction linéaire n'est finalement ni $\wp_{i-1}$ ni $\wp_i$ mais une fonction de ces deux valeurs, à savoir

$$\mathcal{P}_i^{\mathbb{X}} = limiteur(\wp_{i-1}, \wp_i). \tag{3.61}$$

Le limiteur utilisé peut être la fonction minmod, qui renvoie la plus petite (en module) des deux pentes $\wp_{i-1}$ et $\wp_i$, mais aussi van Leer, Superbee ou Ultrabee.

On calcule les quantités $\mathbb{X}^R_{i-1/2}$ et $\mathbb{X}^L_{i+1/2}$ à gauche et à droite de l'interface $i+1/2$ suivant

$$\mathbb{X}^L_{i+1/2} = \mathbb{X}_i + \frac{1}{2}\mathcal{P}_i^{\mathbb{X}} \qquad \text{et} \qquad \mathbb{X}^R_{i-1/2} = \mathbb{X}_i - \frac{1}{2}\mathcal{P}_i^{\mathbb{X}}. \tag{3.62}$$

Fig. 3.5. Reconstruction linéaire.

L'expérience montre que pour la phase Lagrange, il est préférable de limiter des pentes sur la « variable physique » $(Y, v, P)^T$ que sur la variable conservative $(\rho, \rho Y, \rho v)^T$. On peut déduire facilement les variables conservatives initiales $\mathbb{U}^R_{i-1/2}$ et $\mathbb{U}^L_{i+1/2}$. On injecte ensuite $\mathbb{U}^L_{i+1/2}$ et $\mathbb{U}^R_{i+1/2}$ comme états gauches et droits dans le calcul du flux $\mathbb{F}_{i+1/2}(\mathbb{U}^L_{i+1/2}, \mathbb{U}^R_{i+1/2})$ sur chaque interface $i+1/2$.

3.3.1.2 Phase Projection

La montée en ordre en espace pour la phase Projection se fait de manière différente que celle utilisée pour la phase Lagrange puisque la phase Projection consiste en un transport pur.

Nous commençons tout d'abord par un rappel sur la reconstruction d'un schéma d'advection d'ordre 2 pour un système de transport scalaire linéaire suivant

$$\partial_t u + \vartheta \partial_x u = 0, \quad \vartheta \in \mathbb{R}, \ u \in \mathbb{R}. \tag{3.63}$$

Alors le schéma explicite d'ordre 2 pour ce système s'écrit

$$\widehat{u}_i = u_i - \mu \left[u^L_{i+1/2} - u^R_{i-1/2} \right], \tag{3.64}$$

avec

$$u^L_{i+1/2} = u_i + \frac{1-|\mu|}{2} \mathcal{D}^u_i \qquad \text{et} \qquad u^R_{i-1/2} = u_i - \frac{1-|\mu|}{2} \mathcal{D}^u_i, \tag{3.65}$$

Ici, μ est la condition CFL

$$\mu = \vartheta \frac{\Delta t}{\Delta x}, \tag{3.66}$$

et $\mathcal{D}^u_i$ est la pente (multipliée par Δx) sur la maille i

$$\mathcal{D}^u_i = limiteur(\wp_{i-1}, \wp_i), \tag{3.67}$$

avec

$$\wp_i = u_{i+1} - u_i, \tag{3.68}$$

Il est connu [65, 88] que sous la condition CFL $\mu < 1$ et si le $limiteur(\wp_{i-1}, \wp_i)$ vérifie

$$\begin{aligned} \frac{2}{\mu} \min(\wp_{i-1}, 0) \leq limiteur(\wp_{i-1}, \wp_i) \leq \ \frac{2}{\mu} \max(\wp_{i-1}, 0), \\ \frac{2}{1-\mu} \min(\wp_i, 0) \leq limiteur(\wp_{i-1}, \wp_i) \leq \frac{2}{1-\mu} \max(\wp_i, 0), \end{aligned} \tag{3.69}$$

nous garantissons le principe du maximum sur u, *i.e.*

$$\min(u_{i-1}, u_i) \leq \widehat{u}_i \leq \max(u_{i-1}, u_i). \tag{3.70}$$

Appliquons cette reconstruction à la notre schéma Lagrange-Projection où la phase Projection (transport pur) est couplée avec la phase Lagrange. Rappelons que pour ce schéma, nous préférons utiliser la forme conservative (3.40) qui s'écrit, en explicite premier ordre, comme suivant

$$\begin{aligned}\mathbb{U}_i^{n+1} = \mathbb{U}_i^n &- \frac{\Delta t}{\Delta x}\left[(v_{i+1/2}^{n,*})^+\widetilde{\mathbb{U}}_i^n + (v_{i+1/2}^{n,*})^-\widetilde{\mathbb{U}}_{i+1}^n + \mathbb{H}_{i+1/2}^{n,*}\right] \\ &+ \frac{\Delta t}{\Delta x}\left[(v_{i-1/2}^{n,*})^+\widetilde{\mathbb{U}}_{i-1}^n + (v_{i-1/2}^{n,*})^-\widetilde{\mathbb{U}}_i^n + \mathbb{H}_{i-1/2}^{n,*}\right].\end{aligned} \tag{3.71}$$

Pour obtenir le schéma L-P explicite d'ordre deux pour la phase Projection, dans (3.71) on exprime $\mathbb{U}_i^n$ en fonction de $\widetilde{\mathbb{U}}_i^n$ (solution de la phase Lagrange) et on replace les états au centre des mailles par les états reconstruits correspondants, à savoir :

$$\begin{aligned}
(v_{i+1/2}^{n,*})^+\widetilde{\mathbb{U}}_i^n &:= (v_{i+1/2}^{n,*})^+\mathbb{U}_{i+1/2}^{n,L}, & \mathbb{U}_{i+1/2}^{n,L} &= \mathbb{U}_i^n + \frac{1-|\mu_i^n|}{2}\mathcal{D}_i^{\widetilde{\mathbb{U}}}, \\
(v_{i+1/2}^{n,*})^-\widetilde{\mathbb{U}}_{i+1}^n &:= (v_{i+1/2}^{n,*})^-\mathbb{U}_{i+1/2}^{n,R}, & \mathbb{U}_{i+1/2}^{n,R} &= \mathbb{U}_{i+1}^n - \frac{1-|\mu_{i+1}^n|}{2}\mathcal{D}_{i+1}^{\widetilde{\mathbb{U}}}, \\
(v_{i-1/2}^{n,*})^+\widetilde{\mathbb{U}}_{i-1}^n &:= (v_{i-1/2}^{n,*})^+\mathbb{U}_{i-1/2}^{n,L} & \mathbb{U}_{i-1/2}^{n,L} &= \mathbb{U}_{i-1}^n + \frac{1-|\mu_{i-1}^n|}{2}\mathcal{D}_{i-1}^{\widetilde{\mathbb{U}}}, \\
(v_{i-1/2}^{n,*})^-\widetilde{\mathbb{U}}_i^n &:= (v_{i-1/2}^{n,*})^-\mathbb{U}_{i-1/2}^{n,R}, & \mathbb{U}_{i-1/2}^{n,R} &= \mathbb{U}_i^n - \frac{1-|\mu_i^n|}{2}\mathcal{D}_i^{\widetilde{\mathbb{U}}},
\end{aligned} \tag{3.72}$$

avec

$$|\mu_i^n| = (\mu_{i-1/2}^n)^+ - (\mu_{i+1/2}^n)^- \qquad \text{et} \qquad \mu_{i\pm1/2}^n = v_{i\pm1/2}^{n,*}\frac{\Delta t}{\Delta x}, \tag{3.73}$$

et les $\mathcal{D}_i^{\widetilde{\mathbb{U}}} = (\mathcal{D}^{(\rho)}, \mathcal{D}^{(\rho Y)}, \mathcal{D}^{(\rho v)})^T$ sont les vecteurs des pentes calculées sur chaque composante de $\mathbb{U} \equiv (\rho, \rho Y, \rho v)^T \in \Omega$ en utilisant la formule (3.67) présentée précédemment. Après quelques calculs, la montée en ordre du schéma L-P pour la phase Projection s'écrit

$$\begin{aligned}\mathbb{U}_i^{n+1} = \widetilde{\mathbb{U}}_i^n &- \mu_{i-1/2}^-\frac{1-|\mu_i^n|}{2}\mathcal{D}_i^{\widetilde{\mathbb{U}}} - \mu_{i-1/2}^+\Big\{\widetilde{\mathbb{U}}_i^n - (\widetilde{\mathbb{U}}_{i-1}^n + \frac{1-|\mu_{i-1}^n|}{2}\mathcal{D}_{i-1}^{\widetilde{\mathbb{U}}})\Big\} \\ &- \mu_{i+1/2}^+\frac{1-|\mu_i^n|}{2}\mathcal{D}_i^{\widetilde{\mathbb{U}}} + \mu_{i+1/2}^-\Big\{\widetilde{\mathbb{U}}_i^n - (\widetilde{\mathbb{U}}_{i+1}^n - \frac{1-|\mu_{i+1}^n|}{2}\mathcal{D}_{i+1}^{\widetilde{\mathbb{U}}})\Big\}.\end{aligned} \tag{3.74}$$

Dans le contexte des simulations d'écoulements diphasiques, cette reconstruction permet d'assurer le principe du maximum sur la densité ρ et la densité partielle ρY (deux premiers composantes de $\mathbb{U}$), c'est-à-dire

$$\begin{aligned}\rho_i^n > 0 &\Rightarrow \rho_i^{n+1} > 0, \\ (\rho Y)_i^n > 0 &\Rightarrow (\rho Y)_i^{n+1} > 0.\end{aligned} \tag{3.75}$$

Dans ce contexte, le principe du maximum sur la fraction massique du gaz Y, indispensable pour la robustesse du schéma, est aussi souhaité.

$$Y_i^n \in [0,1] \Rightarrow Y_i^{n+1} \in [0,1]. \tag{3.76}$$

Les inégalités dans (3.75) permettent d'assurer uniquement la borne inférieure. En effet, initialement les pentes $\mathcal{D}_i^{(\rho)}$ et $\mathcal{D}_i^{(\rho Y)}$ sur les composantes ρ et ρY sont calculées indépendamment l'une de l'autre, ce qui ne garantit pas la propriété physique de Y. Pour satisfaire pleinement la condition (3.76), nous devons utiliser d'autres valeurs pour ces pentes. L'idée est que, en fonction des valeurs de $\wp_{i-1}$, $\wp_i$ et de μ, les nouvelles pentes $\widetilde{\mathcal{D}}_i^{(\rho)}$ et $\widetilde{\mathcal{D}}_i^{(\rho Y)}$ « se voient » pour avoir une pente « virtuelle » sur $Y = \dfrac{\rho Y}{\rho}$. Les calculs de ces pentes sont couplés de sorte qu'elles restent le plus proche possible des pentes originales (3.67). Nous renvoyons les lecteurs à l'article de Tran [92] pour les détails du calcul.

3.3.2 Montée en ordre en temps

En temps, la montée en ordre du schéma explicite se fait grâce à la méthode de Runge-Kutta d'ordre 2. Notons $\mathbb{L}$ l'opérateur d'évolution du premier ordre en temps défini par le schéma explicite, l'algorithme se décompose en trois étapes

- Premier demi-pas de temps : $\bar{\mathbb{U}} = \mathbb{L}(\mathbb{U}^n)$.
- Deuxième demi-pas de temps : $\bar{\bar{\mathbb{U}}} = \mathbb{L}(\bar{\mathbb{U}})$.
- Mise à jour définitive : $\mathbb{U}^{n+1} = \frac{1}{2}(\mathbb{U}^n + \bar{\bar{\mathbb{U}}})$.

3.4 Intégration du terme source

Jusqu'à présent, nous n'avons pas encore évoqué le terme source S. En pratique, il y a plusieurs façon de prendre en compte : avec *splitting* et sans *splitting*. Ici, nous allons décire la technique la plus simple pour intégrer S dans le cas Lagrange-Projection explicite.

Rappelons l'écriture sous la forme conservative du système initial (1.10) :

$$\partial_t \mathbb{U} + \partial_x \mathcal{F}(\mathbb{U}) = \mathbb{T}(\mathbb{U}), \tag{3.77}$$

où $\mathbb{T}(\mathbb{U}) = (0, 0, -\rho g \sin\mathcal{T} - c\rho v|v|)^T$. Ici, c est une constante lié au frottement.

À cause du terme source, il n'y a pas de solution auto-semblable de Riemann pour (3.77). En revanche, nous savons résoudre séparément dans les cas suivants
- sans source :

$$\partial_t \mathbb{U} + \partial_x \mathcal{F}(\mathbb{U}) = 0, \tag{3.78}$$

- lié uniquement au terme source :

$$\partial_t \mathbb{U} = \mathbb{T}(\mathbb{U}). \tag{3.79}$$

La technique d'intégration du terme source que nous allons présenter ici consiste à utiliser cette méthode de *splitting*, c'est-à-dire que l'on va intégrer la source à la fin de la phase de Projection. Plus précisément, après avoir calculé la solution $\mathbb{U}^{n+1}$ sans source utilisant (3.40), on va résoudre le problème de Cauchy suivant :

$$\begin{aligned} \mathrm{d}_t(\rho) &= 0, \\ \mathrm{d}_t(\rho Y) &= 0, \\ \mathrm{d}_t(\rho v) &= -\rho g \sin\mathcal{T} - c\rho|v|v, \end{aligned} \tag{3.80}$$

pour une donnée initiale $\mathbb{U}^{n+1}(x,t)$.

Les valeurs usuelles de la constante c rentrant dans le terme de frottement sont grandes, de sorte que le système d'EDO (3.80) est raide. Il convient donc d'opposer une intégration inconditionnellement stable, *i.e.* sans restriction sur la taille du pas de temps. A cette fin, nous suggérons d'approcher la solution de (3.80) en intégrant exactement l'EDO

$$\begin{aligned} \mathrm{d}_t(v) &= -g \sin\mathcal{T} - c|v^{n+1}|v, \\ v(x, t=0) &= v^{n+1}, \end{aligned} \tag{3.81}$$

où nous avons gelé la non-linéarité en v sur la donnée initiale $\mathbb{U}^{n+1}(x,t)$.

Après quelques calculs simples, la solution de (3.81) est donnée par

$$v(t) = v^{n+1}e^{-tc|v^{n+1}|} - \frac{g\sin\theta}{c|v^{n+1}|}(1 - e^{-tc|v^{n+1}|}), \tag{3.82}$$

d'où la modification de la vitesse v^{n+1} à l'instant t^{n+1} dans le cas avec source.

3.4.1 Résultats numériques

Dans cette partie, nous présentons des résultats numériques avec le schéma Lagrange-Projection explicite : nous montrons un cas-test sur les tubes à choc, et un cas-test sur les conditions limites. Pour chaque cas-test, nous allons comparer ces résultats numériques avec ceux obtenus le schéma eulérien présenté dans le chapitre 2. Les deux schémas utilisent une reconstruction de pente en espace afin de monter en ordre du schéma.

3.4.1.1 Validation sur tubes à choc

Le premier cas-test est de type tube à choc, c'est-à-dire que l'on résout un pur problème de Riemann. On ne tient donc compte ni des conditions aux limites ni de la contribution du terme source dans cette expérience. Les coefficients de relaxation a et b sont pris locaux pour le schéma Lagrange-Projection afin de comparer avec le schéma eulérien.

La conduite que l'on utilise ici fait 100 m de long et est inclinée à 30 degrés vers le haut. Le pas d'espace est $\Delta x = 0.5$ m. On travaille avec un liquide incompressible ($\rho_l = 1000$ kg/m^3) et un gaz parfait. La loi hydrodynamique est de type Zuber-Findlay CEA. La condition CFL explicite est égale à 0.5. La vitesse du son dans le gaz est pris à $a_g = 100$ m/s.

La discontinuité initiale entre les états gauche (L) et droit (R) est située au milieu de la conduite à $x = 50$ m

$$\begin{pmatrix} \rho \\ Y \\ v \end{pmatrix}_L = \begin{pmatrix} 500 \\ 0.2 \\ 34.4233 \end{pmatrix} \quad \text{et} \quad \begin{pmatrix} \rho \\ Y \\ v \end{pmatrix}_R = \begin{pmatrix} 400 \\ 0.2 \\ 50 \end{pmatrix}.$$

La solution de Riemann est composée de trois ondes, à savoir un choc, une discontinuité de contact et un choc. Les vitesses de propagation sont recpectivement $\omega^- = -77.7$ m/s pour le choc à gauche, $\omega^0 = -4.62$ m/s pour la discontinuité de contact et $\omega^+ = 76.7$ m/s pour le choc à droite.

Le résultat illustré dans la Fig. 3.6 correspond au temps $T = 0.3$ secondes. Sur la densité ρ ainsi que la vitesse v, on constate qu'il y a trois ondes : une onde lente se propageant vers la gauche à une vitesse lente et deux ondes rapides se propageant à des vitesses rapides, une vers la gauche et une vers la droite de la conduite. On retouve les vitesses de propagation correspondantes mentionnées ci-dessus. Sur la fraction massique on observe bien le phénomène lente du fait qu'elle est liée au transport. De plus, à cause du glissement, nous voyons aussi les deux phénomènes rapides sur Y. En revanche, sur la pression P il n'y a que des phénomènes rapides, ce qui est cohérent avec le fait que P est liée uniquement à l'acoustique.

On constate que les résultats des deux schémas eulérien et Lagrange-Projection sont comparables : on obtient la même précision pour les deux schémas tant sur les ondes rapides que sur les ondes lentes.

Fig. 3.6. Expérience tube à choc en explpicite - 3 ondes - Glissement Zuber-Findlay CEA.

3.4.1.2 Validation avec des conditions aux limites

Nous considérons maintenant un cas-test réaliste prenant en compte les conditions aux limites et la source. Dans ce type de cas-test, les conditions aux limites varient pendant un certain temps que l'on appelle *durée de la rampe*, puis restent constantes jusqu'à la fin de la simulation. Plus précisément pour ce cas concret, le débit de gaz est doublé pendant un temps très court (1 seconde) tandis que le débit de liquide et la pression sont constants. Le paramètre *Area* est la section de la conduite (en m^2).

Pour réaliser la simulation, on calcule tout d'abord une solution *stationnaire* à partir des conditions aux limites initiales, puis on démarre la simulation pendant une période de *stabilisation* afin d'avoir une solution *stationnaire numérique*. On modifie ensuite les conditions aux limites. Les caractéristiques mécaniques sont :

- coefficient de frottement : $C_f = 0.005$,
- vitesse du son dans le gaz : $a_g = \sqrt{10^5}$ m/s,
- vitesse du son dans le liquide : $a_l = 500$ m/s,
- masse volumique du liquide : $\rho_l^0 = 1000$ kg/m^3.

Ce cas-test a été utilisé dans [79] et est considéré comme un cas-test « dur » car la rampe est très raide. Les détails de cette expérience se trouvent dans la Fig. 3.8 et les résultats sont présentés dans la Fig. 3.7. On constate tout d'abord que les deux schémas convergent vers l'état stationnaire à $t = 10000$ s. Les deux schémas Lagrange-Projection et eulérien en explicite donnent des résultats tout à fait similaires tant sur la fraction massique du gaz que sur la pression. De plus, les conditions aux limites sont bien respectées pour les deux schémas : le débit du gaz augmente de 0.2 à 0.4 kg/Area/s tandis que le débit du liquide reste pratiquement constant à 20 kg/Area/s (légères variations pendant la rampe).

Fig. 3.7. Expérience avec conditions aux limites - Eulérien (gauche) et Lagrange-Projection (droite) - Schéma explicite.

Géométrie de la conduite :	- Inclinaison	:	horizontale,	
	- Longueur	:	10000 m,	
	- Diamètre	:	0.146 m	
	- Pas d'espace	:	100 m	
Discrétisation :	- CFL explicite	:	0.5	
Source et Ordre :	- Avec source et reconstruction de pente			
Modèles :	- Thermodynamique	:	liquide incompressible	
	- Hydrodynamique	:	glissement nul	
Conditions opératoires :	- Durée de la stabilisation			: 1000 s
	- Durée de la rampe			: 1 s
	- Condition limite amont sur le débit de gaz			: 0.2 kg/Area/s
				à 0.4 kg/Area/s
	- Condition limite amont sur le débit de liquide			: 20 kg/Area/s
	- Condition limite aval sur la pression			: 10 bar

Fig. 3.8. Détails de l'expérience avec conditions aux limites - Schéma Explicite.

3.4.1.3 Convergence en maillage

Dans cette partie, nous allons quantifier l'erreur commise pour les deux schémas explicites. Plus précisément, nous allons évaluer numériquement la norme L^1 en espace de la différence entre la solution obtenue et la solution de référence. Cette erreur, par définition, est donnée par

$$E(h) = \frac{1}{N}\sum_{i=1}^{N}\left|\frac{1}{m}\sum_{j=1}^{m}\mathbb{U}_{i,j}^{h} - \mathbb{U}_{i}^{H}\right|, \tag{3.83}$$

où

- N est le nombre de mailles sur la grille uniforme où on veut calculer l'erreur de la solution,
- m est le ratio entre le nombre de mailles sur la grille de référence (fine) et celui sur la grille considérée,
- $\mathbb{U}_{i}^{H}$ est la solution au centre de la i ième maille sur la grille uniforme considérée,
- $\mathbb{U}_{i,j}^{h}$ est la solution au centre de la $(i-1)*m+j$ ième maille sur la grille de référence.

Prenons une condition initiale très régulière, à savoir

$$\begin{array}{ll} v = 10 \text{ m/s}, & \\ P = 200 \text{ Pa}, & \end{array} \quad \text{et} \quad \begin{array}{l} Y = 0.25 + 0.5 \times e^{10^{-4}\times(x-L/2)^2}, \\ \rho = \rho(Y,v,P) \text{ kg/m}^3, \end{array} \tag{3.84}$$

où x est la position courante et L est la longueur de la conduite. La conduite est horizontale et le glissement est nul.

La solution de référence est la solution d'une expérience réalisée avec beaucoup de mailles (32768 mailles), elle est donc considérée comme très fine. La Fig. 3.9 montre l'évolution de l'erreur absolue de la densité en fonction de la finesse de discrétisation d'un schéma avec reconstruction de pente. On constate que les erreurs diminuent en fonction du nombre de mailles : plus on raffine le maillage, plus l'erreur devient petite. De plus, ces erreurs décroissent de la même façon, voire coïncident pour les deux méthodes. On constate aussi que l'on obtient bien l'ordre deux numériquement pour ce cas-test où la donnée est très régulière. Notons qu'en pratique, pour des solutions initiales discontinues, l'ordre deux n'est pas atteint.

Fig. 3.9. Convergence de l'erreur absolue de la densité en norme L^1 pour les schémas explicites.

3.4.1.4 Conclusion sur le schéma L-P explicite

Nous venons de présenter quelques résultats en explicite avec deux schémas de relaxation: eulérien et Lagrange-Projection. On constate qu'avec une reconstruction de pente en espace (ordre supérieur à 1), nous obtenons des résultats similaires pour les deux schémas explicites : le niveau de diffusion est identique. Le développement du schéma explicite n'est pas le but principal de ce travail mais c'est une étape préliminaire pour valider la méthode Lagrange-Projection empruntée, avant de passer en semi-implicite où nous espérons obtenir des résultats plus intéressants.

3.5 Méthode de relaxation-Lagrange-Projection en semi-implicite

La motivation pour la mise au point d'un schéma sélectivement implicite dans le contexte des écoulements pétroliers a déjà été expliquée dans l'Introduction et au début de la section 2.3. On souhaite que la discrétisation temporelle soit implicite par rapport aux ondes acoustiques rapides, tout en restant explicite par rapport à aux ondes cinématiques lentes. L'implicite par rapport aux ondes de pression, qui ne présentent aucun intérêt pour le pétrolier, permet d'accélérer les simulations. L'explicite par rapport aux ondes de matière, qui intéressent au plus haut point les ingénieurs, permet de garantir la précision sur ces ondes.

A la différence du schéma eulérien présenté en 2.3, la version semi-implicite du schéma relaxation-Lagrange-Projection que nous allons élaborer aura bien une garantie de positivité de la densité et le principe du maximum de la fraction massique du gaz. En effet, pour le premier schéma le pas de temps est calculé à partir d'une CFL^{exp} basée sur les ondes lentes $v \pm b\tau$ et d'une CFL^{imp} basée sur les ondes rapides $v \pm a\tau$. Aucun calcul ne permet de vérifier les conditions de positivité. En revanche, le schéma relaxation-Lagrange-Projection semi-implicite permet d'établir une CFL calée sur la vitesse de transport ϑ de la phase convective afin de conserver ces propriétés. Cette condition CFL est explicitement calculable, à partir de laquelle nous obtenons un pas de temps pour répondre à nos exigences. Le calcul menant à la condition CFL mérite donc toute notre attention.

De surcroît, le schéma relaxation-L-P semi-implicite permet de réduire le temps de calcul grâce à la taille petite des blocs (2×2) dans le système linéaire à résoudre. Rappelons que pour le schéma eulérien semi-implicite, les blocs sont de taille 5×5.

3.5.1 Discrétisation en (τ, Y, v)

Comme vu précédemment, le schéma Lagrange-Projection décompose naturellement le flux en une partie acoustique et une partie de transport. Pour obtenir le schéma semi-implicite, il suffira en gros d'impliciter partiellement la phase Lagrange et d'expliciter la phase Projection.

Avec les mêmes notations qu'en explicite, on propose la discrétisation suivante pour la phase Lagrange

$$\begin{aligned}
&\rho_i^n \frac{\widetilde{\tau}_i^n - \tau_i^n}{\Delta t} - \frac{\widetilde{v}_{i+1/2}^{n,*} - \widetilde{v}_{i-1/2}^{n,*}}{\Delta x} = 0,\\
&\rho_i^n \frac{\widetilde{v}_i^n - v_i^n}{\Delta t} + \frac{\widetilde{\Pi}_{i+1/2}^{n,*} - \widetilde{\Pi}_{i-1/2}^{n,*}}{\Delta x} = 0,\\
&\rho_i^n \frac{\widetilde{\Pi}_i^n - \Pi_i^n}{\Delta t} + a^2 \frac{\widetilde{v}_{i+1/2}^{n,*} - \widetilde{v}_{i-1/2}^{n,*}}{\Delta x} = \eta \rho_i^n (P(\widetilde{\tau}_i^n, \widetilde{v}_i^n, \widetilde{Y}_i^n) - \widetilde{\Pi}_i^n),\\
&\rho_i^n \frac{\widetilde{Y}_i^n - Y_i^n}{\Delta t} - \frac{\Sigma_{i+1/2}^{n,*} - \Sigma_{i-1/2}^{n,*}}{\Delta x} = 0,\\
&\rho_i^n \frac{\widetilde{\Sigma}_i^n - \Sigma_i^n}{\Delta t} - b^2 \frac{Y_{i+1/2}^{n,*} - Y_{i-1/2}^{n,*}}{\Delta x} = 0,
\end{aligned} \tag{3.85}$$

avec les flux numériques de Godunov

$$\begin{aligned}
\widetilde{v}_{i+1/2}^{n,*} &= \frac{\widetilde{v}_i^n + \widetilde{v}_{i+1}^n}{2} - \frac{\widetilde{\Pi}_{i+1}^n - \widetilde{\Pi}_i^n}{2a}, \qquad \widetilde{Y}_{i+1/2}^{n,*} = \frac{\widetilde{Y}_i^n + \widetilde{Y}_{i+1}^n}{2} + \frac{\widetilde{\Sigma}_{i+1}^n - \widetilde{\Sigma}_i^n}{2b},\\
\widetilde{\Pi}_{i+1/2}^{n,*} &= \frac{\widetilde{\Pi}_i^n + \widetilde{\Pi}_{i+1}^n}{2} - a \frac{\widetilde{v}_{i+1}^n - \widetilde{v}_i^n}{2}, \qquad \widetilde{\Sigma}_{i+1/2}^{n,*} = \frac{\widetilde{\Sigma}_i^n + \widetilde{\Sigma}_{i+1}^n}{2} + b \frac{\widetilde{Y}_{i+1}^n - \widetilde{Y}_i^n}{2}.
\end{aligned} \tag{3.86}$$

On remarque que ce système est découplé en deux blocs : un bloc lié à (τ, v, Π) et un bloc lié à (Y, Σ). Le premier bloc avec un terme de relaxation est piloté par les ondes rapides $\pm a$ et donc implicite. Nous allons montrer comment aborder ce dernier système de manière implicite dans le régime de grand paramètre de relaxation η, *i.e.* $\eta \to +\infty$. De manière à traiter ce régime, nous adoptons l'approche développé par Baudin et *al* [6] consistant à réécrire dans un premier temps l'équation en Π selon

$$\widetilde{\Pi}_i^n = P(\widetilde{\tau}_i^n, \widetilde{v}_i^n, \widetilde{Y}_i^n) - \frac{1}{\eta} \left[\frac{\widetilde{\Pi}_i^n - \Pi_i^n}{\Delta t} + a^2 \frac{\widetilde{v}_{i+1/2}^{n,*} - \widetilde{v}_{i-1/2}^{n,*}}{\rho_i^n \Delta x} \right], \tag{3.87}$$

pour en proposer une approximation linéarisé au premier ordre dans la limite formelle $\eta \to +\infty$. Cette approximation linéarisée est donnée par

$$\begin{aligned}
\widetilde{\Pi}_i^n = P(\tau_i^n, v_i^n, Y_i^n) &+ \partial_\tau P(\tau_i^n, v_i^n, Y_i^n) \delta\tau_i\\
&+ \partial_v P(\tau_i^n, v_i^n, Y_i^n) \delta v_i\\
&+ \partial_Y P(\tau_i^n, v_i^n, Y_i^n) \delta Y_i,
\end{aligned} \tag{3.88}$$

avec les incréments

$$\delta\tau_i = \widetilde{\tau}_i^n - \tau_i^n, \quad \delta v_i = \widetilde{v}_i^n - v_i^n, \quad \delta Y_i = \widetilde{Y}_i^n - Y_i^n. \tag{3.89}$$

Cette formule exprime que l'inconnue $\widetilde{\Pi}_i^n$ est en réalité une fonction affine des incréments $\delta\tau_i, \delta v_i$ et δY_i.

Puisque le dernier incrément est explicite, il s'ensuit que la résolution des équations implicites (3.85) peut être ramenée après substitution de $\widetilde{\Pi}_i^n$ grâce à (3.87) à la résolution d'un système linéaire en les deux inconnues $\widetilde{\tau}_i^n$ et $\widetilde{v}_i^n$, à savoir

$$\begin{aligned} \rho_i^n \frac{\widetilde{\tau}_i^n - \tau_i^n}{\Delta t} - \frac{\widetilde{v}_{i+1/2}^{n,*} - \widetilde{v}_{i-1/2}^{n,*}}{\Delta x} &= 0, \\ \rho_i^n \frac{\widetilde{v}_i^n - v_i^n}{\Delta t} + \frac{\widetilde{\Pi}_{i+1/2}^{n,*} - \widetilde{\Pi}_{i-1/2}^{n,*}}{\Delta x} &= 0. \end{aligned} \tag{3.90}$$

Introduisons maintenant un coefficient d'implicitation θ compris entre 0 et 1 : pour $\theta = 0$ on obtient un schéma explicite et la valeur $\theta = 1/2$ correspond à la méthode de Crank-Nicholson. Le premier bloc s'écrit alors

$$\begin{aligned} \rho_i^n \frac{\widetilde{\tau}_i^n - \tau_i^n}{\Delta t} - \frac{\widetilde{v}_{i+1/2}^{n,*,\theta} - \widetilde{v}_{i-1/2}^{n,*,\theta}}{\Delta x} &= 0, \\ \rho_i^n \frac{\widetilde{v}_i^n - v_i^n}{\Delta t} + \frac{\widetilde{\Pi}_{i+1/2}^{n,*,\theta} - \widetilde{\Pi}_{i-1/2}^{n,*,\theta}}{\Delta x} &= 0, \end{aligned} \tag{3.91}$$

avec

$$\begin{aligned} \widetilde{v}_{i+1/2}^{n,*,\theta} &= (1-\theta) v_{i+1/2}^{n,*} + \theta \widetilde{v}_{i+1/2}^{n,*}, \\ \widetilde{\Pi}_{i+1/2}^{n,*,\theta} &= (1-\theta) \Pi_{i+1/2}^{n,*} + \theta \widetilde{\Pi}_{i+1/2}^{n,*}, \end{aligned} \tag{3.92}$$

et $v_{i+1/2}^{n,*}$ et $\Pi_{i+1/2}^{n,*}$ données par la solution du problème de Riemann à l'interface $i+1/2$, à savoir

$$\begin{aligned} v_{i+1/2}^{n,*} &= \frac{v_i^n + v_{i+1}^n}{2} - \frac{\Pi_{i+1}^n - \Pi_i^n}{2a}, \\ \Pi_{i+1/2}^{n,*} &= \frac{\Pi_i^n + \Pi_{i+1}^n}{2} - a \frac{v_{i+1}^n - v_i^n}{2}. \end{aligned} \tag{3.93}$$

Le deuxième bloc de (3.85) est piloté par les ondes lentes $\pm b$, il sera donc traité de manière explicite, à savoir

$$\rho_i^n \frac{\widetilde{Y}_i^n - Y_i^n}{\Delta t} - \frac{\Sigma_{i+1/2}^{n,*} - \Sigma_{i-1/2}^{n,*}}{\Delta x} = 0, \tag{3.94}$$

avec

$$\Sigma_{i+1/2}^{n,*} = \frac{\Sigma_i^n + \Sigma_{i+1}^n}{2} + b \frac{Y_{i+1}^n - Y_i^n}{2}. \tag{3.95}$$

On remarque que le fait de traiter Y explicitement dans la phase Lagrange est « naturel » et conforme à notre philosophie : on n'implicite que ce qui est relié à l'acoustique.

On introduit l'incrément $\delta\Pi_i = \widetilde{\Pi}_i^n - \Pi_i^n$ ainsi que les incréments

$$\begin{aligned} \delta v_{i+1/2}^{n,*} &= \frac{\delta v_i^n + \delta v_{i+1}^n}{2} - \frac{\delta \Pi_{i+1}^n - \delta \Pi_i^n}{2a}, \\ \delta \Pi_{i+1/2}^{n,*} &= \frac{\delta \Pi_i^n + \delta \Pi_{i+1}^n}{2} - a \frac{\delta v_{i+1}^n - \delta v_i^n}{2}. \end{aligned} \tag{3.96}$$

Alors

$$\begin{aligned} \widetilde{v}_{i+1/2}^{n,*} &= v_{i+1/2}^{n,*} + \frac{\delta v_i^n + \delta v_{i+1}^n}{2} - \frac{\delta \Pi_{i+1}^n - \delta \Pi_i^n}{2a}, \\ \widetilde{\Pi}_{i+1/2}^{n,*} &= \Pi_{i+1/2}^{n,*} + \frac{\delta \Pi_i^n + \delta \Pi_{i+1}^n}{2} - a \frac{\delta v_{i+1}^n - \delta v_i^n}{2}, \end{aligned} \tag{3.97}$$

et (3.92) s'écrit

$$\begin{aligned} \widetilde{v}^{n,*,\theta}_{i+1/2} &= v^{n,*}_{i+1/2} + \theta \delta v^{n,*}_{i+1/2}, \\ \widetilde{\Pi}^{n,*,\theta}_{i+1/2} &= \Pi^{n,*}_{i+1/2} + \theta \delta \Pi^{n,*}_{i+1/2}. \end{aligned} \tag{3.98}$$

En injectant les relations de (3.98) dans (3.91), on obtient de nouvelles équations à résoudre

$$\begin{aligned} \delta\tau_i - \frac{\theta\Delta t}{\rho_i^n \Delta x}(\delta v^{n,*}_{i+1/2} - \delta v^{n,*}_{i-1/2}) &= \frac{\Delta t}{\rho_i^n \Delta x}(v^{n,*}_{i+1/2} - v^{n,*}_{i-1/2}), \\ \delta v_i + \frac{\theta\Delta t}{\rho_i^n \Delta x}(\delta \Pi^{n,*}_{i+1/2} - \delta \Pi^{n,*}_{i-1/2}) &= -\frac{\Delta t}{\rho_i^n \Delta x}(\Pi^{n,*}_{i+1/2} - \Pi^{n,*}_{i-1/2}), \\ \delta Y_i &= \frac{\Delta t}{\rho_i^n \Delta x}(\Sigma^{n,*}_{i+1/2} - \Sigma^{n,*}_{i-1/2}). \end{aligned} \tag{3.99}$$

On pose maintenant

$$\mu_i^n = \frac{\theta \Delta t}{\rho_i^n \Delta x}, \tag{3.100}$$

et on introduit les résidus

$$\begin{aligned} (\mathcal{R}_\tau)_i^n &= \frac{1}{\rho_i^n \Delta x}(v^{n,*}_{i+1/2} - v^{n,*}_{i-1/2}), \\ (\mathcal{R}_v)_i^n &= -\frac{1}{\rho_i^n \Delta x}(\Pi^{n,*}_{i+1/2} - \Pi^{n,*}_{i-1/2}), \\ (\mathcal{R}_Y)_i^n &= \frac{1}{\rho_i^n \Delta x}(\Sigma^{n,*}_{i+1/2} - \Sigma^{n,*}_{i-1/2}). \end{aligned} \tag{3.101}$$

Le système (3.99) se réécrit, pour $i \in \{1, \ldots, N\}$

$$\begin{aligned} \delta\tau_i - \mu_i\Big(\frac{\delta v_{i+1} - \delta v_{i-1}}{2} - \frac{\delta\Pi_{i+1} - 2\delta\Pi_i + \delta\Pi_{i-1}}{2a}\Big) &= \Delta t (\mathcal{R}_\tau)_i^n, \\ \delta v_i + \mu_i\Big(\frac{\delta\Pi_{i+1} - \delta\Pi_{i-1}}{2} - a\,\frac{\delta v_{i+1} - 2\delta v_i + \delta v_{i-1}}{2}\Big) &= \Delta t (\mathcal{R}_v)_i^n, \\ \delta Y_i &= \Delta t (\mathcal{R}_Y)_i^n. \end{aligned} \tag{3.102}$$

Les résidus « explicites » $(\mathcal{R}_\tau)_i^n$, $(\mathcal{R}_Y)_i^n$ et $(\mathcal{R}_v)_i^n$ sont connus car ils proviennent des termes explicites à l'instant t^n. En conséquence, on peut calculer immédiatement les incréments δY_i dès que Δt est connu. Il ne nous reste qu'à évaluer les $\delta\tau_i$ et δv_i à l'intérieur et aux bords du domaine.

Simplifier le système (3.102) en utilisant l'implicitation de l'équilibre en Π (3.88) nous conduit à construire un système linéaire tridiagonal par blocs. L'avantage de cette simplification est que la taille des blocs est petite (2×2) et le nouveau système est très facile à résoudre.

3.5.2 Construction du système linéaire par blocs à l'intérieur du domaine

Afin de proposer une méthode de résolution pour le système discrétisé (3.102), nous allons remplacer les incréments $\delta\Pi_i$ par une fonction de $\delta\tau_i$ et δv_i. Par définition de $\delta\Pi_i$ et à partir de l'approximation linéarisée de $\widetilde{\Pi}_i^n$ définie dans (3.88), nous avons

$$\begin{aligned} \delta\Pi_i &= \widetilde{\Pi}_i^n - \Pi_i^n \\ &= P_i^n + (\partial_\tau P)_i^n \delta\tau_i + (\partial_Y P)_i^n \delta Y_i + (\partial_v P)_i^n \delta v_i - \Pi_i^n. \end{aligned} \tag{3.103}$$

Or, la variable de relaxation Π_i^n coïncide avec la pression P_i^n au temps t^n. Il en résulte que

$$\delta\Pi_i = (\partial_\tau P)_i^n \delta\tau_i + (\partial_Y P)_i^n \delta Y_i + (\partial_v P)_i^n \delta v_i. \tag{3.104}$$

Rappelons que l'incrément δY_i est explicitement connu grâce à la troisième équation de (3.102), nous ne considérons donc qu'un système composé de deux équations sur $\delta\tau_i$ et δv_i. En effet, remplacer (3.104) dans les deux premières équations de (3.102) nous permet d'obtenir un système de deux équations avec deux inconnues $\delta\tau_i$ et δv_i, à savoir pour $i \in \{2, \ldots, N-1\}$

$$\begin{aligned}
&\delta\tau_i - \mu_i \frac{\delta v_{i+1} - \delta v_{i-1}}{2} + \mu_i \frac{(\partial_\tau P)_{i+1}\delta\tau_{i+1} - 2(\partial_\tau P)_i \delta\tau_i + (\partial_\tau P)_{i-1}\delta\tau_{i-1}}{2a} \\
&\quad + \mu_i \frac{(\partial_v P)_{i+1}\delta v_{i+1} - 2(\partial_v P)_i \delta v_i + (\partial_v P)_{i-1}\delta v_{i-1}}{2a} &&= (\widetilde{\mathcal{R}}_\tau)_i^n, \\
&\delta v_i + \mu_i \frac{(\partial_\tau P)_{i+1}\delta\tau_{i+1} - (\partial_\tau P)_{i-1}\delta\tau_{i-1}}{2} - \mu_i a \frac{\delta v_{i+1} - 2\delta v_i + \delta v_{i-1}}{2} \\
&\quad + \mu_i \frac{(\partial_v P)_{i+1}\delta v_{i+1} - (\partial_v P)_{i-1}\delta v_{i-1}}{2} &&= (\widetilde{\mathcal{R}}_v)_i^n.
\end{aligned} \tag{3.105}$$

avec

$$\begin{aligned}
(\widetilde{\mathcal{R}}_\tau)_i^n &= \Delta t (\mathcal{R}_\tau)_i^n - \mu_i \frac{(\partial_Y P)_{i+1}\delta Y_{i+1} - 2(\partial_Y P)_i \delta Y_i + (\partial_Y P)_{i-1}\delta Y_{i-1}}{2a}, \\
(\widetilde{\mathcal{R}}_v)_i^n &= \Delta t (\mathcal{R}_v)_i^n - \mu_i \frac{(\partial_Y P)_{i+1}\delta Y_{i+1} - (\partial_Y P)_{i-1}\delta Y_{i-1}}{2}.
\end{aligned} \tag{3.106}$$

Le système (3.105) peut s'écrire maintenant sous forme matricielle. Pour toutes les mailles intérieures, on a

$$\left[\quad \mathbf{C}_{i-1} \quad | \quad \mathbf{D}_i \quad | \quad \mathbf{E}_{i+1} \quad \right] . \left[\begin{array}{c} \delta\mathbf{u}_{i-1} \\ \hline \delta\mathbf{u}_i \\ \hline \delta\mathbf{u}_{i+1} \end{array} \right] = \left[\begin{array}{c} (\widetilde{\mathcal{R}}_\tau)_i^n \\ (\widetilde{\mathcal{R}}_v)_i^n \end{array} \right], \tag{3.107}$$

avec

$$\mathbf{C}_{i-1} = \left[\begin{array}{cc} \mu_i \dfrac{(\partial_\tau P)_{i-1}}{2a} & \dfrac{\mu_i}{2}(1 + \dfrac{(\partial_v P)_{i-1}}{a}) \\ -\mu_i \dfrac{(\partial_\tau P)_{i-1}}{2} & -\dfrac{\mu_i}{2}(a + (\partial_v P)_{i-1}) \end{array} \right], \quad \mathbf{D}_i = \left[\begin{array}{cc} 1 - \mu_i \dfrac{(\partial_\tau P)_i}{a} & -\mu_i \dfrac{(\partial_v P)_i}{a} \\ 0 & 1 + a\mu_i \end{array} \right],$$

$$\mathbf{E}_{i+1} = \left[\begin{array}{cc} \mu_i \dfrac{(\partial_\tau P)_{i+1}}{2a} & -\dfrac{\mu_i}{2}(1 - \dfrac{(\partial_v P)_{i+1}}{a}) \\ \mu_i \dfrac{(\partial_\tau P)_{i+1}}{2} & -\dfrac{\mu_i}{2}(a - (\partial_v P)_{i+1}) \end{array} \right], \quad \text{et} \quad \delta\mathbf{u}_i = \left[\begin{array}{c} \delta\tau_i \\ \delta v_i \end{array} \right]. \tag{3.108}$$

Ces équations sont valides pour toutes les mailles à l'intérieur du domaine, *i.e.* pour $i \in \{2, \ldots, N-1\}$. Pour les mailles aux bords, nous avons d'autres relations à cause de la prise en compte des conditions aux limites. Nous allons maintenant discuter le traitement des conditions aux limites au niveau discret.

3.5.3 Traitement aux limites

Les idées pour le traitement des conditions aux limites dans le contexte d'écoulements pétroliers ont été expliquées dans les sections 2.4.2 et 3.2.4. Ici, nous suivons la même démarche.

Pour le schéma de relaxation Lagrange-Projection en semi-implicite, les conditions aux limites pour la phase Lagrange sont traitées de la même manière qu'en explicite (voir 3.2.4). En revanche, la mise à jour des états fictifs pour la phase Projection est différente. Nous allons étudier maintenant comment calculer ces états fictifs.

3.5.3.1 À l'amont

En supprimant trois ondes comme en explicite (voir section 3.2.4.1), et en imposant les débits $\widehat{q_G}$ et $\widehat{q_T}$ à l'instant t^{n+1} (au lieu de l'instant t^n dans le cas explicite), nous avons

$$\begin{aligned} \widetilde{\rho}_0\widetilde{v}^{*,\theta}_{1/2} &= \widehat{q_T}, \\ \widetilde{\rho}_0\widetilde{Y}_0\widetilde{v}^{*,\theta}_{1/2} - \Sigma^*_{1/2} &= \widehat{q_G}, \\ \widetilde{\Pi}_0 - a\widetilde{v}_0 &= \widetilde{\Pi}_1 - a\widetilde{v}_1, \\ \widetilde{\Sigma}_0 + b\widetilde{Y}_0 &= \widetilde{\Sigma}_1 + b\widetilde{Y}_1, \\ \widetilde{\Pi}_0 + a^2\widetilde{\tau}_0 &= \widetilde{\Pi}_1 + a^2\widetilde{\tau}_1. \end{aligned} \tag{3.109}$$

Fig. 3.10. Les lignes en pointillé représentent les ondes à annuler à l'amont.

Pour trouver les valeurs des inconnues $(\widetilde{\rho}, \widetilde{Y}, \widetilde{v}, \widetilde{\Pi}, \widetilde{\Sigma})_0$, il suffit de projeter ces relations dans le système (3.102) afin de compléter le système linéaire par blocs. La première équation nous donne $\widetilde{v}^{*,\theta}_{1/2} = \widehat{q_T}\widetilde{\tau}_0$ et par linéarisation de $\widetilde{v}^{*,\theta}_{1/2}$ on obtient

$$\theta\delta v^*_{1/2} = q_T\delta\tau_0 + \tau_0\delta q_T. \tag{3.110}$$

Ainsi, au lieu de calculer directement les valeurs des variables de $\widetilde{\mathbb{V}}_0$ nous allons calculer leurs incréments, c'est-à-dire $\delta\mathbb{V}_0 = (\delta\tau, \delta Y, \delta v, \delta\Pi, \delta\Sigma)^T_0$. Les incréments $\delta\Pi_0$ et $\delta\Sigma_0$ ne nous intéressent pas explicitement car on n'en a pas besoin pour évaluer la solution $\widehat{\mathbb{V}}$ (solution issue de la phase Projection) à l'instant t^{n+1}.
L'incrément $\delta v^*_{1/2}$ correspond à la solution du problème de Riemann à l'interface 1/2 et est donnée par

$$\begin{aligned} \delta v^*_{1/2} &= \frac{\delta v_0 + \delta v_1}{2} - \frac{\delta\Pi_1 - \delta\Pi_0}{2a} \\ &= \delta v_0 + \frac{-\delta v_0 + \delta v_1}{2} - \frac{\delta\Pi_1 - \delta\Pi_0}{2a} \\ &= \delta v_0 + \frac{(\delta\Pi_0 - a\delta v_0) - (\delta\Pi_1 - a\delta v_1)}{2a}. \end{aligned} \tag{3.111}$$

La linéarisation de la troisième équation de (3.109) donne $\delta\Pi_0 - a\delta v_0 = \delta\Pi_1 - a\delta v_1$ d'où

$$\delta v^*_{1/2} = \delta v_0. \tag{3.112}$$

La condition (3.110) devient

$$q_T \delta\tau_0 - \theta \delta v_0 = -\tau_0 \delta q_T. \tag{3.113}$$

Par ailleurs, en éliminant $\delta\Pi_0$ dans la première et la troisième équation de (3.109) nous obtenons une autre relation entre $\delta\tau_0$ et δv_0

$$a\delta\tau_0 + \delta v_0 = a\delta\tau_1 + \delta v_1. \tag{3.114}$$

Pour l'état fictif à l'amont, on a enfin

$$\left[\begin{array}{cc|cc} q_T & -\theta & 0 & 0 \\ a & 1 & -a & -1 \end{array}\right] . \left[\begin{array}{c} \delta\mathbf{u}_0 \\ \hline \delta\mathbf{u}_1 \end{array}\right] = \left[\begin{array}{c} (\widetilde{\mathcal{R}}_\tau)_0^n \\ (\widetilde{\mathcal{R}}_v)_0^n \end{array}\right], \tag{3.115}$$

avec les résidus

$$(\widetilde{\mathcal{R}}_\tau)_0^n = -\tau_0 \delta q_T \quad \text{et} \quad (\widetilde{\mathcal{R}}_v)_0^n = 0. \tag{3.116}$$

On remarque maintenant que le système (3.107) décrit une relation naturelle entre les états « vraiment » intérieurs ($i \in \{2, \ldots, N-1\}$) parce qu'elle ne concerne pas la maille fictive. Pour $i = 1$, comme il y a une condition aux limites imposée sur l'état $\mathbb{V}_0$ en plus, cette relation ne sera pas la même. Il faudra donc intégrer la contribution de $\delta\Pi_0$ (due à la condition aux limites) dans (3.102). Par simple linéarisation de $\delta\Pi_1$ dans la dernière équation de (3.109), on a

$$\begin{aligned} \delta\Pi_0 &= -a^2\delta\tau_0 + (\delta\Pi_1 + a^2\delta\tau_1) \\ &= -a^2\delta\tau_0 + \left[(\partial_\tau P)_1\delta\tau_1 + (\partial_Y P)_1\delta Y_1 + (\partial_v P)_1\delta v_1) + a^2\delta\tau_1\right] \\ &= -a^2\delta\tau_0 + \left[(\partial_\tau P)_1 + a^2\right]\delta\tau_1 + (\partial_v P)_1\delta v_1 + (\partial_Y P)_1\delta Y_1. \end{aligned} \tag{3.117}$$

En injectant cet incrément dans (3.102), on obtient, pour $i = 1$,

$$\begin{aligned} \delta\tau_1 - \mu_1\frac{\delta v_2 - \delta v_0}{2} + \mu_1\frac{(\partial_\tau P)_2\delta\tau_2 + (a^2 - (\partial_\tau P)_1)\delta\tau_1 - a^2\delta\tau_0}{2a} & \\ + \mu_1\frac{(\partial_v P)_2\delta v_2 - (\partial_v P)_1\delta v_1}{2a} &= (\widetilde{\mathcal{R}}_\tau)_1^n, \\ \delta v_1 + \mu_1\frac{(\partial_\tau P)_2\delta\tau_2 + a^2\delta\tau_0 - [(\partial_\tau P)_1 + a^2]\delta\tau_1}{2} - \mu_1 a\frac{\delta v_2 - 2\delta v_1 + \delta v_0}{2} & \\ + \mu_1\frac{(\partial_v P)_2\delta v_2 - (\partial_v P)_1\delta v_1}{2} &= (\widetilde{\mathcal{R}}_v)_1^n. \end{aligned} \tag{3.118}$$

Le second membre de (3.118) doit être modifié aussi à cause de la contribution de δY_1 induite par la linéarisation de $\delta\Pi_1$

$$\begin{aligned} (\widetilde{\mathcal{R}}_\tau)_1^n &= \Delta t(\widetilde{\mathcal{R}}_\tau)_1^n - \mu_1\frac{(\partial_Y P)_2\delta Y_2 - (\partial_Y P)_1\delta Y_1}{2a}, \\ (\widetilde{\mathcal{R}}_v)_1^n &= \Delta t(\widetilde{\mathcal{R}}_v)_1^n - \mu_1\frac{(\partial_Y P)_2\delta Y_2 - (\partial_Y P)_1\delta Y_1}{2}. \end{aligned} \tag{3.119}$$

On peut écrire le système (3.118) sous forme matricielle

$$\left[\begin{array}{c|c|c} \mathbf{C}_0 & \mathbf{D}_1 & \mathbf{E}_2 \end{array}\right] . \left[\begin{array}{c} \delta\mathbf{u}_0 \\ \hline \delta\mathbf{u}_1 \\ \hline \delta\mathbf{u}_2 \end{array}\right] = \left[\begin{array}{c} (\widetilde{\mathcal{R}}_\tau)_1^n \\ (\widetilde{\mathcal{R}}_v)_1^n \end{array}\right], \tag{3.120}$$

avec

$$\mathbf{C}_0 = \begin{bmatrix} -\mu_1\frac{a}{2} & \frac{\mu_1}{2} \\ \mu_1\frac{a^2}{2} & -\mu_1\frac{a}{2} \end{bmatrix}, \quad \mathbf{D}_1 = \begin{bmatrix} 1+\mu_1\frac{a^2-(\partial_\tau P)_1}{2a} & -\mu_1\frac{(\partial_v P)_1}{2a} \\ -\mu_1\frac{a^2+(\partial_\tau P)_1}{2} & 1+\mu_1(a-\frac{(\partial_v P)_1}{2}) \end{bmatrix},$$
$$\mathbf{E}_2 = \begin{bmatrix} \mu_1\frac{(\partial_\tau P)_2}{2a} & -\frac{\mu_1}{2}(1-\frac{(\partial_v P)_2}{a}) \\ \mu_1\frac{(\partial_\tau P)_2}{2} & -\frac{\mu_1}{2}(a-(\partial_v P)_2) \end{bmatrix}. \tag{3.121}$$

Une fois le système complet construit, on le résout pour obtenir l'incrément $\delta\tau_0$ et on en déduit $\widetilde{\rho}_0$. À partir de la deuxième équation de (3.109) la fraction massique $\widetilde{Y}_0$ de l'état fictif après la phase Lagrange est donnée par

$$\widetilde{Y}_0 = \frac{\widehat{q_T} + \Sigma^*_{1/2}}{\widetilde{\rho}_0 \widetilde{v}^{*,\theta}_{1/2}}. \tag{3.122}$$

***Remarque** 3.2.* Dans (3.117), nous n'avons pas utilisé directement la linéarisation de $\delta\Pi_0$ mais celle de $\delta\Pi_1$ correspondant à la solution dans la première maille à l'intérieur du domaine. Cette technique nous permet de prendre en compte la condition limite imposée en l'intégrant dans le système de départ. La matrice ainsi que le second membre doivent donc être modifiés suivant (3.119).

3.5.3.2 À l'aval

Fig. 3.11. Les lignes en pointillé représentent les ondes à annuler à l'aval.

Les conditions aux limites imposées à l'aval reposent sur la pression p et la fraction massique Y_{N+1}. La suppression de trois ondes « sortantes » du domaine nous conduit à

$$\begin{aligned} \widetilde{\Pi}_{N+1} + a\widetilde{v}_{N+1} &= \widetilde{\Pi}_N + a\widetilde{v}_N, \\ \widetilde{\Pi}_{N+1} + a^2\widetilde{\tau}_{N+1} &= \widetilde{\Pi}_N + a^2\widetilde{\tau}_N, \\ \widetilde{\Pi}_{N+1} &= \widehat{p} \quad + \widetilde{\Sigma}_{N+1}\Phi(\widehat{p}, Y_{N+1}, \widetilde{v}_{N+1}), \\ \widetilde{\Sigma}_{N+1} - b\widetilde{Y}_{N+1} &= \widetilde{\Sigma}_N - b\widetilde{Y}_N, \end{aligned} \tag{3.123}$$

avec la fraction massique du gaz $\widetilde{Y}_{N+1} = \widetilde{Y}_N$. La pression $\widehat{p}$ est prise à l'instant t^{n+1}.

La difficulté pour la résolution réside ici dans la non-linéarité de la loi de glissement $\phi(\widehat{p}, Y, v)$. Nous ne pouvons pas résoudre le système précédent par un calcul simple. Ceci nous oblige encore à travailler avec les incréments (ce qui est cohérent avec ce que l'on a fait précédemment). Nous allons premièrement calculer $\delta\Pi_{N+1}$ par deux méthodes différentes :

- par la linéarisation de la troisième équation de (3.123)

$$\delta\Pi_{N+1} = \delta p + \Sigma_{N+1}\frac{\partial\phi}{\partial v}\Big|_{p,Y}\delta v_{N+1} + \phi\delta\Sigma_{N+1}. \tag{3.124}$$

- par le remplacement de la linéarisation de $\delta\Pi_N$ dans la deuxième équation de (3.123)

$$\begin{aligned}\delta\Pi_{N+1} &= -a^2\delta\tau_{N+1} + (\delta\Pi_N + a^2\delta\tau_N)\\ &= -a^2\delta\tau_{N+1} + \left[(\partial_\tau P)_N\delta\tau_N + (\partial_Y P)_N\delta Y_N + (\partial_v P)_N\delta v_N + a^2\delta\tau_N\right]\\ &= -a^2\delta\tau_{N+1} + \left[(\partial_\tau P)_N + a^2\right]\delta\tau_N + (\partial_v P)_N\delta v_N + (\partial_Y P)_N\delta Y_N.\end{aligned} \tag{3.125}$$

L'égalité de (3.124) et (3.125) donne

$$-\left[(\partial_\tau P)_N + a^2\right]\delta\tau_N - (\partial_v P)_N\delta v_N + a^2\delta\tau_{N+1} + \Sigma_{N+1}\frac{\partial\phi}{\partial v}\Big|_{p,Y}\delta v_{N+1} = (\widetilde{\mathcal{R}}_v)^n_{N+1}, \tag{3.126}$$

avec

$$(\widetilde{\mathcal{R}}_v)^n_{N+1} = -\delta p - \phi\delta\Sigma_{N+1} + (\partial_Y P)_N\delta Y_N. \tag{3.127}$$

Par ailleurs en éliminant $\delta\Pi_{N+1}$ dans les deux premières équations de (3.123), on a

$$a\delta\tau_N - \delta v_N = a\delta\tau_{N+1} - \delta v_{N+1}. \tag{3.128}$$

En écrivant sous forme matricielle les équations (3.127) et (3.128) on a

$$\left[\begin{array}{cc|cc} a & -1 & -a & 1\\ -\left[(\partial_\tau P)_N + a^2\right] & -(\partial_v P)_N & a^2 & \Sigma_{N+1}\frac{\partial\phi}{\partial v}\big|_{p,Y}\end{array}\right]\cdot\left[\begin{array}{c}\delta\mathbf{u}_N\\ \hline \delta\mathbf{u}_{N+1}\end{array}\right] = \left[\begin{array}{c}(\widetilde{\mathcal{R}}_\tau)^n_{N+1}\\ (\widetilde{\mathcal{R}}_v)^n_{N+1}\end{array}\right], \tag{3.129}$$

avec

$$(\widetilde{\mathcal{R}}_\tau)^n_{N+1} = 0. \tag{3.130}$$

Il ne nous reste qu'à intégrer la condition limite dans la relation de l'état fictif avec les états intérieurs, nous allons donc injecter (3.125) dans (3.102). Pour $i = N$, après quelques transformations on aboutit à

$$\begin{aligned}&\delta\tau_N - \mu_N\frac{\delta v_{N+1} - \delta v_{N-1}}{2} + \mu_N\frac{-a^2\delta\tau_{N+1} + \left[a^2 - (\partial_\tau P)_N\right]\delta\tau_N + (\partial_\tau P)_{N-1}\delta\tau_{N-1}}{2a}\\ &\quad + \mu_N\frac{-(\partial_v P)_N\delta v_N + (\partial_v P)_{N-1}\delta v_{N-1}}{2a} = (\widetilde{\mathcal{R}}_\tau)^n_N,\\ &\delta v_N + \mu_N\Big(\frac{-a^2\delta\tau_{N+1} + \left[a^2 + (\partial_\tau P)_N\right]\delta\tau_N - (\partial_\tau P)_{N-1}\delta\tau_{N-1}}{2} - a\frac{\delta v_{N+1} - 2\delta v_N + \delta v_{N-1}}{2}\Big)\\ &\quad + \mu_N\frac{(\partial_v P)_N\delta v_N - (\partial_v P)_{N-1}\delta v_{N-1}}{2} = (\widetilde{\mathcal{R}}_v)^n_N,\end{aligned} \tag{3.131}$$

avec les seconds membres modifiés

$$\begin{aligned}(\widetilde{\mathcal{R}}_\tau)^n_N &= \Delta t(\mathcal{R}_\tau)^n_N - \mu_N\frac{-(\partial_Y P)_N\delta Y_N - (\partial_Y P)_{N-1}\delta Y_{N-1}}{2a},\\ (\widetilde{\mathcal{R}}_v)^n_N &= \Delta t(\mathcal{R}_v)^n_N - \mu_N\frac{(\partial_Y P)_N\delta Y_N - (\partial_Y P)_{N-1}\delta Y_{N-1}}{2}.\end{aligned} \tag{3.132}$$

Sous forme matricielle, le système (3.132) se traduit par

$$\left[\begin{array}{c|c|c}\mathbf{C}_{N-1} & \mathbf{D}_N & \mathbf{E}_{N+1}\end{array}\right]\cdot\left[\begin{array}{c}\delta\mathbf{u}_{N-1}\\ \hline \delta\mathbf{u}_N\\ \hline \delta\mathbf{u}_{N+1}\end{array}\right] = \left[\begin{array}{c}(\widetilde{\mathcal{R}}_\tau)^n_N\\ (\widetilde{\mathcal{R}}_v)^n_N\end{array}\right], \tag{3.133}$$

avec

$$\mathbf{C}_{N-1} = \begin{bmatrix} -\mu_N \dfrac{(\partial_\tau P)_{N-1}}{2a} & \dfrac{\mu_N}{2}(1+\dfrac{(\partial_v P)_{N-1}}{a}) \\ -\mu_N \dfrac{(\partial_\tau P)_{N-1}}{2} & -\dfrac{\mu_N}{2}(a+\dfrac{(\partial_v P)_{N-1}}{2}) \end{bmatrix},$$

$$\mathbf{D}_N = \begin{bmatrix} 1+\mu_N \dfrac{a^2-(\partial_\tau P)_N}{2a} & -\mu_N \dfrac{(\partial_v P)_N}{2a} \\ \mu_N \dfrac{a^2+(\partial_\tau P)_N}{2} & 1+\mu_N(a+\dfrac{(\partial_v P)_N}{2}) \end{bmatrix}, \tag{3.134}$$

$$\mathbf{E}_{N+1} = \begin{bmatrix} -\mu_N \dfrac{a}{2} & -\dfrac{\mu_N}{2} \\ -\mu_N \dfrac{a^2}{2} & -\mu_N \dfrac{a}{2} \end{bmatrix}.$$

3.5.4 Système par blocs complet et résolution

Grâce à la simplification du système (3.102) et les relations liées aux conditions limites imposées, nous avons maintenant toutes les équations nécessaires pour construire un système linéaire dont la variable est $\delta\mathbf{u}_i = (\delta\tau_i, \delta v_i)^T$ pour $i \in \{0,\dots,N+1\}$. En fait, en combinant (3.107), (3.115), (3.120), (3.129) et (3.133), nous obtenons

$$\begin{bmatrix} \mathbf{D}_0 & \mathbf{E}_1 & 0 & \cdots & \cdots & \cdots & \cdots & 0 \\ \mathbf{C}_0 & \mathbf{D}_1 & \mathbf{E}_2 & \ddots & & & & \vdots \\ 0 & \mathbf{C}_1 & \mathbf{D}_2 & \mathbf{E}_3 & \ddots & & & \vdots \\ \vdots & \ddots & \ddots & \ddots & \ddots & \ddots & & \vdots \\ \vdots & & \ddots & \ddots & \ddots & \ddots & \ddots & \vdots \\ \vdots & & & \ddots & \mathbf{C}_{N-2} & \mathbf{D}_{N-1} & \mathbf{E}_N & 0 \\ \vdots & & & & \ddots & \mathbf{C}_{N-1} & \mathbf{D}_N & \mathbf{E}_{N+1} \\ 0 & \cdots & \cdots & \cdots & \cdots & 0 & \mathbf{C}_N & \mathbf{D}_{N+1} \end{bmatrix} \cdot \begin{bmatrix} \delta\mathbf{u}_0 \\ \delta\mathbf{u}_1 \\ \delta\mathbf{u}_2 \\ \cdots \\ \cdots \\ \delta\mathbf{u}_{N-1} \\ \delta\mathbf{u}_N \\ \delta\mathbf{u}_{N+1} \end{bmatrix} = \begin{bmatrix} \mathbf{F}_0 \\ \mathbf{F}_1 \\ \mathbf{F}_2 \\ \cdots \\ \cdots \\ \mathbf{F}_{N-1} \\ \mathbf{F}_N \\ \mathbf{F}_{N+1} \end{bmatrix}, \tag{3.135}$$

avec

$$\mathbf{C}_i = \begin{bmatrix} \mu_{i+1} \dfrac{(\partial_\tau P)_i}{2a} & \dfrac{\mu_{i+1}}{2}(1+\dfrac{(\partial_v P)_i}{a}) \\ -\mu_{i+1} \dfrac{(\partial_\tau P)_i}{2} & -\dfrac{\mu_{i+1}}{2}(a+(\partial_v P)_i) \end{bmatrix} \text{ pour } i \in \{1,\dots,N-1\},$$

$$\mathbf{D}_i = \begin{bmatrix} 1-\mu_i \dfrac{(\partial_\tau P)_i}{a} & -\mu_i \dfrac{(\partial_v P)_i}{a} \\ 0 & 1+a\mu_i \end{bmatrix} \quad \text{pour } i \in \{2,\dots,N-1\}, \tag{3.136}$$

$$\mathbf{E}_i = \begin{bmatrix} \mu_{i-1} \dfrac{(\partial_\tau P)_i}{2a} & -\dfrac{\mu_{i-1}}{2}(1-\dfrac{(\partial_v P)_i}{a}) \\ \mu_{i-1} \dfrac{(\partial_\tau P)_i}{2} & -\dfrac{\mu_{i-1}}{2}(a-(\partial_v P)_i) \end{bmatrix} \text{ pour } i \in \{2,\dots,N\},$$

et

$$
\begin{aligned}
\mathbf{D}_0 &= \begin{bmatrix} q_T & -\theta \\ a & 1 \end{bmatrix}, \quad \mathbf{E}_1 = \begin{bmatrix} 0 & 0 \\ -a & -1 \end{bmatrix}, \\
\mathbf{C}_0 &= \begin{bmatrix} -\mu_1 \frac{a}{2} & \frac{\mu_1}{2} \\ \mu_1 \frac{a^2}{2} & -\mu_1 \frac{a}{2} \end{bmatrix}, \quad \mathbf{D}_1 = \begin{bmatrix} 1+\mu_1 \frac{a^2-(\partial_\tau P)_1}{2a} & -\mu_1 \frac{(\partial_v P)_1}{2a} \\ -\mu_1 \frac{a^2+(\partial_\tau P)_1}{2} & 1+\mu_1(a-\frac{(\partial_v P)_1}{2}) \end{bmatrix}, \\
\mathbf{D}_N &= \begin{bmatrix} 1+\mu_N \frac{a^2-(\partial_\tau P)_N}{2a} & -\mu_N \frac{(\partial_v P)_N}{2a} \\ \mu_N \frac{a^2+(\partial_\tau P)_N}{2} & 1+\mu_N(a+\frac{(\partial_v P)_N}{2}) \end{bmatrix}, \quad \mathbf{E}_{N+1} = \begin{bmatrix} -\mu_N \frac{a}{2} & -\frac{\mu_N}{2} \\ -\mu_N \frac{a^2}{2} & -\mu_N \frac{a}{2} \end{bmatrix}, \\
\mathbf{C}_N &= \begin{bmatrix} a & -1 \\ -\left[(\partial_\tau P)_N + a^2\right] & -(\partial_v P)_N \end{bmatrix}, \quad \mathbf{D}_{N+1} = \begin{bmatrix} -a & 1 \\ a^2 & \Sigma_{N+1} \frac{\partial \phi}{\partial v}\big|_{p,Y} \end{bmatrix}.
\end{aligned}
\tag{3.137}
$$

Les $\mathbf{F}_i$ sont tout simplement les vecteurs des résidus $((\widetilde{\mathcal{R}}_\tau)_i^n, (\widetilde{\mathcal{R}}_v)_i^n)^T$ que l'on a définis dans (3.106), (3.116), (3.119), (3.127) et (3.132). Le système (3.135) est un système linéaire tri-diagonal par blocs de taille 2×2. On peut le résoudre en utilisant une factorisation LU par blocs.

***Remarque** 3.3.* Notons que dans la pratique, les dérivées $(\partial_v P)_i$ sont négligeables devant le coefficient de relaxation a^2. Par conséquent, elles ne sont pas implémentées afin d'alléger le code. Nous les écrivons ici dans le but d'établir le système complet.

***Remarque** 3.4.* En comparaison avec l'ancienne méthode de résolution présentée dans [5,6], qui doit résoudre un système par blocs de taille 5×5, nous avons ici un système par blocs de taille plus petite 2×2. Nous espérons donc accélérer le temps de calcul, qui sera justifié par la suite.

***Remarque** 3.5.* Pour le calcul de pas de temps (voir section suivante), il est pratique d'écrire les secondes membres $((\widetilde{\mathcal{R}}_\tau)_i^n, (\widetilde{\mathcal{R}}_v)_i^n)^T$, $i \in \{1, .., N\}$ définis dans (3.106), (3.119) et (3.132) sous la forme suivante

$$
\begin{aligned}
(\widetilde{\mathcal{R}}_\tau)_i^n &= \Delta t (\mathcal{R}_\tau^{(1)})_i^n - (\Delta t)^2 (\mathcal{R}_\tau^{(2)})_i^n, \\
(\widetilde{\mathcal{R}}_v)_i^n &= \Delta t (\mathcal{R}_v^{(1)})_i^n - (\Delta t)^2 (\mathcal{R}_v^{(2)})_i^n,
\end{aligned}
\tag{3.138}
$$

avec

$$
(\mathcal{R}_\tau^{(1)})_i^n = (\mathcal{R}_\tau)_i^n \qquad \text{et} \qquad (\mathcal{R}_v^{(1)})_i^n = (\mathcal{R}_v)_i^n, \quad i \in \{1, .., N\},
\tag{3.139}
$$

et

$$
\begin{aligned}
(\mathcal{R}_\tau^{(2)})_1^n &= (\mathcal{R}_v^{(2)})_1^n = \frac{\theta}{\rho_1 \Delta x} \frac{(\partial_Y P)_2 (\mathcal{R}_Y)_2^n - (\partial_Y P)_1 (\mathcal{R}_Y)_1^n}{2a}, \\
(\mathcal{R}_\tau^{(2)})_N^n &= \frac{\theta}{\rho_N \Delta x} \frac{-(\partial_Y P)_N (\mathcal{R}_Y)_N^n - (\partial_Y P)_{N-1} (\mathcal{R}_Y)_{N-1}^n}{2a}, \\
(\mathcal{R}_v^{(2)})_N^n &= \frac{\theta}{\rho_N \Delta x} \frac{(\partial_Y P)_N (\mathcal{R}_Y)_N^n - (\partial_Y P)_{N-1} (\mathcal{R}_Y)_{N-1}^n}{2a},
\end{aligned}
\tag{3.140}
$$

et pour $i \in [2,..,N-1]$

$$
\begin{aligned}
(\mathcal{R}_\tau^{(2)})_i^n &= \frac{\theta}{\rho_i \Delta x} \frac{(\partial_Y P)_{i+1}(\mathcal{R}_Y)_{i+1}^n - 2(\partial_Y P)_i(\mathcal{R}_Y)_i^n + (\partial_Y P)_{i-1}(\mathcal{R}_Y)_{i-1}^n}{2a}, \\
(\mathcal{R}_v^{(2)})_i^n &= \frac{\theta}{\rho_i \Delta x} \frac{(\partial_Y P)_{i+1}(\mathcal{R}_Y)_{i+1}^n - (\partial_Y P)_{i-1}(\mathcal{R}_Y)_{i-1}^n}{2}.
\end{aligned}
\tag{3.141}
$$

Ces formules sont obtenues en remplaçant les δY_i^n par leurs expressions originales (troisième équation de (3.102)) et en séparant ce qui est lié à Δt et à $(\Delta t)^2$.

3.5.5 Condition de positivité et principe du maximum

Comme en explicite, la positivité de ρ (plus précisément de $\widetilde{\rho}_i^n$) à chaque phase est assurée si la condition

$$
1 + \frac{\Delta t}{\Delta x}\left[(\widetilde{v}_{i+1/2}^{n,*})^- - (\widetilde{v}_{i-1/2}^{n,*})^+\right] > 0, \tag{3.142}
$$

que nous avons déjà rencontrée en (3.45), est satisfaite.

Une condition suffisante pour avoir (3.142) est

$$
\frac{\Delta t}{\Delta x} \max_i |\widetilde{v}_{i+1/2}^{n,*}| < \frac{1}{2}. \tag{3.143}
$$

En semi-implicite, la valeur de $\widetilde{v}_{i+1/2}^{n,*}$ n'est pas connue à l'instant t^n. Néammoins, nous allons voir qu'il est possible de donner une condition sur Δt de manière à satisfaire (3.143).

Finalement, pour assurer la positivité de ρ et le principe du maximum sur Y, il faut que

$$
\frac{\Delta t}{\Delta x} \max_i \max\left(|\widetilde{v}_{i+1/2}^{n,*}|, b_{i+1/2}^n\right) < \frac{1}{2}. \tag{3.144}
$$

Théorème *3.6. Le schéma résultant de l'enchaînement des deux étapes Lagrange-Projection se présente sous forme conservative. Il est consistant et positif au sens*

$$
\rho_i^{n+1} > 0 \quad \text{et} \quad Y_i^{n+1} \in [0,1] \tag{3.145}
$$

dès que

$$
\frac{\Delta t}{\Delta x} < \frac{1}{\mathcal{B}(\mathbb{U}^n, CL^n, \partial_t CL^n)} \tag{3.146}
$$

où la borne $\mathcal{B}(\mathbb{U}^n, CL^n, \partial_t CL^n)$ est explicitement calculable à l'instant t^n et correspond à une vitesse lente.

Démonstration La démonstration se référence aux calculs dans la section 3.5.6 suivante. □

3.5.6 Calcul de pas de temps et condition de positivité

Une nouveauté dans la méthode Lagrange-Projection est l'approximation a priori du pas de temps Δt. Ce dernier est calculé d'une manière totalement différente des estimations habituelles qui reposent directement sur les normes globales. Ici, nous allons utiliser une représentation locale explicite de la solution du système linéaire.

Comme on l'a déjà vu, pour assurer la positivité de la densité dans le cas d'un schéma L-P semi-implicite on doit avoir

$$1 - 2\frac{\Delta t}{\Delta x} \max_i |\widetilde{v}^{*,\theta}_{i+1/2}| > 0. \tag{3.147}$$

D'après notre construction implicite nous avons

$$\widetilde{v}^{*,\theta}_{i+1/2} = v^*_{i+1/2} + \theta \delta v^*_{i+1/2},$$

d'où

$$|\widetilde{v}^{*,\theta}_{i+1/2}| \leq |v^*_{i+1/2}| + \theta |\delta v^*_{i+1/2}|. \tag{3.148}$$

Il suffira donc d'avoir, sur chaque arête $i+1/2, i \in \{1, \ldots, N\}$,

$$\begin{aligned} & 1 - 2\frac{\Delta t}{\Delta x}\left[|v^*_{i+1/2}| + \theta |\delta v^*_{i+1/2}|\right] > 0 \\ \Leftrightarrow\ & \Delta x - 2\Delta t\left[|v^*_{i+1/2}| + \theta |\delta v^*_{i+1/2}|\right] > 0. \end{aligned} \tag{3.149}$$

La valeur $v^*_{i+1/2}$ est connue explicitement. En revanche, $\delta v^*_{i+1/2}$ résulte de la solution d'un système linéaire dont le second membre dépend lui-même de Δt. L'idée est donc de majorer $\delta v^*_{i+1/2}$ par

$$|\delta v^*_{i+1/2}| < M^{(1)}_{i+1/2}\Delta t + M^{(2)}_{i+1/2}(\Delta t)^2, \tag{3.150}$$

avec $M^{(1)}_{i+1/2}$ et $M^{(2)}_{i+1/2}$ deux constantes positives. Dans ce cas, nous aurions à satisfaire pour chaque interface une condition de type

$$-2\theta M^{(2)}_{i+1/2}(\Delta t)^3 - 2\theta M^{(1)}_{i+1/2}(\Delta t)^2 - 2|v^*_{i+1/2}|\Delta t + \Delta x > 0. \tag{3.151}$$

Notons que la dépendance quadratique en Δt du second membre de (3.150) est une conséquence directe de celle du second membre du système linéaire implicite qui dépend non seulement de Δt mais aussi de $(\Delta t)^2$ (voir (3.138)). Il faudra donc majorer $|\delta v^*_{i+1/2}|$ par une fonction du second degré en Δt. En conséquence, (3.151) est une inéquation du troisième degré en Δt qui possède certaines propriétés intéressantes.

Nous allons voir maintenant comment déterminer $M^{(1)}_{i+1/2}$ et $M^{(2)}_{i+1/2}$ afin d'obtenir le meilleur majorant possible de $|\delta v^*_{i+1/2}|$.

3.5.6.1 Majorations de $|\delta v^*_{i+1/2}|$

Afin de trouver une majoration de $|\delta v^*_{i+1/2}|$, nous allons procéder par plusieurs étapes successives.

1. Modification du système
   L'expérience numérique montre que les termes $(\partial_v P)_i$ dans le système linéaire par blocs sont beaucoup plus petits que le coefficient de relaxation a. Ils sont donc négligeables devant tous les autres termes (liés à $(\partial_\tau P)_i$) dans ces blocs. Ceci est cohérent avec le fait que la pression P ne dépend a priori que de τ et Y, la dépendance de P par rapport à v est indirecte via le glissement ϕ. Ainsi dans la pratique, nous ne prenons pas en compte l'effet des $(\partial_v P)_i$

   $$(\partial_v P)_i = 0. \tag{3.152}$$

   De plus, on décide de remplacer tous les $(\partial_\tau P)_i$ par

   $$(\partial_\tau P)_i = -a^2, \tag{3.153}$$

   ce qui revient à « sur-dissiper ». La structure du système par blocs devient plus simple.

2. Changement de variables
   Au lieu de travailler avec $(\delta\tau_i, \delta v_i)$, il est plus avantageux de travailler avec les variables

$$\begin{aligned} \delta C_i^+ &= \frac{\delta v_i + a\delta\tau_i}{2}, \\ \delta C_i^- &= \frac{\delta v_i - a\delta\tau_i}{2}, \end{aligned} \tag{3.154}$$

   issues de la diagonalisation du système modifié à l'étape 1.

3. Résolution explicite « à la main »
   Par un procédé de double balayage, on parvient à établir une expresion explicite de la solution $(\delta C_i^+, \delta C_i^-)$ au moyen des coefficients de propagation (3.156) et (3.157) définies ci-dessous. À partir des $(\delta C_i^+, \delta C_i^-)$, on déduit l'expression algébrique pour $\delta v_{i+1/2}^*$.

4. Majoration local directe
   On s'appuie sur l'expression de $\delta v_{i+1/2}^*$ pour trouver un majorant de la forme (3.150).

Les détails de ces 4 étapes se trouvent dans l'annexe A. Ici on ne montre que des résultats finals. Tout d'abord, introduisons quelques grandeurs algébriques particulièrement commodes pour la suite.

***Définition** 3.7. Définissons*

- *les nouveaux résidus*

$$\begin{aligned} (\mathcal{R}_{\mathcal{C}^+})_i^n &= \frac{(\widetilde{\mathcal{R}}_v)_i^n + a(\widetilde{\mathcal{R}}_\tau)_i^n}{2}, \\ (\mathcal{R}_{\mathcal{C}^-})_i^n &= \frac{(\widetilde{\mathcal{R}}_v)_i^n - a(\widetilde{\mathcal{R}}_\tau)_i^n}{2}, \end{aligned} \tag{3.155}$$

- *le coefficient local de propagation*

$$e_i(\Delta t) = \frac{a\mu_i^n}{1 + a\mu_i^n}, \qquad \textit{avec } \mu_i^n = \frac{\theta\Delta t}{\rho_i^n \Delta x}, \tag{3.156}$$

- *le coefficient cumulé de propagation*

$$E_j^i(\Delta t) = \begin{cases} \prod_{k=j}^i e_k(\Delta t) & \textit{si } j \le i, \\ 1 & \textit{sinon.} \end{cases} \tag{3.157}$$

Les résidus $(\mathcal{R}_{\mathcal{C}^+})_i^n$ et $(\mathcal{R}_{\mathcal{C}^-})_i^n$ dépendent en fait du pas de temps Δt à cause de la dépendance de $(\widetilde{\mathcal{R}}_v)_i^n$ et $(\widetilde{\mathcal{R}}_v)_i^n$ de Δt. Normalement, nous devrions les écrire $(\mathcal{R}_{\mathcal{C}^+})_i^n(\Delta t)$ et $(\mathcal{R}_{\mathcal{C}^-})_i^n(\Delta t)$ mais nous avons supprimé cette dépendance pour simplifier la notation. D'après (3.138) on écrit les résidus $(\mathcal{R}_{\mathcal{C}^+})_i^n$ et $(\mathcal{R}_{\mathcal{C}^-})_i^n$ en fonction de Δt, à savoir

$$\begin{aligned} (\mathcal{R}_{\mathcal{C}^+})_i^n &= \Delta t(\mathcal{R}_{C^+}^{(1)})_i^n - (\Delta t)^2(\mathcal{R}_{C^+}^{(2)})_i^n, \\ (\mathcal{R}_{\mathcal{C}^-})_i^n &= \Delta t(\mathcal{R}_{C^-}^{(1)})_i^n - (\Delta t)^2(\mathcal{R}_{C^-}^{(2)})_i^n. \end{aligned} \tag{3.158}$$

Les termes $(\mathcal{R}_{C^+}^{(p)})_i^n$ sont définis par

$$(\mathcal{R}_{C^+}^{(p)})_i^n = \frac{(\mathcal{R}_v^{(p)})_i^n + a(\mathcal{R}_\tau^{(p)})_i^n}{2}, \qquad \text{et} \qquad (\mathcal{R}_{C^-}^{(p)})_i^n = \frac{(\mathcal{R}_v^{(p)})_i^n - a(\mathcal{R}_\tau^{(p)})_i^n}{2}, \tag{3.159}$$

pour $p = 1, 2$. Ils sont connus et ne dépendent pas de Δt.

L'écriture de $(\mathcal{R}_{\mathcal{C}^+})_i^n$ et $(\mathcal{R}_{\mathcal{C}^+})_i^n$ sous forme de (3.158) nous facilitera la tâche pour trouver une majoration de $|\delta v^*_{i+1/2}|$. En effet, en séparant tout ce qui est lié à Δt et à $(\Delta t)^2$, nous pouvons majorer les coefficients de chacun de ces termes de manière indépendante.

En ce qui concerne les coefficients e_i et E_j^i, on remarque qu'ils dépendent aussi de Δt à cause des μ_i^n. Cependant, nous constatons que les coefficients locaux de propagation e_i (on verra dans l'annexe pourquoi ils s'appellent ainsi) sont tous compris entre 0 et 1 d'après leur définition. En conséquence, les coefficients cumulés de propagation E_j^i appartiennent aussi à l'intervalle $[0,1]$. Nous verrons plus tard que grâce à ces propriétés, nous pouvons éliminer la dépendance en Δt des E_j^i ce qui simplifie beaucoup la procédure de majoration de $|\delta v^*_{i+1/2}|$. Dans la suite, on n'écrit pas cette dépendance afin d'abréger les formules.

***Lemme** 3.8. Une majoration de $|\delta v^*_{i+1/2}|$ est*

$$|\delta v^*_{i+1/2}| \leq (\Delta t)^2 \mathcal{M}^{(2)}_{i+1/2}(\Delta t) + \Delta t \mathcal{M}^{(1)}_{i+1/2}(\Delta t) + \mathcal{M}^{(0)}_{i+1/2}(\Delta t) \tag{3.160}$$

avec

$$\mathcal{M}^{(0)}_{i+1/2}(\Delta t) = \frac{2(1+\Theta)E_1^i}{2\theta}\tau_0|\delta q_T| + \frac{E_{i+1}^N}{a}|\delta p_{N+1}|, \tag{3.161}$$

et pour $p = 1, 2$

$$\begin{aligned}\mathcal{M}^{(p)}_{i+1/2}(\Delta t) = \quad & 2(1-E_1^i)\max_{1\leq k\leq i}|(\mathcal{R}^{(p)}_{C^-})^n_k| + E_{i+1}^N(1-E_{i+1}^N)\max_{i+1\leq k\leq N}|(\mathcal{R}^{(p)}_{C^-})^n_k| \\ & + 2\Theta E_1^i(1-E_1^i)\max_{1\leq k\leq i}|(\mathcal{R}^{(p)}_{C^+})^n_k| + \quad (1-E_{i+1}^N)\max_{i+1\leq k\leq N}|(\mathcal{R}^{(p)}_{C^+})^n_k|.\end{aligned} \tag{3.162}$$

Dans (3.161), le paramètre Θ est défini par

$$\Theta = \frac{\theta - \dfrac{q_T^n}{a}}{\theta + \dfrac{q_T^n}{a}}, \tag{3.163}$$

où q_T^n est le débit total imposé à l'instant t^n à l'entrée de la conduite. Les incréments δq_T^n et δp_{N+1} sont respectivement la différence de débit total à l'amont et de la pression à l'aval entre l'instant t^n et t^{n+1}, *i.e* $\delta q_T = q_T^{n+1} - q_T^n$ et $\delta p_{N+1} = p_{N+1}^{n+1} - p_{N+1}^n$.

La démonstration de ce Lemme est lourde et se trouve dans l'annexe A. On remarque que les termes $\mathcal{M}^{(p)}_{i+1/2}$, $p = 1, 2$, au second membre de (3.160) dépendent de E_1^i et de E_{i+1}^N. Plus précisément, (3.160) peut s'écrire

$$|\delta v^*_{i+1/2}| \leq \Delta t\left[f_1(E_1^i) + f_2(E_{i+1}^N)\right] + (\Delta t)^2\left[f_3(E_1^i) + f_4(E_{i+1}^N)\right], \tag{3.164}$$

où les fonctions $(f_k)_{k=1,\dots,4}$ sont définies par

$$\begin{aligned}
f_1(E_1^i) &= 2(1-E_1^i)\max_{1\leq k\leq i}|(\mathcal{R}^{(1)}_{C^-})^n_k| + 2\Theta E_1^i(1-E_1^i)\max_{1\leq k\leq i}|(\mathcal{R}^{(1)}_{C^+})^n_k| + \frac{2(1+\Theta)E_1^i}{2\theta}\tau_0|q_T'|, \\
f_2(E_{i+1}^N) &= E_{i+1}^N(1-E_{i+1}^N)\max_{i+1\leq k\leq N}|(\mathcal{R}^{(1)}_{C^-})^n_k| + (1-E_{i+1}^N)\max_{i+1\leq k\leq N}|(\mathcal{R}^{(1)}_{C^+})^n_k| + \frac{E_{i+1}^N}{a}|p'_{N+1}|, \\
f_3(E_1^i) &= 2(1-E_1^i)\max_{1\leq k\leq i}|(\mathcal{R}^{(2)}_{C^-})^n_k| + 2\Theta E_1^i(1-E_1^i)\max_{1\leq k\leq i}|(\mathcal{R}^{(2)}_{C^+})^n_k|, \\
f_4(E_{i+1}^N) &= E_{i+1}^N(1-E_{i+1}^N)\max_{i+1\leq k\leq N}|(\mathcal{R}^{(2)}_{C^-})^n_k| + (1-E_{i+1}^N)\max_{i+1\leq k\leq N}|(\mathcal{R}^{(2)}_{C^+})^n_k|.
\end{aligned} \tag{3.165}$$

Pour obtenir (3.164) on a approché δq_T et $\delta\Pi_{N+1}$ par $\Delta t q'_T$ et $\Delta t p'_{N+1}$. Les fonctions $(f_i)_{i=1,\dots,4}$ peuvent toutes se mettre sous la même forme $f(\xi) = (1-\xi)A+\xi B+\xi(1-\xi)D$ avec tous les coefficients positifs ou nuls et ξ est compris entre 0 et 1. Nous sommes donc amenés à étudier le maximum de cette fonction sur l'intervalle $[0,1]$ d'où la proposition suivante.

***Proposition** 3.3. Soit la fonction $f : \xi \in [0,1] \longmapsto \mathbb{R}_+$ définie par*

$$f(\xi) = (1-\xi)A + \xi B + \xi(1-\xi)D \quad \text{avec } A, B, D \geq 0. \tag{3.166}$$

Alors son maximum est donné par

$$\max_{\xi\in[0,1]} f(\xi) = \varphi(A,B;D) := \begin{cases} \max(A;B) & \text{si } D \leq |B-A|, \\ \dfrac{A+B}{2} + \dfrac{D}{4} + \dfrac{(B-A)^2}{4D} & \text{si } D > |B-A|. \end{cases} \tag{3.167}$$

La démonstration n'est pas compliquée, on peut la trouver dans l'annexe A. Lorsque l'on applique cette proposition aux quatre fonctions $(f_k)_{k=1,\dots,4}$ définies ci-dessus, nous pouvons établir une autre majoration de $|\delta v^*_{i+1/2}|$ tout en oubliant la dépendance en Δt des E^i_1 et E^N_{i+1} (compris entre 0 et 1). La valeur du maximum, notée $\varphi(A,B;D)$, est désormais vue comme une fonction à trois arguments.

Le Lemme suivant est alors la conséquence de l'application de la Proposition 3.3 à (3.164).

***Lemme** 3.9. Une majoration de $|\delta v^*_{i+1/2}|$ est donnée par*

$$|\delta v^*_{i+1/2}| \leq M^{(1)}_{i+1/2}\Delta t + M^{(2)}_{i+1/2}\Delta t^2, \tag{3.168}$$

avec

$$\begin{aligned} M^{(1)}_{i+1/2} = \quad & 2\varphi\Big(\max_{1\leq k\leq i}|(\mathcal{R}^{(1)}_{C^-})^n_k|, \quad \frac{1+\Theta}{2\theta}\tau_0|\delta q_T|; \Theta\max_{1\leq k\leq i}|(\mathcal{R}^{(1)}_{C^+})^n_k|\Big) \\ + \quad & \varphi\Big(\max_{i+1\leq k\leq N}|(\mathcal{R}^{(1)}_{C^+})^n_k|, \frac{1}{a}|\delta P_{N+1}|; \quad \max_{i+1\leq k\leq N}|(\mathcal{R}^{(1)}_{C^-})^n_k|\Big), \end{aligned} \tag{3.169}$$

et

$$\begin{aligned} M^{(2)}_{i+1/2} = \quad & 2\varphi\Big(\max_{1\leq k\leq i}|(\mathcal{R}^{(2)}_{C^-})^n_k|, \quad 0; \Theta\max_{1\leq k\leq i}|(\mathcal{R}^{(2)}_{C^+})^n_k|\Big) \\ + \quad & \varphi\Big(\max_{i+1\leq k\leq N}|(\mathcal{R}^{(2)}_{C^+})^n_k|, 0; \max_{i+1\leq k\leq N}|(\mathcal{R}^{(2)}_{C^-})^n_k|\Big). \end{aligned} \tag{3.170}$$

En vue de l'implémentation numérique, on utilise cette majoration qui peut être mise en œuvre facilement. Notons que d'après la définition de φ, $M^{(1)}_{i+1/2}$ et $M^{(2)}_{i+1/2}$ sont deux constantes positives.

3.5.6.2 Calcul de Δt

Ayant obtenu la majoration de $|\delta v^*_{i+1/2}|$ suivant le Lemme 3.9, nous pouvons calculer un pas de temps pour notre schéma Lagrange-Projection semi-implicite permettant de préserver la positivité du schéma. Le pas de temps optimal Δt est le plus grand nombre satisfaisant

$$\begin{aligned} & -2\theta M^{(2)}_{i+1/2}\Delta t^3 - 2\theta M^{(1)}_{i+1/2}\Delta t^2 - 2|v^*_{i+1/2}|\Delta t + \Delta x > 0 \\ \iff\ & 2\theta M^{(2)}_{i+1/2}\Delta t^3 + 2\theta M^{(1)}_{i+1/2}\Delta t^2 + 2|v^*_{i+1/2}|\Delta t - \Delta x < 0 \end{aligned} \tag{3.171}$$

Pour avoir le pas de temps Δt, il suffit donc de trouver la solution positive de $A(\Delta t)^3 + B(\Delta t)^2 + C\Delta t + D = 0$ avec $A = 2\theta M^{(2)}_{i+1/2}, B = 2\theta M^{(1)}_{i+1/2}, C = 2|v^*_{i+1/2}|$ et $D = -\Delta x$. Cette solution positive existe grâce aux propriétés de A, B, C (tous positifs) et D (négatif).

Fig. 3.12. La fonction $g(\zeta) = A\zeta^3 + B\zeta^2 + C\zeta + D$ avec $A, B, C \geq 0$ et $D < 0$ est croissante sur $[0; +\infty]$ et admet une unique solution ζ^* sur cet intervalle.

***Proposition** 3.4. Considérons une fonction $g(\zeta)$ définie par*

$$\zeta \in [0; +\infty] \longmapsto g(\zeta) = A\zeta^3 + B\zeta^2 + C\zeta + D \text{ avec } A, B, C \geq 0 \text{ et } D < 0, \tag{3.172}$$

Alors $g(\zeta)$ est croissante sur $[0; +\infty]$ et $g(\zeta) = 0$ admet une unique solution ζ^* positive (représentée sur la Fig. 3.12).

Démonstration On calcule la dérivée de g

$$g'(\zeta) = 3A\zeta^2 + 2B\zeta + C \geq 0 \text{ sur } [0; +\infty]. \tag{3.173}$$

Alors $g(\zeta)$ est croissante sur $[0; +\infty]$.

De plus $g(0) = D < 0$ et $\lim_{\zeta \to +\infty} g(\zeta) = +\infty$, $g(\zeta) = 0$ admet donc une unique solution positive ζ^* sur $[0; +\infty]$. □

Nous prenons alors le pas de temps Δt égal à l'unique solution positive de

$$2\theta M^{(2)}_{i+1/2}\Delta t^3 + 2\theta M^{(1)}_{i+1/2}\Delta t^2 + 2|v^*_{i+1/2}|\Delta t - \Delta x = 0. \tag{3.174}$$

Ce pas de temps est basé sur les ondes lentes en utilisant le nombre CFL^{exp} et on le note $\Delta t^{exp\#}$ pour distinguer avec le pas de temps Δt^{exp} calculé avec le schéma explicite. En supposant que le schéma semi-implicite que l'on vient de construire reste stable pour de grands pas de temps, on introduit le nombre CFL^{imp}. Le pas de temps doit être alors déterminé en conformité avec la condition CFL^{imp}, basé sur les ondes acoustiques. Il faudra donc prendre un Δt tel que

$$\Delta t = \min\left(\Delta t^{exp\#}, \frac{\mathrm{CFL}^{imp}\Delta x}{c}\right), \tag{3.175}$$

où c est la vitesse caractéristique maximale sur toutes les interfaces à l'intant t^n, c'est-à-dire

$$c = \max_i \max\left(|v^* - a\tau^*_L|^n_{i+1/2}, |v^* + a\tau^*_R|^n_{i+1/2}\right), \tag{3.176}$$

avec $v^{n,*}_{i+1/2}$ et $\tau^{n,*}_{i+1/2}$ sont la vitesse et le volume spécifique dans la solution du problème de Riemann à l'interface $i + 1/2$ (voir (2.60)).

Dans le cadre de notre travail, on prend $\mathrm{CFL}^{exp} = 0.5$ et $\mathrm{CFL}^{imp} = 20$.

Fig. 3.13. La fonction $h(\zeta) = A\zeta^2 + B\zeta + C$ avec $A, B \leq 0$ et $C > 0$ est décroissante sur $[0; +\infty$ et admet une unique solution ζ^* sur cet intervalle.

***Remarque** 3.10.* Dans le cas de glissement nul, il n'y aura pas de contribution de $\mathcal{R}_Y$, *i.e.* $(\mathcal{R}_Y)_i = 0$. En conséquence, les termes $M^{(2)}_{i+1/2}$ sont nuls $(\mathcal{R}^{(2)}_{C^+})_i = (\mathcal{R}^{(2)}_{C^-})_i = (\mathcal{R}^{(2)}_{\tau})_i = (\mathcal{R}^{(2)}_{v})_i = 0$ et nous avons uniquement l'inégalité du second degré

$$-2\theta M^{(1)}_{i+1/2}\Delta t^2 - 2|v^*_{i+1/2}|\Delta t + \Delta x > 0 \tag{3.177}$$

Considérons $h(\zeta) = A\zeta^2 + B\zeta + C$ avec $A, B \leq 0$ et $C > 0$. Alors $h'(\zeta) = 2A\zeta + B < 0$ sur $[0; +\infty$ donc h est décroissante sur $[0; +\infty]$ (voir Fig. 3.13).

De plus, $h(0) = C > 0$ et $\lim_{\zeta \to +\infty} h(\zeta) = -\infty$ donc $h(\zeta) = 0$ admet une solution unique positive.

Appliquons à notre cas, nous trouvons facilement le pas de temps dans le cas sans glissement

$$\Delta t = \frac{-|v^*_{i+1/2}| + \sqrt{|v^*_{i+1/2}|^2 + 2\theta M^{(1)}_{i+1/2}}}{2\theta M^{(1)}_{i+1/2}} \tag{3.178}$$

3.5.7 Montée en ordre et intégration du terme source

La montée en ordre en espace du schéma L-P semi-implicite se fait de manière identique que le schéma L-P explicite (voir section 3.3). En revanche, la montée en ordre 2 en temps se fait à travers le paramètre d'implicitaion θ. Nous ne faisons pas le Runge-Kutte pour le schéma semi-implicite. L'intégration du terme source est identique que celle présentée pour le schéma explicite (3.4).

3.5.8 Résultats numériques

Dans cette partie, nous présentons quelques résultats numériques obtenus avec le schéma Lagrange-Projection semi-implicite et les comparer avec ceux obtenus par le schéma eulérien sur différents aspects : précision et robustesse.

3.5.8.1 Validation sur tubes à chocs

Le cas-test utilisé ici est de type tube à chocs. On ne prend en compte ni les conditions aux limites ni le terme source. La valeur de la condition CFL implicite est égale à 20. Les résultats sont obtenus

avec une reconstruction de pente en espace. Le coefficient d'implicitation θ dans le cas du schéma Lagrange-Projection semi-implicite est pris égal à 0.7.

La conduite est de longueur 100 m dont la discontinuité se situe au milieu à 50 m. Le pas d'espace est $\Delta x = 0.5$ m. On travaille avec un liquide incompressible et un gaz parfait.

On considère les états gauche (L) et droit (R)

$$\begin{pmatrix} \rho \\ Y \\ v \end{pmatrix}_L = \begin{pmatrix} 453.197 \\ 0.00705 \\ 24.8074 \end{pmatrix} \quad \text{et} \quad \begin{pmatrix} \rho \\ Y \\ v \end{pmatrix}_R = \begin{pmatrix} 454.915 \\ 0.0108 \\ 1.7461 \end{pmatrix}.$$

La solution de Riemann est une séquence de trois ondes composées d'un choc, une discontinuité de contact et un choc. Elles se propagent repectivement aux vitesses $\omega^- = -40.03$ m/s (onde rapide), $\omega^0 = 10$ m/s (onde lente) et $\omega^+ = 67.24$ m/s (onde rapide).

Fig. 3.14. Tubes à choc - Glissement nul - 3 ondes.

Le résultat illustré dans la Fig. 3.14 correspond au temps $T = 0.5$ s. Sur les trois premiers composantes (ρ, Y, v) on observe bien trois ondes : une onde rapide vers la gauche avec la vitesse ω^-, une onde lente vers la droite avec la vitesse ω^0 et une onde rapide vers la droite avec la vitesse ω^+. On constate que les résultats sont comparables entre les deux schémas. Sur les ondes rapides le schéma L-P donne une diffusion moins importante que le schéma eulérien. Tandis que sur l'onde lente la qualité des résultats est identique, *i.e* on obtient la même précision sur cette onde de transport qui nous intéresse particulièrement.

3.5.8.2 Permier cas-test avec des conditions aux limites

Le premier cas-test consiste à simuler le passage en monophasique gaz dans une conduite horizontale. Les détails de cette expérience se trouvent dans la Fig. 3.15. Le résultat du schéma eulérien est présenté dans la Fig. 3.16 et celui de L-P dans la Fig. 3.17. On constate que les deux résultats sont très similaires et les conditions aux limites sont bien respectées.

Pour démontrer l'intérêt de ce cas-test, détaillons quelques points importants de cette simulation. Tout d'abord, on laisse stabiliser l'écoulement pendant 200 secondes (période de *stabilisation*) afin d'atteindre l'état *stationnaire numérique*. À 250 secondes, on annule le débit de liquide à l'amont et simultanément on double la pression à l'aval. Pendant cette *durée de la rampe*, le débit de gaz reste constant. Dès l'augmentation de la pression à l'aval, on constate qu'une entrée instantanée de gaz dans la conduite à cet endroit ce qui a pour effet de générer une poche de gaz ici, *i.e.* $R_g = 1$ à $x = 3000$ m.

De plus, lors de la coupure totale du liquide à 450 secondes, une autre poche de gaz se forme à l'amont et s'étend vers la sortie de la conduite. Cet état à l'amont demeure jusqu'à la fin de la simulation. En conséquence, à partir de 500 secondes, il y a un mélange gaz-liquide entouré par deux poches de gaz dans la conduite. Ce phénomène s'achève à 1000 secondes quand l'écoulement devient diphasique à l'aval. Au temps 2500 secondes, la poche de gaz propageant de l'amont arrive à $x = 1000$ m et c'est au tour de ce point de devenir monophasique gaz.

À travers ce cas-test, nous constatons que le passage d'un état diphasique à un état monophasique gaz est bien contrôlé. Le principe du maximum sur les fractions massiques du gaz n'est pas violé avec le schéma Lagrange-Projection semi-implicite, *i.e.* $Y \in [0, 1]$. De plus, les conditions aux limites sont bien respectées ce qui démontre l'efficacité du traitement de ces conditions.

Géométrie de la conduite :	- Longueur	:	4000 m, horizontal	
	- Diamètre	:	0.146 m	
	- Pas d'espace	:	12.5 m	
Discrétisation :	- CFL implicite	:	20.0	
Source et Ordre :	- Avec source et reconstruction de pente			
Modèles :	- Thermodynamique	:	liquide incompressible	
	- Hydrodynamique	:	glissement nul	
Conditions opératoires :	- Durée de la stabilisation			: 200 s
	- Durée de la rampe			: 250 s
	- Condition limite amont sur le débit de gaz			: 10 kg/m^2/s
	- Condition limite amont sur le débit de liquide			: 1000 kg/m^2/s
				à 0 kg/m^2/s
	- Condition limite aval sur la pression			: 1 bar
				à 2 bar

Fig. 3.15. Détails de la première expérience avec conditions aux limites - Schéma semi-implicite.

Fig. 3.16. Variations des débits à l'amont pour le schéma eulérien - Schéma semi-implicite.

Fig. 3.17. Variations des débits à l'amont pour le schéma Lagrange-Projection - Schéma semi-implicite.

3.5.8.3 Deuxième cas-test avec des conditions aux limites

Le deuxième cas-test consiste à considérer un cas difficile où apparaît le phénomène de severe slugging, *i.e.* la formation des bouchons au point bas de la conduite qui précède une conduite vertical, appellé aussi « riser ». Les détails de ce cas-test sont présentés dans la Fig. 3.18 et les résultats sont montrés dans la Fig. 3.19.

Dans ces résultats, nous montrons l'évolution au cours du temps des débit de gaz et de liquide, de la vitesse et de la pression du mélange à $x = 60$ m, c'est-à-dire en bas du riser. Nous constatons tout d'abord qu'il s'agit d'un phénomène périodique, chaque période est composée de 4 phases suivantes :

- la pression en bas du riser vertical n'est pas suffisamment forte pour pousser le liquide vers la sortie de la conduite, le liquide reste donc dans cette colonne et empêche le gaz à passer, ce qui explique une augmentation de la pression au coin. Pendant ce temps, la variation des débits du gaz et du liquide est faible,
- le mélange est cumulé dans la colonne jusqu'à quand la pression en bas du riser excède une valeur critique et pousse violemment l'ensemble liquide et gaz vers le haut d'où l'augmentation brutale des débits du gaz et du liquide en ce point de forte pression,
- la sortie du mélange ainsi que la détente du gaz dans le riser engendre la décroissance de la pression en bas du riser,
- la pression décroît de plus en plus vite et cela provoque la retombée du liquide dans la la colonne ce qui produit une oscillation de la vitesse en bas du riser. Il y a de nouveau des accumulations du mélange dans le coin et le phénomène recommence ainsi.

Géométrie de la conduite :	- Longueur	:	74 m
	- Inclinaison :	:	0° pour $x \in [0, 60]$ et 90° pour $x \in [60, 74]$
	- Diamètre	:	0.05 m
	- Nombre de mailles	:	256
Discrétisation :	- CFL explicite	:	0.5
	- CFL explicite	:	20
Source et Ordre :	- Avec source et reconstruction de pente		
Modèles :	- Thermodynamique	:	liquide incompressible
	- Hydrodynamique	:	glissement Zuber-Findlay IFP
Conditions opératoires :	- Durée de la stabilisation	:	0 s
	- Durée de la rampe	:	pas de rampe
	- Condition limite amont sur le débit de gaz	:	1.96e-4 kg/Area/s
	- Condition limite amont sur le débit de liquide	:	7.854e-2 kg/Area/s
	- Condition limite aval sur la pression	:	1 bar

Fig. 3.18. Détails de l'expérience severe slugging - Schéma semi-implicite.

Fig. 3.19. Évolution du débi de gaz et du liquide, de la vitesse et de la pression en bas de la conduite vertical. Phénomène périodique avec des slugs.

Ce cas test démontre l'efficacité de la méthode de Lagrange-Projection appliquée au cas-test réaliste d'écoulements diphasiques. Les résultats sont tout à fait comparables avec ceux obtenus avec le schéma eulérien. Le schéma Lagrange-Projection semi-implicite en uniforme permet donc de simuler les problèmes de severe slugging.

3.5.8.4 Convergence en maillage

Nous allons présenter dans cette partie la convergence en maillage en montrant les erreurs en norme L^1 obtenus avec deux schémas : eulérien et Lagrange-Projection en semi-implicite. L'erreur calculée est définie par (3.83) et la donnée initiale est une donnée très régulière définie par (3.84). La solution de référence est calculée sur une grille uniforme très fine comme en explicite (32768 mailles). On constate que les deux schémas semi-implicite donnent des mêmes erreurs, comme vu précédemment en explicite. La Fig. 3.20 montre que les deux schémas sont d'ordre 2 en espace.

Fig. 3.20. Convergence de l'erreur absolue de la densité en norme L^1 - Semi-implicite.

3.5.9 Analyse de temps CPU pour le schéma Lagrange-Projection semi-implicite

Comme nous avons déjà évoqué précédemment, un avantage majeur du schéma Lagrange-Projection semi-implicite est sa rapidité. En effet, au lieu de résoudre un système linéaire par blocs de taille 5×5 comme fait le schéma eulérien, nous ne travaillons qu'avec des blocs de tailles 2×2. Par conséquent, nous avons moins de calculs à faire ce qui permet d'accélérer les simulations numériques. Afin de démontrer cette performance, nous avons fait une analyse algorithique pour le schéma L-P semi-implicite d'ordre 1 en temps et en espace, sans terme source ni conditions aux limites. C'est le cas le plus simple où nous pouvons voir facilement le facteur de gain en temps de calcul du schéma L-P par rapport au schéma eulérien. Les détails de cette analyse algorithmique sont présentés dans l'annexe B.

Nous allons tout d'abord faire un récapitulatif du nombre d'opérations et d'évaluations pour chaque méthode (Tab. 3.1 et 3.2). Dans ces tableaux, N désigne le nombre de mailles sur le maillage uniforme utilisé. Ici, les opérations sont les opérateurs $+, -, *, /, \sqrt{}, \max(,), \min(,)$ tandis que les évaluations sont les appels à une boîte « noire » pour avoir $P, \partial_\tau P, \partial_Y P, \partial_v P, \sigma$ et $\partial_Y \sigma$. Nous constatons que plus de la moitié des opérations est consacrée au calcul de pas de temps (étape préliminaire) pour le schéma L-P. Pour ce dernier, la construction et la résolution du système linéaire ne prennent que 40% d'opérations tantdis qu'il en nécessite 87% pour le schéma eulérien (étape mathématique et résolution du système linéaire). Au total, il faut compter 230 opérations pour le schéma L-P semi-implicite plus 11 évaluations. Quant au schéma eulérien, il a besoins de 901 opérations et 13 évaluations.

Nous nous intéressons particulièrement au gain de temps de calcul du schéma L-P semi-explicite. Le rapport de coût entre le schéma L-P et eulérien est $901/230 \approx 3.92$ ce qui est important. Comparer même au schéma eulérien réduite avec 773 opérations (voir annexe B), nous avons un gain de temps de calcul à l'ordre de $773/230 \approx 3.19$. La raison de ce gain important est que le schéma eulérien doit résoudre un système linéaire constitué des matrices de taille deux fois plus grand (5×5) dont la construction nécessite déjà beaucoup d'opérations (matrice de diffusion de type Roe). C'est pourquoi nous avons adopté la décomposition Lagrange-Projection qui permet non seulement de réduire la taille du système mais aussi de simpliciter la construction des matrices par blocs.

Évaluations de base	$13N$ évaluations	
Étape physique	$113N$ opérations	13%
Étape mathématique	$404N$ opérations	45%
Résolution du système linéaire	$384N$ opérations	42%
Total	$901N$ opérations	100

Tableau 3.1 Nombre d'opérations et d'évaluations du schéma eulérien semi-implicite.

Évaluations de base	$11N$ évaluations	
Étape préliminaire	$128N$ opérations	56%
Étape Lagrange	$96N$ opérations	40%
Étape Projection	$6N$ opérations	4%
Total	$230N$ opérations	100

Tableau 3.2 Nombre d'opérations et d'évaluations du schéma Lagrange-Projection semi-implicite.

Regardons maintenant ce que donnent les résultats du temps de calcul. Le tableau 3.3 montre les temps CPU ainsi que les gains entre deux schéma L-P et eulérien semi-implicite pour trois expériences. La première correspond à l'expérience avec conditions aux limites et glissement nul. Elle est détaillée dans la Fig. 3.15 précédente. La deuxième est celle déjà présentée dans la partie explicite (voir 3.4.1.2) mais avec un glissement non nul (Zuber-Findlay IFP) et la conduite est inclinée de 30 degrés vers le haut. La troisième correspond à un cas-test dont le régime d'écoulement suit la loi dispersé, la conduite est aussi inclinée de 30 degrés. Toutes les simulations sont faites avec le schéma semi-implicite d'ordre 2 en temps avec reconstruction de pente en espace. Le terme source est pris en compte et le nombre de maille est égal à 256.

Cas-test	Eulérien	LP	Eulérien/LP
1. Expérience 1 - Glissement nul	25.93 s	17.57 s	**1.48**
2. Expérience 2 - Glissement Z-F IFP	64.7 s	52.94 s	**1.22**
3. Expérience 3 - Glissement dispersé	3 min 42 s	2 min 20 s	**1.59**

Tableau 3.3 Temps CPU sur les cas-tests avec conditions aux limites en semi-implicite. Gain en temps CPU entre le schéma L-P semi-implicite et eulérien semi-implicite.

Nous constatons que le schéma L-P est plus rapide que le schéma eulérien dans toutes les trois expériences. Les facteurs de gain en temps CPU varient entre 1.2 et 1.6 pour ces cas-tests réalistes. Notons que ces gains ne correspondent pas aux gains théoriques calculés précédemment (égal à 3.19), car le passage à l'ordre 2 en espace, le traitement des termes sources et des conditions limites introduisent un degré de complexité algorithmique supplémentaire.

Fig. 3.21. Évolution du pas de temps au cours du temps - Eulérien (gauche) et Lagrange-Projection (droite).

Afin de mieux comprendre l'influence des différents critères intervenant dans le choix du pas de temps Δt, nous allons analyser l'évolution de celui-ci en semi-implicite. La Fig. 3.21 montre cette évolution au cours du temps pour l'expérience 3.8. On constate que sous la même condition CFL explicite, le schéma eulérien donne un pas de temps plus grand que celui du L-P. Ceci est dû au fait que l'approximation a priori de Δt dans le schéma Lagrange-Projection permet non seulement d'assurer la stabilité du système mais aussi de préserver la positivité de la densité et ainsi que le principe du maximum de la fraction massique. Les pas de temps, pour les deux schémas, sont pris finalement par la CFL implicite pour ne pas détériorer les ondes acoustiques. On obtient pour les deux schémas un pas de temps comparable d'où l'intérêt du schéma L-P au niveau de temps CPU, grâce à un plus petit nombre d'opérations élémentaires nécessaires à chaque pas de temps.

3.5.10 Conclusion sur le schéma L-P semi-implicite

Nous venons de présenter le schéma Lagrange-Projection semi-implicite. Ce schéma possède plusieurs avantages par rapport au schéma eulérien présenté dans le chapitre 2. Tout d'abord, comme en explicite le schéma L-P semi-implicite garantie la positivité de la densité et le principe du maximum sur la fraction massique du gaz. Plus précisément, le pas de temps est calculé explicitement suivant une estimation a priori et sous une certaine condition CFL afin d'assurer non seulement la stabilité mais aussi la positivité du schéma. Deuxièmement, grâce à la taille plus petite du système linéaire par blocs (2 au lieu de 5), le schéma L-P semi-implicite consomme moins de temps de calcul que le schéma eulérien. De plus, les résultats numériques obtenus sont encourageants. Ils sont comparables entre les deux schémas et on obtient la même précision tant sur les ondes lentes que sur les ondes rapides. Nous avons pu aussi démontrer la robustesse du schéma L-P en montrant les simulations des cas-tests réalistes en conduites pétrolières. Pour ces derniers, les conditions aux limites sont bien traitées numériquement.

Le développement du schéma Lagrange-Projection semi-implicite de base en maillage uniforme permet, en vue d'accélération de temps de calcul et de précision, d'utiliser la technique d'adaptation en espace (Multirésolution) et en temps (Pas de Temps Local) que nous allons présenter dans les deux chapitres qui suivent.

Chapitre 4

Méthode de Multirésolution (MR)

La méthode de relaxation en Langrange-Projection du chapitre précédent a permis de rendre plus robustes les simulations en semi-implicite, de par la garantie de positivité. Accessoirement, elle a aussi permis de rendre ces simulations plus rapides, par la diminution de la taille des blocs. La recherche d'une performance encore meilleure en temps de calcul, un des axes de cette thèse, passe par l'utilisation de la technique d'adaptation de maillage en espace ou méthode Multirésolution (MR).

En effet, l'utilisation d'une grille fine sur tout le domaine permet d'avoir une bonne précision de la solution, mais conduit à un volume de calculs important. Or, une grille fine n'est nécessaire qu'aux endroits où la solution présente de fortes variations. En revanche, dans les zones où la solution est régulière, on peut utiliser des mailles grossières pour la représenter. L'idée est donc d'utiliser un maillage adaptatif où la taille des mailles varie en fonction de la régularité locale de la solution. Nous pouvons ainsi éviter les calculs inutiles et par conséquent réduire les appels aux lois de fermeture très coûteuses, ce qui permet d'avoir un gain de temps CPU significatif.

L'étude de la régularité locale de la solution est basée sur l'analyse Multirésolution discrète introduite par Harten à la fin des années 1980 [53, 57]. À partir de la représentation d'une fonction discrétisée –par ses valeurs ponctuelles ou moyennes– sur une grille uniforme, on introduit une hiérarchie dyadique de grilles de plus en plus grossières et une base d'ondelettes biorthogonales permettant de calculer la représentation multiéchelle de la fonction dans la hiérarchie de grilles. La régularité de l'ondelette sous-jacente induit une correspondance entre les coefficients de la fonction dans la représentation multiéchelle et sa régularité **locale**.

L'utilisation de la Multirésolution dans le contexte des schémas numériques pour les lois de conservations a été également proposée par Harten dans [54–56]. Les fonctions flux dans les systèmes hyperboliques rendant compte de manière réaliste des phénomènes physiques sont souvent extrêmement coûteuses à calculer car très non linéaires. De plus les solutions de ces équations pouvant présenter des singularités en temps fini, les schémas numériques développés pour les résoudre mettent souvent en œuvre comprennent souvent des tests (appelés « limiteurs ») eux aussi coûteux en temps de calculs. En revanche, comme c'est le cas dans notre application, les zones de singularités de la solution sont souvent petites et localisées. Un calcul approché et moins coûteux des flux dans le reste du domaine d'étude peut donc faire gagner du temps de calcul.

À chaque pas de temps la taille des coefficients dans la représentation multiéchelle –nommés encore les « détails de la solution »– est comparée à un seuil en dessus duquel la solution est considérée comme localement singulière. La prédiction des zones de singularité d'un pas de temps à l'autre utilise le caractère hyperbolique des équations. Les singularités se propagent en effet à vitesse finie qui est contrôlée par la condition de stabilité CFL sur le pas de temps.

En pratique ces indicateurs d'erreurs sont ensuite utilisés dans le calcul des flux en parcourant la hiérarchie des niveaux de représentation du plus grossier vers le plus fin. Dans les zones où la solution est régulière on peut interpoler les valeurs des flux numériques à partir des valeurs des flux déjà calculés sur la grille immédiatement plus grossière. Dans les zones de singularité le flux numérique coûteux du schéma de référence est utilisé.

Cette méthode Multirésolution a par la suite été développée pour le cas bidimensionnel sur des maillages cartésiens pour des schémas différences finies [15] et des volumes finis nonstructurés [1]. Des applications à de nombreux systèmes d'équations multidimensionnelles résolus par des schémas de type différences finies ont été implémentées par Chiavassa et Donat [22–24], et pour des volumes finis sur des maillages triangulaires dans [26].

La méthode de Multirésolution adaptative que nous avons implémentée dans cette thèse est basée sur la méthode mise au point par Cohen et *al* dans [27]. Elle utilise la méthode de Multirésolution d'Harten pour analyser la régularité locale de la solution et définir à chaque pas de temps le niveau de discrétisation adéquat pour représenter localement la solution. L'évolution d'une grille hybride, composée de mailles sélectionnées dans la hiérarchie de grilles en fonction de la taille des détails est prédite à chaque pas de temps. La solution est discrétisée sur cette grille hybride et le schéma volume fini de référence est adapté pour être utilisé sur cette grille non uniforme. Par ailleurs on peut à tout instant revenir à la représentation sur la grille uniforme la plus fine. L'analyse mathématique et numérique de ce schéma a été faite dans [27] dans le cas des grilles cartésiennes et développée ensuite pour différents systèmes d'équations par exemple dans [28, 29]. L'implémentation de ce type de schéma pleinement adaptatif peut être faite de manière indépendante comme une « sur-couche » par rapport au schéma de référence. Différentes approches existent pour gagner de la place mémoire en plus du temps de calcul, utilisant les tables de hachage [75] ou des arbres [82]. Dans notre cas unidimensionnel ce critère de la place mémoire n'est pas crucial et nous ne nous y attaquerons pas.

La nouveauté dans notre travail est le couplage de l'algorithme de Multirésolution adaptative introduit dans [27] avec un schéma sélectivement implicite et fonctionnant avec un pas de temps grand par rapport aux ondes acoustiques. L'implémentation des conditions aux limites variables en temps a également donné lieu à des aménagement de l'algorithme initial. La robustesse de la prédiction de la grille d'un pas de temps à l'autre a été étudiée pour le schéma semi implicite purement eulérien dans des travaux antérieurs à cette thèse dans [2, 3].

Dans ce chapitre nous rappelons les principes de la Multirésolution discrète d'Harten. Nous présentons ensuite son application à un schéma volumes finis explicite puis semi-implicite. Notre code est écrit de manière à pouvoir traiter les deux cas, et à s'adapter à plusieurs schémas de référence. La méthode MR est implémentée comme une sur-couche servant à piloter la grille adaptative pendant l'évolution de la solution (voir l'annexe C). Dans la dernière partie du chapitre nous présentons des résultats numériques pour différents cas tests industriels.

4.1 Principe de la méthode de Multirésolution

La méthode de Multirésolution est basée sur une représentation multi-échelles où la solution est représentée localement sur une hiérarchie dyadique de grilles de discrétisation. Grâce aux opérations de codage, décodage et seuillage, on peut déterminer le niveau de régularité de la solution à partir duquel on construit la grille adaptative. De plus, grâce à la gestion de la grille adaptative suivant l'heuristique d'Harten, on peut détecter automatiquement les zones d'irrégularité de la solution qui se déplacent au cours du temps. Cela permet de rendre la méthode MR plus robuste et rapide.

4.1.1 Représentation multi-échelles

Considérons $\mathbb{U}(x,t^n)$ la solution discrète du système équilibre (2.1). Le but principal de la méthode Multirésolution est de représenter cette solution sur une grille adaptative de taille de mailles variable. Nous allons construire cette grille adaptative à partir d'un ensemble de $K+1$ ($K \in \mathbb{N}^*$) grilles uniformes (taille de maille constante) régulièrement échelonnées, *i.e.* une grille uniforme de niveau k ($0 \leq k < K$) a des mailles deux fois plus grandes que celles de niveau $k+1$. Sur le domaine de calcul $[0,L]$ on considère au départ une grille uniforme $\mathcal{S}^0$ contenant N_0 ($N_0 \in \mathbb{N}^*$) mailles notées M_i^0 pour $i \in \{1,\ldots,N_0\}$. La différence avec la numérotation $i \in \{0,\ldots,N_0-1\}$ utilisée dans certaines des publications est due à la présence d'une maille fictive M_0 à l'extérieur du domaine. On note x_i^0 le centre de la maille M_i^0, Δx_i^0 sa longueur et $x_{i+1/2}^0$ l'interface séparant deux mailles M_i^0 et M_{i+1}^0. Ici, l'indice 0 désigne le niveau de la grille $\mathcal{S}^0$ parmi les $K+1$ grilles à construire. À partir de cette grille grossière, on construit une autre grille plus fine notée $\mathcal{S}^1$, de niveau 1, en divisant uniformément chacune des mailles M_i^0 en deux mailles plus petites notées M_{2i-1}^1 et M_{2i}^1. En conséquence, nous avons $M_i^0 = M_{2i-1}^1 \cup M_{2i}^1$ et la nouvelle interface $x_{2i-1/2}^1 := x_{i-1/2}^0$. On recommence ce processus de raffinement pour obtenir $K+1$ grilles uniformes de niveau 0 à K. La taille de maille de la grille uniforme au niveau k est

$$\Delta x^k = \frac{L}{N_0 2^k}, \qquad k = 0,\ldots,K. \tag{4.1}$$

Cette notation Δx^k est différente que celle utilisée classiquement pour les schémas numériques où Δx_i représente la taille de la maille M_i, i étant l'indice d'espace de la grille en considération.

L'ensemble des $K+1$ grilles imbriquées construites ainsi constitue une hiérarchie de grilles dyadiques sur laquelle on va représenter la solution.

Notons que dans la pratique, nous construisons ces grilles dyadiques de manière inverse, *i.e.* étant donné un niveau de raffinement K nous partons d'une grille dont le nombre de mailles est $N_0 2^K$. Ensuite, nous construisons des grilles de plus en plus grossières de niveau $K-1$ au niveau 0 en regroupant deux mailles consécutives de même niveau pour en faire une seule maille de niveau inférieur. La manière de construire une telle hiérarchie de grilles ne change pas fondamentalement l'idée de la méthode de Multirésolution. Nous pouvons utiliser la première comme la deuxième méthode sans problème.

Afin de trouver une grille adaptative pour la solution $\mathbb{U}(x,t^n)$, nous allons tout d'abord la représenter sur l'ensemble des $K+1$ grilles dyadiques construites précédemment. Initialement, cette solution est définie par ses valeurs discrètes sur la grille la plus fine (niveau K). On va effectuer une opération appelée Projection qui donne la valeur de la solution sur une grille plus grossière. Itérer cette opération permet de calculer les valeurs discrètes de la solution sur des grilles uniformes, du niveau le plus fin K au niveau le plus grossier 0. Il est indispensable de définir simultanément une opération inverse pour reconstruire, à partir de la solution sur la grille la plus grossière, la solution sur les grilles plus fines. Cette procédure s'appelle l'opération de Prédiction. En effet, nous verrons plus tard que le détail ou la différence entre la valeur moyenne issue de l'opération de Projection et la valeur approchée issue de l'opération de Prédiction permet de déterminer la régularité de la solution. Les opérations Projection, Prédiction ainsi que la définition précise des détails sont importantes pour la compréhension de la méthode Multirésolution. Nous allons maintenant y entrer en détails.

4.1.1.1 Opération de Projection et de Prédiction

Considérons la solution discrète $\mathbb{U}^k = (\mathbb{U}_i^k)_{i=1,\ldots,N_0 2^k}$ sur une grille uniforme de niveau k. En vue d'une méthode volumes finis, les valeurs $\mathbb{U}_i^k$ représentent la moyenne de la solution exacte $\mathbb{U}(x,t^n)$ sur les mailles M_i^k. L'opérateur de Projection P_k^{k-1} permettant de calculer la solution discrète $\mathbb{U}^{k-1}$ sur la

grille plus grossière de niveau $k-1$ à partir de $\mathbb{U}^k$ est naturellement défini par

$$\mathbb{U}^{k-1} = P_k^{k-1}\mathbb{U}^k, \tag{4.2}$$

avec

$$\mathbb{U}_i^{k-1} = \frac{1}{2}\left(\mathbb{U}_{2i-1}^k + \mathbb{U}_{2i}^k\right). \tag{4.3}$$

La valeur de la solution discrète au niveau grossier $k-1$ est obtenue en moyennant les valeurs sur les deux subdivisions lors de la construction des grilles dyadiques. Partant de la grille la plus fine, nous pouvons donc déduire toutes les valeurs représentant la solution sur tous les niveaux de grille, *i.e.* $\mathbb{U}^{K-1}, \mathbb{U}^{K-2}, \ldots, \mathbb{U}^0$. Cette opération n'est pas réversible d'où la nécessité de définir l'opération de Prédiction suivante.

L'opération de Prédiction consiste à approcher les valeurs de la solution sur une grille fine de niveau k à partir des valeurs sur une grille grossière de niveau $k-1$. Il y a plusieurs façons pour faire cela. Dans ce travail, nous adoptons une interpolation polynomiale d'ordre deux à l'intérieur du domaine et d'ordre un aux bords du domaine, à savoir

$$\widehat{\mathbb{U}}^k = P_{k-1}^k\mathbb{U}^k, \tag{4.4}$$

avec

$$\begin{aligned}\widehat{\mathbb{U}}_{2i-1}^k &= \mathbb{U}_i^{k-1} - \frac{1}{8}\left(\mathbb{U}_{i+1}^{k-1} - \mathbb{U}_{i-1}^{k-1}\right),\\ \widehat{\mathbb{U}}_{2i}^k &= \mathbb{U}_i^{k-1} + \frac{1}{8}\left(\mathbb{U}_{i+1}^{k-1} - \mathbb{U}_{i-1}^{k-1}\right),\end{aligned} \tag{4.5}$$

pour $i = 2, \ldots, N_0 2^{k-1} - 1$, et

$$\begin{aligned}\widehat{\mathbb{U}}_1^k &= \frac{1}{4}\left(5\mathbb{U}_1^{k-1} - \mathbb{U}_2^{k-1}\right), & & & \widehat{\mathbb{U}}_{N_0 2^k-1}^k &= \frac{1}{4}\left(\mathbb{U}_{N_0 2^{k-1}-1}^{k-1} + 3\mathbb{U}_{N_0 2^{k-1}}^{k-1}\right),\\ \widehat{\mathbb{U}}_2^k &= \frac{1}{4}\left(3\mathbb{U}_1^{k-1} + \mathbb{U}_2^{k-1}\right), & \text{et} & & \widehat{\mathbb{U}}_{N_0 2^k}^k &= \frac{1}{4}\left(-\mathbb{U}_{N_0 2^{k-1}-1}^{k-1} + 5\mathbb{U}_{N_0 2^{k-1}}^{k-1}\right).\end{aligned} \tag{4.6}$$

Notons que l'opérateur de prédiction ou de reconstruction P_{k-1}^k que l'on utilise ici respecte certaines contraintes de localité et de consistance, *i.e.* seules les valeurs aux mailles proches sont utilisées dans l'interpolation et le produit des deux opérateurs Prédiction et Projection est l'identité sur $\mathcal{S}^{k-1}$ ($P_{k-1}^k \circ P_k^{k-1} = \mathrm{Id}_{\mathcal{S}^{k-1}}$) (voir [27]).

Les deux opérations présentées ci-dessus sont illustrées dans la Fig. 4.1 où on présente quatre niveaux de raffinement de $k-2$ à $k+1$.

4.1.1.2 Détails et arbre de la représentation

À priori, la valeur prédite $\widehat{\mathbb{U}}_{2i}^k$ sur la maille M_{2i}^k est différente de la valeur originale $\mathbb{U}_{2i}^k$. Nous définissons le détail comme la différence entre la valeur moyenne initiale et la valeur moyenne approchée issue de l'opération Prédiction, à savoir

$$\mathbb{D}_i^{k-1} = \mathbb{U}_{2i}^k - \widehat{\mathbb{U}}_{2i}^k. \tag{4.7}$$

Remarquons que grâce à la propriété de consistance des opérateurs, le détail sur la maille M_{2i-1}^k n'est autre que l'opposé de $\mathbb{D}_i^{k-1}$. La relation (4.7) est équivalente à

$$\mathbb{D}_i^{k-1} = -\mathbb{U}_{2i-1}^k + \widehat{\mathbb{U}}_{2i-1}^k. \tag{4.8}$$

Nous n'avons donc besoin de connaître que les détails sur les mailles paires ou impaires sur un niveau donné k ainsi que les valeurs moyennes au niveau $k-1$, pour pouvoir calculer les valeurs

Fig. 4.1. Opérations de Projection et Prédiction sur quatre niveaux de grilles de $k-2$ à $k+1$. La Projection consiste à moyenner les valeurs sur les mailles fines pour avoir la valeur au niveau plus grossier. La Prédiction consiste à faire une interpolation polynomiale pour approcher la solution sur un niveau plus fin.

moyennes au niveau k. Le vecteur de détail $\mathbb{D}^{k-1}$ défini par (4.7) et la solution $\mathbb{U}^{k-1}$ au niveau $k-1$ ont la même taille. En conséquence, connaissant les valeurs $\mathbb{D}_i^{k-1}$ et $\mathbb{U}_i^{k-1}$ sur la grille grossière de niveau $k-1$, nous pouvons déterminer entièrement les valeurs initiales $\mathbb{U}_i^k$ sur la grille plus fine de niveau k. Plus précisément, à partir des $\mathbb{U}_i^{k-1}$ pour $i \in \{1, \ldots, N_0 2^{k-1}\}$, nous calculons les valeurs prédites $\widehat{\mathbb{U}}_{2i}^k$ suivant (4.5) et (4.6). On y rajoute ensuite les détails $\mathbb{D}_i^{k-1}$ pour retrouver les valeurs de $\mathbb{U}^k$:

$$\begin{aligned} \mathbb{U}_{2i}^k &= \widehat{\mathbb{U}}_{2i}^k + \mathbb{D}_i^{k-1}, \\ \mathbb{U}_{2i-1}^k &= 2\,\mathbb{U}_i^{k-1} - \mathbb{U}_{2i}^k. \end{aligned} \tag{4.9}$$

Itérer l'opération de Projection du niveau le plus fin K au niveau le plus grossier 0 en définissant le vecteur de détail $\mathbb{D}^k$ à chaque itération donne l'algorithme de codage qui fournit une représentation multi-échelle $\mathbb{M}^K = \mathcal{M}\mathbb{U}^K$ de la solution (Algo. 1). Ici l'opérateur $\mathcal{M}$ agissant sur la solution $\mathbb{U}^K$ sur la grille la plus fine est défini par

$$\mathcal{M} : \mathbb{U}^K \longmapsto \mathbb{M}^K = (\mathbb{U}^0, \mathbb{D}^0, \ldots, \mathbb{D}^{K-1}). \tag{4.10}$$

L'intérêt d'une telle représentation est que la régularité de la solution peut être mesurée par la taille des détails. Nous allons voir précisément comment analyser cette régularité.

L'opération inverse est l'algorithme de décodage est est présentée dans l'Algo. 2.

4.1.2 Seuillage

Comme présenté précédemment, l'idée principale de la méthode Multirésolution est de représenter la solution sur une hiérarchie de grilles dyadiques où on peut sélectionner localement le niveau de raffinement du maillage en fonction de la régularité locale de la solution. Cette régularité dépend en effet du détail (défini dans (4.7)) de la solution à chaque niveau. Plus précisément, si le détail est

Algorithme 1 : Codage

Données : $\mathbb{U}$ sur le niveau le plus fin K
Résultat : $\mathbb{M}^K$
début
  pour *niveau k décroissant de* K *à* 1 **faire**
    pour *i croissant de 1 à* $N_0 2^{k-1}$ **faire**
      Calculer $\mathbb{U}_i^{k-1}$ par (4.3)
    fin
    pour *i croissant de 1 à* $N_0 2^{k-1}$ **faire**
      Calculer $\widehat{\mathbb{U}}_{2i}^k$ par (4.5)
      $\mathbb{D}_i^{k-1} = \mathbb{U}_{2i}^k - \widehat{\mathbb{U}}_{2i}^k$
    fin
  fin
fin

Algorithme 2 : Décodage

Données : $\mathbb{M}^K$
Résultat : $\mathbb{U}$
début
  pour *niveau k croissant de* 1 *à* K **faire**
    pour *i croissant de* 1 *à* $N_0 2^{k-1}$ **faire**
      Calculer $\widehat{\mathbb{U}}_{2i}^k$ par (4.5)
    fin
    pour *i croissant de* 1 *à* $N_0 2^{k-1}$ **faire**
      $\mathbb{U}_{2i}^k = \widehat{\mathbb{U}}_{2i}^k + \mathbb{D}_i^{k-1}$
      $\mathbb{U}_{2i-1}^k = 2\mathbb{U}_i^{k-1} - \mathbb{U}_{2i}^k$
    fin
  fin
fin

en dessous d'un seuil, dépendant du niveau k, cela signifie que la solution est localement bien interpolée par un polynôme de degré 2 et donc qu'elle est considérée comme régulière. On dit dans ce cas que le détail est négligeable et on peut effectuer une compression des données en annulant ce détail. Ce détail est dit seuillé et la solution peut être représentée localement sur la grille plus grossière de niveau $k-1$. En revanche, si le détail est au dessus du seuil autorisé, cela signale une zone de forts gradients de la solution et par conséquent ce niveau fin doit être utilisé pour décrire la solution. Nous allons maintenant détailler l'algorithme de seuillage servant à évaluer la régularité locale de la solution.

Tout d'abord on définit une suite de paramètres de seuillage $(\varepsilon_k)_{k=0,\ldots,K-1}$ où le seuil ε_k pour le niveau k est défini par

$$\varepsilon_k = 2^{k-K}\varepsilon, \qquad k = 0, \ldots, K-1, \tag{4.11}$$

où ε est donné.

Plus la résolution est grossière, plus le paramètre de seuillage est petit. Pour mesurer la régularité de la solution sur la maille i au niveau k, nous comparons la norme du détail $\mathbb{D}_{\lfloor \frac{i+1}{2} \rfloor}^{k-1}$ calculé sur cette

maille au paramètre de seuillage correspondant. La norme du détail $\mathbb{D}^{k-1}_{\lfloor \frac{i+1}{2} \rfloor}$ est donnée par

$$\left\| \mathbb{D}^{k-1}_{\lfloor \frac{i+1}{2} \rfloor} \right\| := \max_{j=1,\ldots,3} \frac{\left| \mathbb{D}^k_i[j] \right|}{\max\limits_{l=1,\ldots,N_0 2^k} \left| \mathbb{U}^k_l[j] \right|}. \tag{4.12}$$

Ce choix de norme pondérée est basé sur les trois premières composantes de la solution $\mathbb{U}$. En effet, puisque nous ne traitons que les équations sur les variables $\rho, \rho v$ et ρY, les détails sont calculés uniquement sur ces composantes. Les deux autres composantes $\rho\Pi$ et $\rho\Sigma$ sont calculées à partir de ces dernières, leur régularité n'est donc pas significative.

Introduisons l'ensemble Γ_ε gouverné par ε

$$\Gamma_\varepsilon = \{(i,k) \text{ t.q. } \left\| \mathbb{D}^k_i \right\| \geq \varepsilon_k\}, \tag{4.13}$$

contenant les indices pour lesquels on ne seuille pas le détail. Dans les zones où la solution est considérée comme régulière, les détails négligeables sont mis à zéro. On définit un opérateur de seuillage $\mathfrak{T}_\varepsilon$ agissant sur la représentation multi-échelle $\mathbb{M}^K$:

$$\mathfrak{T}_\varepsilon(\mathbb{D}^k_i) = \begin{cases} \mathbb{D}^k_i & \text{si } (i,k) \in \Gamma_\varepsilon, \\ 0 & \text{sinon.} \end{cases} \tag{4.14}$$

Ainsi on définit un opérateur d'approximation à ε près, notée $\mathcal{A}_\varepsilon$, qui est la succession des opérations de codage, seuillage et décodage, à savoir

$$\mathcal{A}_\varepsilon = \mathcal{M}^{-1}\mathfrak{T}_\varepsilon\mathcal{M}. \tag{4.15}$$

On peut montrer que l'erreur introduite par l'approximation $\mathbb{U}^K_\varepsilon = \mathcal{A}_\varepsilon\mathbb{U}^K$ est contrôlée par le paramètre de seuillage ε (voir [27]), *i.e* il existe une constante C telle que

$$\left\| \mathbb{U}^K - \mathbb{U}^K_\varepsilon \right\|_{l^1} \leq C\varepsilon, \tag{4.16}$$

où la norme l^1 de $\mathbb{U}^k$ est définie par

$$\left\| \mathbb{U}^k \right\|_{l^1} = \Delta x^k \sum_{l=1}^{N_0 2^k} \left| \mathbb{U}^k_l \right|. \tag{4.17}$$

Dans la pratique, nous tirons avantage du fait que les irrégularités de la solution se localisent aux zones fines où se trouvent les détails non seuillés (voir [27]). La solution peut être alors représentée sur une grille adaptative obtenue à partir de la représentation multi-échelle $\mathbb{M}^K_\varepsilon = \mathfrak{T}_\varepsilon\mathcal{M}\mathbb{U}^K$. Nous allons voir maintenant comment obtenir cette grille adaptative.

4.1.3 Construction de la grille adaptative

Avant de définir la grille adaptative pour représenter la solution, nous avons besoin d'imposer à l'ensemble Γ_ε défini dans (4.13) une structure d'arbre et qui plus est une struture d'arbre graduel. Dans le cas des grilles dyadiques avec notre système de numérotation, un ensemble Γ est un arbre graduel sous la condition suivante :

$$(i,k) \in \Gamma \Longrightarrow (\lfloor \frac{i+1+l}{2} \rfloor, k-1) \in \Gamma, \qquad l = -1, 0, 1. \tag{4.18}$$

Dans la pratique, on ne construit pas directement un arbre graduel à partir des détails significatifs suivant (4.18). En effet, on passe à un ensemble de mailles actives dans les grilles dyadiques dont la

Algorithme 3 : Marquage des mailles actives

Données : L'ensemble des détails importants Γ_ε (équations (4.13))
Résultat : L'ensemble des mailles actives $\mathcal{M}_\varepsilon$ avec le l'arbre graduel Γ_ε correspondant
début
  pour *niveau k décroissant de* $K-1$ *à* 0 **faire**
    pour *i croissant de* 1 *à* $N_0 2^k$ **faire**
      si $(i,k) \in \Gamma_\varepsilon$ **alors**
        ajouter $(2i-1,k+1)$ et $(2i,k+1)$ dans $\mathcal{M}_\varepsilon$.
        si *i est pair* **alors**
          ajouter $(i-1,k),(i,k),(i+1,k)$ et $(i+2,k)$ dans $\mathcal{M}_\varepsilon$
        fin
        sinon
          ajouter $(i-2,k),(i-1,k),(i,k)$ et $(i+1,k)$ dans $\mathcal{M}_\varepsilon$
        fin
      fin
    fin
  fin
  pour *niveau k décroissant de* $K-2$ *à* 1 **faire**
    pour *i croissant de* 1 *à* $N_0 2^k$ **faire**
      si $(2i,k+1) \in \mathcal{M}_\varepsilon$ **alors**
        ajouter $(i-1,k)$,(i,k) et $(i+1,k)$ dans $\mathcal{M}_\varepsilon$.
        si *i est pair* **alors**
          ajouter $(i+2,k)$ dans $\mathcal{M}_\varepsilon$
        fin
        sinon
          ajouter $(i-2,k)$ dans $\mathcal{M}_\varepsilon$
        fin
      fin
    fin
  fin
fin

représentation des détails correspondants constitue un arbre graduel. Autrement dit, un arbre Γ est un arbre graduel si, sur les grilles dyadiques correspondantes, deux mailles voisines sont toujours situées soit sur le même niveau, soit sur des niveaux consécutifs (inférieur ou supérieur). Il n'y a pas plus d'un niveau d'écart entre deux mailles voisines sur la grille adaptative. Pour faire cela, si un détail est important on active les mailles correspondantes ainsi que certaines mailles adjacentes de niveaux plus grossier pour avoir une structure d'arbre graduel. Cette méthode de construction est détaillée dans l'Algo. 3 où $\mathcal{M}_\varepsilon$ représente l'ensemble des mailles actives.

Nous montrons dans la Fig. 4.2 l'exemple d'un arbre graduel. L'intérêt de travailler avec un tel arbre graduel apparaît lors de la reconstruction des valeurs $\widehat{\mathbb{U}}^k_{2i}$ sur la maille M^k_{2i} sur la grille de niveau k. En effet, cette reconstruction nécessite la connaissance des trois valeurs $\mathbb{U}^{k-1}_{i-1}$, $\mathbb{U}^{k-1}_i$ et $\mathbb{U}^{k-1}_{i+1}$ sur la grille de niveau $k-1$. Si on ne connaît pas ces valeurs pour faire l'interpolation polynomiale (4.5), il faut descendre au niveau inférieur, *i.e.* niveau $k-2$, pour les calculer. Cette opération locale peut être longue et nécessite quasiment une reconstruction globale. La structure d'un arbre graduel nous facilitera ce décodage local tout en évitant les calculs inutiles.

Nous allons maintenant construire une grille adaptative à partir de l'ensemble des mailles actives

$\mathcal{M}_\varepsilon$. Les valeurs moyennes sur la grille grossière ainsi que l'ensemble des détails non seuillés permettent de représenter de manière précise la solution sur un ensemble de mailles de différents niveaux. Nous parlons alors d'une grille adaptative où la taille locale de maille est déterminée par l'indice de la grille la plus fine contenant un détail non négligeable.

$$\mathcal{S}_\varepsilon = \Big\{(i,k) \in \{1,\dots,N_0 2^k\} \times \{0,\dots,K\} \quad \text{t.q.} \quad ((i,k) \in \mathcal{M}_\varepsilon \text{ et } (2i,k+1) \notin \mathcal{M}_\varepsilon) \quad \text{ou} \quad (i,K) \in \mathcal{M}_\varepsilon\Big\}. \tag{4.19}$$

Pour trouver la solution $\mathbb{U}_\varepsilon$ représentée sur cette grille adaptative $\mathcal{S}_\varepsilon$ à partir de la représentation multi-échelle $\mathbb{M}_\varepsilon^K = \mathfrak{T}_\varepsilon \mathcal{M} \mathbb{U}^K$, on applique l'algorithme de décodage partiel Algo. 4. Notons que la représentation de la valeur moyenne $\mathbb{U}_i^k$ sur le niveau intermédiaire k ne veut pas dire que la solution est considérée comme constante localement sur toute la longueur de la maille M_i^k. Elle signifie simplement que les valeurs moyennes sur les grilles plus fines $k+1,\dots,K$ peuvent être reconstituées à partir des valeurs moyennes au niveau grossier k grâce à l'opérateur de Prédiction P_k^{k+1} pour $k < K$. L'algorithme permettant à partir de la solution $\mathbb{U}_\varepsilon$ sur la grille adaptative de retrouver la représentation multi-échelle $\mathbb{D}_\varepsilon^K$ est appelé le codage partiel. Il est détaillé dans l'Algo. 5.

4.1.4 Évolution de l'arbre

Avant d'appliquer la méthode de Multirésolution à notre schéma volumes finis, nous regardons d'abord l'évolution de l'arbre Γ_ε d'un pas de temps à l'autre. En effet, la solution évolue au cours du temps ce qui implique le déplacement des singularités (zones irrégulières de la solution). Par conséquent, l'arbre de représentation de la solution est différent à chaque instant. Nous souhaitons alors faire évoluer l'arbre Γ_ε^n, adapté à la solution à l'instant t^n, pour obtenir un arbre Γ_ε^{n+1} adapté à la solution à l'instant t^{n+1}. Si c'est le cas, la grille adaptative correspondante est actualisée à chaque pas de temps et s'adapte dynamiquement à l'évolution de la solution.

Fig. 4.2. Arbre graduel : Les détails importants en gras $\mathbb{D}_i^k$ (resp. $\mathbb{D}_{i/2}^{k-1}$) signifient que les mailles correspondantes M_{2i-1}^{k+1} et M_{2i}^{k+1} (resp. M_{i-1}^k et M_i^k) sont activées. Les autres mailles autour sont aussi activées pour rendre l'arbre graduel : deux mailles consécutives dans la grille adaptative ont des tailles égales ou doubles l'une de l'autre.

Algorithme 4 : Décodage partiel

Données : $\mathbb{M}_\varepsilon^K \in \Gamma_\varepsilon$
Résultat : $\mathbb{U}_\varepsilon$
début
  pour *niveau* k *croissant de* 1 *à* K **faire**
    pour i *croissant de* 1 *à* $N_0 2^{k-1}$ **faire**
      si $(2i,k) \in \mathcal{S}_\varepsilon$ **alors**
        Calculer $\widehat{\mathbb{U}}_{2i}^k$ par (4.5)
        $\mathbb{U}_{2i}^k = \widehat{\mathbb{U}}_{2i}^k + \mathbb{D}_i^{k-1}$ $\mathbb{U}_{2i-1}^k = 2\mathbb{U}_i^{k-1} - \mathbb{U}_{2i}^k$
      fin
    fin
  fin
fin

Algorithme 5 : Codage partiel

Données : $\mathbb{U}_\varepsilon$
Résultat : $\mathbb{M}_\varepsilon^K$
début
  pour *niveau* k *décroissant de* K *à* 1 **faire**
    pour i *croissant de 1 à* $N_0 2^{k-1}$ **faire**
      si $(2i,k) \in \mathcal{S}_\varepsilon$ **alors**
        Calculer $\mathbb{U}_{2i}^k$ par (4.3)
        Calculer $\widehat{\mathbb{U}}_{2i}^k$ par (4.5) $\mathbb{D}_i^{k-1} = \mathbb{U}_{2i}^k - \widehat{\mathbb{U}}_{2i}^k$
      fin
    fin
  fin
fin

Tout cela est possible grâce à la nature hyperbolique de notre système. En effet pour un système d'équations hyperboliques, les singularités se déplacent dans un sens connu à une vitesse finie. Le déplacement au cours d'un pas de temps est contrôlé par la condition de stabilité CFL. Nous pouvons donc prédire un arbre plus grand à l'instant t^n qui englobe les zones éventuellement irrégulières de la solution à l'instant t^{n+1}. Pour cela, nous allons adopter la stratégie de prédiction de l'arbre de représentation suivant l'heuristique d'Harten [56]. Plus précisément, l'arbre graduel Γ_ε^n à l'instant t^n obtenu en appliquant l'opérateur $\mathcal{A}_\varepsilon = \mathcal{M}^{-1}\mathfrak{T}_{\Gamma_\varepsilon^n}\mathcal{M}$ à $\mathbb{U}^n$ peut être agrandi à un arbre $\widetilde{\Gamma}_\varepsilon^{n+1}$ contenant Γ_ε^n et Γ_ε^{n+1} tout en assurant l'estimation (4.16) pour $\mathcal{A}_\varepsilon = \mathcal{M}^{-1}\mathfrak{T}_{\widetilde{\Gamma}_\varepsilon^{n+1}}\mathcal{M}$. Il est toujours possible de prédire un tel arbre : il suffit de prendre comme arbre l'ensemble entier des indices. Une telle prédiction n'a aucun intérêt, du fait que l'on veut éviter de tout recalculer sur le niveau le plus fin ce qui ferait perdre la performance de la méthode MR. Nous cherchons alors une prédiction qui inclue le moins d'indices inutiles possibles à l'instant t^{n+1}.

La stratégie d'Harten consiste à agrandir l'arbre Γ_ε^n en fonction de la taille de ses détails. Plus précisément, l'arbre $\widetilde{\Gamma}_\varepsilon^{n+1}$ est obtenu en supposant que les voisins adjacents d'un détail non négligeable peuvent devenir non négligeables au pas de temps suivant, ils sont donc rajoutés dans l'arbre initial. De plus, si un détail est particulièrement grand, *i.e.* deux fois plus grand que le seuil correspondant, les subdivisions au niveau plus fin sont également rajoutées dans Γ_ε^n. Cette dernière opération a pour but

Algorithme 6 : Prédiction de l'arbre $\widetilde{\Gamma}_\varepsilon^{n+1}$

```
Données : l'arbre Γ_ε^n
Résultat : l'arbre Γ̃_ε^{n+1}
début
    pour niveau k décroissant de K à 1 faire
        pour i croissant de 1 à N_0 2^k faire
            si (i,k) ∈ Γ_ε^n alors
                ajouter (i-1,k), (i,k) et (i+1,k) dans Γ̃_ε^{n+1}
                si D_i^k ≥ 2ε^k alors
                    ajouter (2i-1,k+1) et (2i,k+1) dans Γ̃_ε^{n+1}
                fin
            fin
        fin
    fin
fin
```

de prévoir la croissance des singularités ou même une apparition éventuelle de nouvelles singularités.

Notons qu'une stratégie plus contraignante conduisant à des arbres plus « étouffés » est nécessaire pour obtenir une estimation d'erreur

$$\|\mathbb{U}^n - \mathbb{U}_\varepsilon^n\| < C\varepsilon. \tag{4.20}$$

Ici, $\mathbb{U}^n$ est la solution volumes finis sur la grille uniforme de niveau K et $\mathbb{U}_\varepsilon^n$ la solution Multirésolution reconstruite au niveau K au temps t^n (voir Cohen et *al.* [27]).

L'algorithme pour prédire le nouvel arbre de représentation est illustré dans l'Algo. 6. L'arbre $\widetilde{\Gamma}_\varepsilon^{n+1}$ est ensuite gradué en appliquant l'Algo. 3. Notons ici que grâce à la condition CFL inférieure à 1 pour un schéma explicite, le choix de ne rajouter qu'une seule maille à droite et à gauche d'un détail important dans l'arbre est justifié par le fait que les singularités ne se déplacent au plus que d'une maille en un pas de temps. En conséquence, elles sont toujours bien représentées à chaque instant au niveau où les détails sont significatifs, en utilisant la stratégie de prédiction d'arbre d'Harten. Dans le cas du schéma semi-implicite la CFL est beaucoup plus grande que 1 vis à vis des ondes rapides acoustiques. Cependant, ces dernières étant implicitées avec un schéma à trois points, elles sont régularisées très rapidement et sont donc bien représentées sur les niveaux plus grossiers de discrétisation (voir [81]).

L'algorithme final de la méthode MR est présenté dans l'Algo. 7 où T est le temps final de la simulation et $t^n = \sum_{p=0}^{n-1} \Delta t_p$. Dans cet algorithme, on remarque que l'opération de restriction de l'arbre $\widetilde{\Gamma}_\varepsilon^{n+1}$ à Γ_ε^{n+1} signifie que l'arbre Γ_ε^{n+1} est calculé mais on ne met pas effectivement les détails négligeables à zéro. Le seuillage effectif a lieu après la prédiction de $\widetilde{\Gamma}_\varepsilon^{n+2}$ à l'itération suivante. La solution finale est obtenue en reconstruisant la solution adaptative sur la grille la plus fine. Cette opération se fait une seule fois à la fin de la simulation pour avoir une solution finale précise. La cinquième opération dans la boucle en temps qui est l'évolution de la solution à chaque pas de temps sera présentée dans la suite où on utilise notre schéma volume fini Lagrange-Projection semi-implicite.

Dans la suite, on note $\mathbb{U}_j^n$ les valeurs de la solution discrétisée au temps t^n sur la grille adaptative

$$\{\mathbb{U}_j^n\}_{j=1,\dots,\mathcal{N}} := \{\mathbb{U}_i^k(t^n) \text{ t.q. } (i,k) \in \mathcal{S}_\varepsilon\}, \tag{4.21}$$

où $\mathcal{N}$ est la dimension de $\mathcal{S}_\varepsilon$. L'indice j ici correspond à l'ordre des mailles situées dans la grille adaptative parcourue dans le sens des x croissant, indépendamment du niveau de discrétisation.

Algorithme 7 : Algorithme de l'AMR

début
  Encoder la solution initiale puis définir l'arbre Γ_ε^0
  pour $t^n < T$ **faire**
    Prédiction de l'arbre $\widetilde{\Gamma}_\varepsilon^{n+1}$ grâce à l'algorithme 6
    Graduation de $\widetilde{\Gamma}_\varepsilon^{n+1}$ grâce à l'algorithme 3
    Calcul de la grille adaptative $\widetilde{\mathcal{S}}_\varepsilon^{n+1}$ grâce à (4.19)
    Décodage partiel de $\mathbb{U}_\varepsilon^n$ sur $\widetilde{\mathcal{S}}_\varepsilon^{n+1}$ grâce à l'algorithme 4
    Évolution de $\mathbb{U}_\epsilon^n$ à $\mathbb{U}_\varepsilon^{n+1}$ sur la grille adaptative $\widetilde{\mathcal{S}}_\varepsilon^{n+1}$
    Codage partiel de $\mathbb{U}_\epsilon^{n+1}$ sur $\widetilde{\Gamma}_\varepsilon^{n+1}$ grâce à l'algorithme 5
    Restriction de l'arbre $\widetilde{\Gamma}_\varepsilon^{n+1}$ à Γ_ε^{n+1}
  fin
  Décodage de $\mathbb{U}_\epsilon^N$ sur la grille la plus fine S^K
fin

4.2 Multirésolution appliquée aux schémas numériques

Nous avons vu dans la section précédente comment représenter la solution sur une grille adaptative et comment construire cette grille adaptative en fonction de la régularité locale de la solution. Dans cette partie, nous allons voir comment coupler cette technique d'adaptation et le schéma numérique Lagrange-Projection afin de travailler avec un maillage adaptatif.

En effet, le schéma numérique intervient dans la cinquième étape de la boucle en temps de l'algorithme de la MR, afin de faire évoluer la solution d'un pas de temps à l'autre. Il est donc indispensable de calculer les flux numériques aux interfaces de la grille. Par ailleurs du fait que la solution à l'instant t^n peut être approchée par sa représentation sur la grille adaptative, il suffit de calculer les flux uniquement sur la grille adaptative pour bien approcher la solution au temps suivant. C'est un des plus grands avantages de la méthode de Multirésolution : éviter les calculs inutiles dans les zones régulières en utilisant un maillage plus grossier.

Avant d'aller dans les détails du calcul des flux, nous rappelons d'abord les schémas numériques que nous utilisons pour appliquer la méthode MR, à savoir les schémas volumes finis Lagrange-Projection explicite et semi-implicite. Pour chaque schéma, nous redonnons le calcul du pas de temps.

4.2.1 Adaptation en espace pour le schéma Lagrange-Projection explicite et semi-implicite

Rappelons que sur un maillage uniforme où le pas d'espace Δx est constant, le schéma Lagrange-Projection explicite ou semi-implicite premier ordre peut s'écrire sous la forme conservative

$$\mathbb{U}_i^{n+1} = \mathbb{U}_i^n - \frac{\Delta t}{\Delta x}(\mathbb{F}_{i+1/2}^n - \mathbb{F}_{i-1/2}^n). \tag{4.22}$$

Notons que l'indice i correspond à la numérotation des mailles sur la grille uniforme. Le flux numérique dépend de la solution de la phase Lagrange :

- pour le schéma explicite

$$\mathbb{F}_{i+1/2}^n = (v_{i+1/2}^{n,*})^+ \widetilde{\mathbb{U}}_i^n + (v_{i+1/2}^{n,*})^- \widetilde{\mathbb{U}}_{i+1}^n + \mathbb{H}_{i+1/2}^{n,*}, \tag{4.23}$$

- pour le schéma semi-implicite

$$\mathbb{F}_{i+1/2}^n = (\widetilde{v}_{i+1/2}^{n,*})^+ \widetilde{\mathbb{U}}_i^n + (\widetilde{v}_{i+1/2}^{n,*})^- \widetilde{\mathbb{U}}_{i+1}^n + \mathbb{H}_{i+1/2}^{n,*}. \tag{4.24}$$

Dans (4.23), $\widetilde{\mathbb{U}}_i^n$ est la solution lagrangienne issue de la phase Lagrange (voir section 3.2.1 chapitre 3) sur la maille M_i sur la grille uniforme, et $\mathbb{H}_{i+1/2}^{n,*} = (0, -\Sigma_{i+1/2}^{n,*}, \Pi_{i+1/2}^{n,*})^T$. Les valeurs $v_{i+1/2}^{n,*}, \Sigma_{i+1/2}^{n,*}$ et $\Pi_{i+1/2}^{n,*}$ sont solutions du problème de Riemann à l'interface $i+1/2$ à l'instant t^n avec comme états gauches et droits respectivement $(v_i^n, v_{i+1}^n), (\Sigma_i^n, \Sigma_{i+1}^n)$ et (Π_i^n, Π_{i+1}^n). Dans le cas semi-implicite, la valeur $\widetilde{v}_{i+1/2}^{n,*}$ est celle du problème de Riemann à l'interface $i+1/2$ à la fin de la phase Lagrange (voir les formules pages 44 et 60).

Nous verrons dans l'annexe C, dédié à la description du code, qu'une fois défini le maillage adaptatif, le schéma numérique est le même en uniforme qu'en Multirésolution. En conséquence, lors de l'utilisation de la méthode MR, les deux schémas ci-dessus vont s'appliquer à la solution discrétisée sur la grille adaptative définie par (4.21) et peuvent s'écrire sous la même forme

$$\mathbb{U}_j^{n+1} = \mathbb{U}_j^n - \frac{\Delta t}{\Delta x_j}(\mathbb{F}_{j+1/2}^n - \mathbb{F}_{j-1/2}^n). \tag{4.25}$$

Ici, l'indice j correspond à la numérotation des mailles sur la grille adaptative (voir (4.21)). Les flux $\mathbb{F}_{j+1/2}^n$ dans (4.25) sont calculés à partir de la solution adaptative $\widetilde{\mathbb{U}}_j^n$ issue de la phase lagrangienne. Nous allons maintenant détailler comment évaluer ces flux.

4.2.2 Évaluation des flux et calcul de pas de temps

Il existe deux façons de calculer les flux numériques dans le contexte de l'application de la méthode MR aux schémas volumes finis : on peut les calculer à partir des valeurs reconstruites localement sur la grille la plus fine ou bien à partir des valeurs reconstruites sur la grille adaptative, *i.e.* utiliser directement la solution adaptative discrétisée.

La première méthode consiste à reconstruire localement les valeurs moyennes de la solution au niveau le plus fin en itérant localement l'opérateur de reconstruction (4.4). Pour cela, il faut connaître les valeurs reconstruites de la solution sur quatre mailles consécutives de même niveau autour de l'interface où on veut calculer le flux. Grâce à la propriété de graduation de la grille adaptative, deux mailles voisines sont de même taille ou bien appartiennent à deux niveaux de résolution consécutifs donc ont des tailles doubles l'une de l'autre. Il peut donc y avoir au maximum un rapport de quatre entre les tailles de quatre mailles consécutives. On se ramène à traiter dix configurations possibles pour les largeurs des mailles. Les détails de cette reconstruction locale se trouvent dans [81], section 7.2.2. Notons que cette stratégie est celle utilisée dans [27] pour obtenir l'estimation d'erreur (4.20).

La deuxième méthode consiste à calculer le flux numérique directement à partir de la solution adaptative entourant l'interface en considération. Nous n'avons donc pas besoin de faire une reconstruction locale, le schéma volumes finis s'applique directement sur la grille adaptative. Cette méthode est couramment utilisée et c'est d'ailleurs la seule possibilité dans les applications en dimension deux ou trois. Cette méthode est justifiée dans le cas où le schéma de référence sur la grille uniforme est d'ordre élevé pour compenser la perte de précision résultant de son utilisation sur des mailles de grande taille dans les zones de régularité de la solution. Une estimation d'erreur analogue à celle obtenue avec la méthode ci-dessus a récemment été obtenue pour cette stratégie par [76]. Dans le cadre de nos applications, puisqu'on utilise des schémas en espace avec reconstruction affine et limiteur de pente, on adopte donc cette méthode de calcul de flux (voir aussi [4]).

Rappelons quelques points importants dans la reconstruction avec pente de type MUSCL dans le cas de mailles de tailles variables. Soit $\mathbb{X}$ le vecteur des variables que l'on veut reconstruire. Ces dernières peuvent être primitives $\mathbb{X} := (\rho, Y, v)^T$ (phase Lagrange) ou conservatives $\mathbb{X} := (\rho, \rho Y, \rho v)^T$

(phase Projection). Pour chaque maille M_i, on calcule d'abord deux pentes $\wp_{j-1}$ et $\wp_j$ avec

$$\wp_j = 2\frac{\mathbb{X}_{j+1} - \mathbb{X}_j}{\Delta x_{j+1} + \Delta x_j}. \tag{4.26}$$

La pente $\mathcal{P}_j$ utilisée pour la reconstruction linéaire sur $\mathbb{X}_j$ est définie par

$$\mathcal{P}_j = limiteur(\wp_{j-1}, \wp_j), \tag{4.27}$$

où le *limiteur* peut être une des fonctions limiteurs classiques : minmod, Van Leer, Superbee ou Ultrabee.

Fig. 4.3. Reconstruction linéaire dans le cas de la Multirésolution où le pas d'espace est variable.

Les valeurs de reconstruction avec pente à gauche et à droite de l'interface $j+1/2$ sont

$$\mathbb{X}^L_{j+1/2} = \mathbb{X}_j + \frac{1}{2}\Delta x_j \mathcal{P}_j \qquad \text{et} \qquad \mathbb{X}^R_{j-1/2} = \mathbb{X}_j - \frac{1}{2}\Delta x_j \mathcal{P}_j. \tag{4.28}$$

Le flux $\mathbb{F}_{j+1/2}$ est calculé à partir des états $\mathbb{X}^L_{j+1/2}$ et $\mathbb{X}^R_{j+1/2}$.

Notons qu'il est facile de vérifier que les recontructions (4.28) sont équivalentes à celle du cas d'un maillage uniforme lorsque $\Delta x_{j+1} = \Delta x_j$ (voir page 51).

Le pas de temps lors de l'adaptation en espace doit être calculé de manière à vérifier localement la condition de stabilité qui dépend de la taille des mailles. Dans le cas d'un schéma Lagrange-Projection, la condition CFL que doit vérifier le pas de temps est

- Pour le schéma explicite

$$\Delta t^{exp} = \mathrm{CFL}^{exp}\frac{\min\limits_j(\Delta x_j)}{c}. \tag{4.29}$$

- Pour le schéma semi-implicite

$$\Delta t^{imp} = \min\left(\Delta t^{exp\#}, \mathrm{CFL}^{imp}\frac{\min\limits_j(\Delta x_j)}{c}\right). \tag{4.30}$$

Dans ces équations (4.29) et (4.30), c est la vitesse caractéristique maximale sur toutes les interfaces à l'intant t^n, c'est-à-dire

$$c = \max_j \max\left(|v^* - a\tau^*_L|^n_{j+1/2}, |v^* + a\tau^*_R|^n_{j+1/2}\right), \tag{4.31}$$

où a est le coefficient de relaxation, $v^{n,*}_{j+1/2}$ et $\tau^{n,*}_{j+1/2}$ sont la vitesse et le volume spécifique dans la solution du problème de Riemann à l'interface $j+1/2$ (voir (2.60)).

La CFL^{exp} est prise égale à 0.5 et la CFL^{imp} est beaucoup plus grande (10 ou 20). Notons que le pas de temps $\Delta t^{exp\#}$ dans la formule (4.30) n'est pas celui calculé par (4.29). C'est en effet le pas de temps calculé à partir du système par blocs du schéma implicite (voir section 3.5.2), basé sur la condition CFL^{exp} (voir section 3.5.6.2, théorème 3.6).

4.2.3 Traitement aux limites

Nous avons vu comment déterminer la grille adaptative pour représenter la solution de manière optimale, *i.e* la taille de maille sur cette grille adaptative varie en fonction de la régularité locale de la solution. Cette détermination est valide à l'intérieur du domaine, il faut aussi définir la taille des mailles fictives à l'extérieur du domaine. En effet, une analyse adaptative prenant compte les variations en temps est nécessaire pour imposer les conditions aux limites directement sur le maillage adaptatif. Dans la pratique, lorsqu'il y a une variation de conditions aux limites, on attribue obligatoirement le niveau $K-1$ aux deux premières mailles intérieures de la conduite. Cela permet d'avoir une précision de la solution à ces endroits. Les variations de la solution à l'intérieur de la conduite résultant des variations en temps sur les conditions aux limites sont ensuite captées automatiquement par l'analyse adaptative et pilotent le raffinement.

4.3 Résultats numériques

Afin de valider la méthode Multirésolution adaptée au schéma Lagrange-Projection, nous allons montrer dans ce chapitre quelques résultats numériques réalisés avec le schéma semi-implicite. Le premier cas-test est un tube à choc. Le deuxième cas-test consiste à considérer un écoulement avec variation de conditions limites à l'entrée d'une conduite inclinée d'un angle constant de 30 degrés. Pour le troisième cas-test, on applique les mêmes configurations d'écoulement et de conditions limites mais cette fois-ci dans une conduite en V afin de mettre en évidence que l'arbre de la Multirésolution suit efficacement la discontinuité de la solution. Finalement, nous montrons un cas-test réaliste, de type severe slugging, pour démontrer la robustesse de la méthode Lagrange-Projection. Pour mettre en valeur les performances de la méthode Multirésolution nous faisons une analyse précise du temps de calcul pour chacun de ces cas-tests.

Nous rappelons que toutes les simulations ont été réalisées avec un schéma avec reconstruction en espace (MUSCL) et Runge-Kutta en temps. Les lois de fermeture et le terme source seront précisés pour chaque cas-test spécifique. La condition CFL explicite est prise égale à 0.5 et la condition implicite est prise égale à 20.

Nous appelons la solution « adaptative » sur une grille de 512 mailles la solution obtenue avec la méthode MR sur une hiérarchie de grilles dont la plus fine a 512 mailles ($\mathcal{S}^K$). La solution adaptative est reconstruite en complétant par 0 tous les détails manquants dans Γ_ε et en appliquant l'algorithme de décodage. La solution « uniforme » correspondante est la solution obtenue avec le schéma uniforme sur la grille de 512 mailles. Les erreurs montrées dans les résultats numériques sont soit l'erreur relative d'adaptation due au seuillage soit l'erreur relative totale. La première est l'erreur relative entre la solution adaptative reconstruite sur $\mathcal{S}^K$ et la solution calculée sur cette grille uniforme. La deuxième est l'erreur entre la solution adaptative reconstruite sur $\mathcal{S}^K$ et la solution exacte. Cette dernière est approchée par la solution calculée sur une grille uniforme beaucoup plus fine que $\mathcal{S}^K$ dite

grille de référence. Par définition, l'erreur relative d'adaptation E^S est donnée par

$$E^S(h) = \frac{\sum_{i=1}^{N} \left| \mathbb{U}_i^u - \mathbb{U}_i^a \right|}{\sum_{i=1}^{N} \left| \mathbb{U}_i^u \right|}, \tag{4.32}$$

et l'erreur relative totale E^T est donnée par

$$E^T(h) = \frac{\sum_{i=1}^{N} \left| \frac{1}{m} \sum_{j=1}^{m} \mathbb{U}_{i,j}^h - \mathbb{U}_i^a \right|}{\sum_{i=1}^{N} \left| \frac{1}{m} \sum_{j=1}^{m} \mathbb{U}_{i,j}^h \right|}, \tag{4.33}$$

où

- $N = 2^K N_0$ est le nombre de mailles sur la grille uniforme la plus fine (niveau K),
- m est le ratio entre le nombre de mailles sur la grille de référence (fine) et celui sur la grille uniforme de niveau K,
- $\mathbb{U}_i^a$ est la solution adaptative reconstruite au centre de la i ième maille sur la grille uniforme de niveau K,
- $\mathbb{U}_i^u$ est la solution uniforme au centre de la i ième maille sur la grille uniforme de niveau K,
- $\mathbb{U}_{i,j}^h$ est la solution au centre de la $(i-1) * m + j$ ième maille sur la grille de référence.

Nous montrons ici uniquement les erreurs associées à la densité. Le gain CPU est le rapport entre le temps CPU de la simulation obtenu avec une grille uniforme et celui obtenu avec une grille adaptative.

4.3.1 Tubes à chocs

Tout d'abord nous testons la méthode PTL sur un problème de tube à choc. Il s'agit d'un problème de Riemann dans une conduite de 32 km de long, inclinée de 30 degrés vers le haut. Le mélange est initialement séparé en deux états au milieu de la conduite

$$\begin{pmatrix} \rho \\ Y \\ v \end{pmatrix}_L = \begin{pmatrix} 540 \\ 0.2 \\ -10 \end{pmatrix} \quad \text{et} \quad \begin{pmatrix} \rho \\ Y \\ v \end{pmatrix}_R = \begin{pmatrix} 400 \\ 0.4 \\ -10 \end{pmatrix}.$$

Dans la Fig. 4.4, nous montrons les résultats numériques obtenus avec la grille adaptative (MR) et avec la grille uniforme (U). Le coefficient d'implicitation du schéma Lagrange-Projection est pris égal à 0.7. La solution initiale dénotée par Ini évolue après 42 secondes de simulation. Trois ondes se propagent : une onde lente vers la gauche et deux ondes rapides, une vers la gauche et une vers la droite de la conduite. Pour la méthode MR, nous utilisons 5 niveaux de grilles avec le paramètre de seuillage $\varepsilon = 10^{-3}$. Les carrés représentent le niveau de raffinement utilisés localement par la méthode MR.

On constate que la solution adaptative est très similaire avec la solution uniforme sur toutes les composantes, tant sur l'onde lente que sur les ondes rapides. De plus, on voit bien qu'aux zones où la solution est régulière, on n'a que des grosses mailles de niveau 0. En revanche, là où la solution est irrégulière on utilise des mailles plus petites de niveau 1 au niveau le plus fin 5.

Fig. 4.4. Problème de tube à choc. Comparaison entre les simulations sur grille uniforme et sur grille adaptative. Schéma semi-implicite ordre 2, 4096 mailles, 5 niveaux de rafinement, $\varepsilon = 10^{-3}$.

	$\varepsilon = 10^{-2}$				$\varepsilon = 10^{-3}$			
	Rapport UNI/MR				Rapport UNI/MR			
$N(K)$	Loi *simple*		Loi *réaliste*		Loi *simple*		Loi *réaliste*	
	rCPU	rNbLoi	rCPU	rNbLoi	rCPU	rNbLoi	rCPU	rNbLoi
1024(3)	**5.68**	5.66	**5.25**	5.05	**4.30**	4.16	**4.40**	4.04
2048(4)	**11.76**	12.19	**10.04**	9.40	**7.19**	7.08	**7.62**	6.82
4096(5)	**20.21**	26.46	**19.57**	17.84	**14.08**	12.29	**13.36**	11.86

Tableau 4.1 Rapport temps CPU et nombre d'appels aux lois d'état - Uniforme (U) versus Multirésolution (MR). Schéma semi-implicite avec reconstruction de pente. Cas-test tube à choc.

Pour ce cas-test, nous montrons la performance de la méthode MR dans le Tab. 4.1 où on montre les gains de temps CPU (représenté par rCPU) pour deux types de loi hydrodynamique, l'une *simple* et l'autre *réaliste*, *i.e.* loi très compliquée (voir chapitre 6). Le paramètre $N(K)$ est le nombre de mailles utilisées sur la grille uniforme la plus fine et K est le nombre de niveaux. Nous constatons que nous obtenons des gains importants lors de l'utilisation de la méthode Multirésolution. Ces gains sont bien corrélés avec le rapport des nombres d'appels aux lois de fermeture. Nous verrons plus tard que pour des cas-tests plus complexes, par exemple avec des conditions aux limites, cette corrélation est plus nette pour la loi réaliste.

4.3.2 Variation de conditions limites, conduite inclinée

Nous considérons une conduite de longueur $L = 10000$ m, inclinée d'un angle de 30 degrés : $\mathcal{T}(x) = 30$ pour tout $x \in [0, L]$. Le diamètre de cette conduite est $D = 0.146$ m. Les conditions d'injection (temps de stabilisation, rampe) sont détaillées dans la Fig. 4.5. Les paramètres physiques et les conditions limites sont résumés dans la Fig. 4.5. Le paramètre *Area* est l'aire d'une section transversale à la conduite. Les résultats numériques sur une grille de 512 mailles sont illustrés dans la Fig. 4.6 où nous montrons la solution finale de la densité, la vitesse, la pression et le débit total le long de la conduite à l'instant $t = 2000$ s.

La performance de la méthode Multirésolution est validée tout d'abord par la qualité de la solution. Nous constatons que la solution adaptative obtenue avec une grille adaptative de 5 niveaux et un seuillage $\varepsilon = 0.001$ coïncide sur toutes ses composantes avec la solution uniforme. L'erreur relative d'adaptation associée à la densité est de l'ordre de 10^{-3}.

Géométrie de la conduite :	- Longueur	:	1000 m
	- Inclinaison :	:	30 degrés.
	- Diamètre	:	0.146 m
	- Nombre de mailles	:	512
Discrétisation :	- CFL explicite	:	0.5
	- CFL implicite	:	20
Source et Ordre :	- Schéma semi-implicite		
	- Avec source et recontruction de pente		
Modèles :	- Thermodynamique	:	liquide compressible
	- Hydrodynamique	:	glissement Zuber-Findlay IFP
Conditions opératoires :	- Durée de la stabilisation	:	100 s
	- Durée de la rampe	:	30 s
	- Condition limite amont sur le débit de gaz	:	0.1 kg/Area/s
		à	10 kg/Area/s
	- Condition limite amont sur le débit de liquide	:	20 kg/Area/s
	- Condition limite aval sur la pression	:	1 bar

Fig. 4.5. Détails de l'expérience 1. Conduite inclinée de 30 degrés.

Fig. 4.6. Expérience 1 - Écoulement dans une conduite inclinée de 30 degrés avec
- Solution à 2000 secondes de simulation avec un schéma MR de 512 mailles.

Dans ces figures, les carrés indiquent le niveau de raffinement, noté de 1 à 5 sur l'axe à droite, utilisé localement à chaque abscisse de la conduite. On remarque que la grille est finement raffinée à l'aval, là où il y a une forte variation de la densité et de la vitesse. En revanche, comme la solution est assez lisse ailleurs due à l'inclinaison constante de la conduite, on peut utiliser des mailles plus grossières ce qui permet d'accélérer le temps de calcul. C'est le deuxième point démontrant la performance de la méthode Multirésolution : le gain CPU entre la solution adaptative et la solution uniforme sur la grille la plus fine. Pour ce cas-test, nous avons un gain CPU de l'ordre de 2,8 pour 512 mailles.

Nous avons réalisé cette simulation avec plusieurs valeurs de paramètre du seuillage ε et avec différents nombres de mailles. Les résultats sont illustrés dans la Fig. 4.7 où on montre l'erreur relative d'adaptation en fonction du gain CPU, et dans la figure Fig. 4.8 où cette erreur est représentée en fonction de ε. On prend toujours 5 niveaux de raffinement pour la grille adaptative.

Le point A correspond à la solution adaptative montrée dans la Fig. 4.6 Nous constatons tout d'abord que pour un niveau de seuillage donné, nous n'obtenons pas le même gain CPU ni la même erreur relative d'adaptation pour différents nombres de mailles. Par exemple pour $\varepsilon = 0.001$, une grille adaptative de 512 mailles donne un gain CPU égal à 2.8 et une grille deux fois plus fine (1024 mailles) donne un gain CPU égal 4. Tandis que pour cette même valeur de seuillage et avec 2048 mailles, on obtient un gain CPU de l'ordre de 7.5 ce qui est très significatif. Pour un même ε, plus la grille est fine plus on gagne en temps CPU, et plus ε est petit plus la solution est précise grâce à la convergence de

Fig. 4.7. Erreur relative d'adaptation sur la densité en fonction du gain CPU, obtenue avec des seuillages et nombre de mailles différents - Conduite inclinée d'un angle constant de 30 degrés.

Fig. 4.8. Erreur relative d'adaptation sur la densité en fonction du seuillage ε, obtenue avec différents nombres de mailles - Conduite inclinée de 30 degrés.

l'erreur relative d'adaptation montrée dans la Fig. 4.8. Cette figure corrobore parfaitement l'estimation d'erreur théorique (4.20). Notons en revanche que lorsque ε est suffisamment petit on perd en temps de calcul (gain CPU inférieur à 1 pour les points à gauche de la droite verticale $x = 1$ dans la Fig. 4.7). Ceci est dû au fait que dans ce cas, aucun détail n'est seuillé et on n'utilise que les mailles les plus fines. Le temps de calcul est donc plus important avec cette grille uniforme fine à cause des opérations codage, décodage de la méthode Multirésolution. D'un autre coté, si on prend un seuillage trop grand pour avoir un gain CPU plus intéressant, on perd en précision. L'important est donc de trouver un seuillage et un nombre de niveaux de raffinement qui donnent un bon compromis entre la précision et la performance. En général, on prend $\varepsilon = 0.001$ avec 5 niveaux de rafinement.

Fig. 4.9. Expérience 2 - Écoulement dans une V-conduite - Solution obtenue après 2000 secondes de simulation sur un maillage adaptatif ayant 512 mailles sur la grille la plus fine.

4.3.3 Variation des conditions limites. Cas d'une conduite en V

Considérons une conduite de géométrie un peu plus compliquée où il y a un changement d'angle au milieu de la conduite. Elle a donc la forme de la lettre V.

$$\mathcal{T}(x) = \begin{cases} -30, & 0 \leq x \leq L/2 \\ +30, & L/2 < x < L. \end{cases} \tag{4.34}$$

Tous les autres paramètres ainsi que les conditions limites sont pris identiques au premier cas-test. Les résultats numériques avec 512 mailles sont présentés dans la Fig. 4.9. La solution adaptative est obtenue avec $\varepsilon = 0.001$ et 5 niveaux de raffinement.

Les solutions uniforme et adaptative sont comparables sur toutes les composantes. L'erreur relative d'adaptation est de l'ordre de 10^{-3}. Remarquons que, tout comme dans le premier cas-test, la solution est calculée sur un maillage plus grossier dans la plus grande partie du domaine et que les singularités sont bien suivies par l'arbre de la Multirésolution. Plus précisément, le maillage reste toujours raffiné (niveau 5) au voisinage de la discontinuité sur l'inclinaison au point bas de la conduite.

La méthode Multirésolution est une fois de plus favorable grâce à son efficacité et sa performance en voyant les résultats de gain CPU présentés dans la Fig. 4.10.

Fig. 4.10. Erreur relative d'adaptation de la densité en fonction de gain CPU - V-conduite.

Nous constatons que l'accélération de la méthode Multirésolution est particulièrement bonne pour ce cas-test pour un maillage ayant plus de 512 mailles avec des seuillages différents. Le point A représente la solution présentée dans la Fig. 4.9. Pour une erreur relative d'adaptation de 10^{-3} par rapport à la solution uniforme, nous avons un gain de temps CPU très prometteur à l'ordre de 3.3.

Nous avons calculé également l'erreur relative totale de la solution adaptative par rapport à une solution uniforme de référence ayant 8192 mailles. Les résultats numériques sont présentés dans la Fig. 4.11 où on montre ces erreurs relatives totales en fonction du gain CPU. La figure à gauche correspond au premier cas-test et celle à droite correspond au deuxième cas-test.

Pour chaque nombre de mailles, une ligne horizontale marque l'erreur relative totale du à la discrétisation obtenue en utilisant un maillage uniforme. On observe que dans tous les cas l'erreur relative totale pour la méthode MR converge vers cette limite inférieure. Par ailleurs, regardons de plus près les points de calcul indiqués par les symboles pleins •, ▲ et ♦ qui sont intéressants du point de vue à la fois de la précision et de la performance. En effet, comparons par exemple les points A et B sur la figure à droite. Le point A est calculé avec 512 mailles au niveau le plus fin et donne une erreur relative totale de l'ordre de 0.007. Son temps de calcul est identique (gain CPU égale à 1) à celui de la solution uniforme ayant le même nombre de mailles. Le point B de son côté est calculé sur une grille adaptative deux fois plus fine (1024 mailles) et donne un gain CPU égal à 4.5 par rapport à la solution uniforme de 1024 mailles. Notons que pour un schéma numérique d'ordre 2 en temps et en espace utilisant une grille uniforme, doubler le nombre de mailles revient à multiplier le temps CPU par 4. En conséquence, le point B est plus intéressant que le point A. En effet, le point B correspond à une simulation 4.5 fois plus rapide que la solution uniforme sur 1024 mailles, elle-même 4 fois plus coûteuse que la solution uniforme sur 512 mailles (point A). Il est en même temps plus précis car son erreur relative totale est plus faible (0.004 au lieu de 0.007). Nous obtenons donc pour un coût équivalent (gain $4.5/4 \approx 1.1$) une meilleure solution ($0.007/0.004 \approx 1.8$ fois plus précise). De la même manière, le point C est plus intéressant que le point A car il est plus rapide ($6.5/4 \approx 1.6$ fois) pour une précision équivalente.

On remarque encore ici que plus le seuillage ε est grand, plus on gagne en temps CPU mais plus on perd simultanément en précision. Là encore, il s'agit de choisir un niveau de seuillage réalisant un compromis acceptable. Un bon critère est de prendre un seuillage tel que l'erreur d'adaptation et l'erreur de discrétisation soient du même ordre (voir Schneider [18, 27].

Fig. 4.11. Erreur relative totale de la solution adaptative par rapport à la solution uniforme de référence (8192 mailles).

4.3.4 Severe Slugging

Nous traitons maintenant un cas-test réaliste et typique dans l'industrie pétrolière. Il consiste à modéliser le phénomène dit de *severe slugging*, c'est-à-dire la formation des bouchons au point bas de la conduite qui précède le riser vertical. Les deux tronçons sont assimilés à une seule conduite ayant pour longueur $L = 74$ m et pour diamètre $D = 0.05$ m. Son inclinaison est

$$\mathcal{T}(x) = \begin{cases} 0, & 0 \leq x \leq 60 \\ 90, & 60 < x < L. \end{cases} \tag{4.35}$$

Les détails correspondants à cette simulation sont résumés dans la Fig. 4.12. Pour ce cas-test, les conditions limites sont maintenues constantes à l'amont comme à l'aval. Le débit de gaz est beaucoup plus faible que le débit de liquide, c'est pour cette raison qu'on utilise la loi de glissement Zuber-Findlay IFP. C'est un cas-test que l'on a déjà présenté dans le cas uniforme (voir Fig. 3.19, page 82). Les résultats numériques sont présentés dans la Fig. 4.13. On constate que les résultats obtenus avec la méthode de Multirésolution sont comparables à ceux obtenus avec le maillage uniforme.

Géométrie de la conduite :	- Longueur	: 74 m
	- Inclinaison :	: 0° pour $x \in [0, 60]$ et 90° pour $x \in [60, 74]$
	- Diamètre	: 0.05 m
	- Nombre de mailles	: 256
	- Multirésolution	: 5 niveaux et $\varepsilon = 0.0001$
Discrétisation :	- CFL explicite : 0.5	
	- CFL implicite : 20	
Source et Ordre :	- Avec source et recontruction de pente	
Modèles :	- Thermodynamique	: liquide incompressible
	- Hydrodynamique	: glissement Zuber-Findlay IFP
Conditions opératoires :	- Durée de la stabilisation	: 0 s
	- Durée de la rampe	: pas de rampe
	- Condition limite amont sur le débit de gaz	: 1.96e-4 kg/Area/s
	- Condition limite amont sur le débit de liquide	: 7.854e-2 kg/Area/s
	- Condition limite aval sur la pression	: 1 bar

Fig. 4.12. Détails de l'expérience 3. Severe slugging.

Fig. 4.13. Expérience 3. Solutions uniforme (UNI) et adaptative (MR) pour le cas test severe slugging. Variations des débits du gaz et du liquide, de la vitesse et de la pression du mélange en fonction du temps à l'abscisse $x = 60$ m, *i.e* au point bas de la partie verticale de la conduite.

4.4 Conclusion sur la méthode de Multirésolution

Nous avons vu dans ce chapitre que l'utilisation du schéma Lagrange-Projection avec la stratégie Multirésolution permet d'accélérer le temps de calcul des simulations numériques de plusieurs cas-tests industriels. Les gains en temps de calcul sont significatifs et les résultats numériques sont très encourageants.

On constate que la qualité des résultats dépend non seulement du choix des paramètres de la Multirésolution, à savoir le paramètre de seuillage ε et le nombre de niveaux de raffinement K, mais aussi le nombre de mailles N utilisées sur la grille uniforme la plus fine. Les gains en temps CPU et en précision dépendent du choix judicieux de ces paramètres, il faut donc trouver un bon compromis entre eux. En général, nous utilisons 5 niveaux de grilles et ε est pris égal à 10^{-3}. On constate aussi que plus N est grand, plus le gain en temps CPU est important. La méthode Multirésolution commence à être intéressante dès que le nombre de mailles et le nombre de niveaux sont suffisamment grands.

Nous présentons dans le chapitre suivant le couplage de la stratégie pas de temps local avec la méthode Multirésolution, qui permet d'accélérer considérablement les simulations, notamment sur des cas-tests réalistes utilisant une loi hydrodynamique très compliquée.

Chapitre 5

Méthode de Pas de Temps Local (PTL)

Le gain en temps CPU apporté par la méthode Multirésolution est, comme on l'a vu au chapitre 4, déjà très appréciable. Dans ce chapitre, nous cherchons une accélération encore plus marquante pour les simulations en question.

Une des limitations de la méthode multirésolution présentée dans le chapitre précédent est que le pas de temps du schéma est contrôlé par la condition de stabilité (3.41) dans le cas explicite ou (3.144) dans le cas implicite, et donc par la taille de la plus petite maille dans la grille adaptative. Pour pallier cet inconvénient Oshers et Sanders [78] ont introduit le concept de schéma *à pas de temps local* où la solution évolue avec des pas de temps différents contrôlés localement par le pas d'espace. Depuis ces travaux précurseurs dans le cas d'une équation hyperbolique scalaire discrétisée sur une grille non uniforme mais fixe, plusieurs démarches ont été développées parmi lesquelles [30, 40, 89]. Dans le cadre des méthodes de multirésolution adaptative l'article fondateur de Müller et Stiriba [75] et les applications en multidimension [63,74], utilisent la notion d'un temps macroscopique adapté aux mailles les plus grosses de la grille. Ce pas de temps est décomposé en sous pas de temps élémentaires adaptés aux mailles les plus fines. A chaque pas de temps élémentaire les mailles appartenant à certains niveaux seulement dans la grille adaptative sont traitées. La solution est synchronisée sur la grille entière à chaque pas de temps macroscopique. En explicite les schémas utilisés sont Euler ou Muscl. En implicite, où les applications sont principalement stationnaires ou bien contrôlés par une condition sur le pas de temps, le schéma de Crank-Nicholson est utilisé. Une autre méthode de montée en ordre en temps par Runge Kutta, combinant le schéma adaptatif multirésolution et une stratégie de pas de temps local, est proposée plus récemment par Schneider dans [43].

(a) (b)

Fig. 5.1. (a) Méthode MR - Pas de temps constant. (b) Méthode PTL - Pas de temps variable.

Pour introduire la stratégie de pas de temps local dans l'algorithme Lagrange-Projection, nous sommes partis de la méthode proposée dans [75]. Nous présentons ici en détail cette méthode dans le cas explicite. Nous avons introduit la gestion du pas de temps de telle manière que la condition

de stabilité explicite (3.41) ou implicite (3.144) soit vérifiée à chaque pas de temps intermédiaire. Les applications que nous avons vues jusqu'alors dans la littérature utilisent un pas de temps fixe valable pour l'ensemble de la simulation [34, 75].

Le plan de ce chapitre est le suivant. Tout d'abord, nous présentons le principe de la méthode de pas de temps local sur deux niveaux puis nous la généralisons sur plusieurs niveaux de raffinement. À chaque fois, nous précisons le calcul des pas de temps correspondants. Ensuite nous montrons les résultats numériques obtenus avec cette méthode PTL. La dernière partie est consacrée à la convergence et la performance en temps CPU. Une extension de la méthode PTL vers le schéma semi-implicite et d'ordre supérieur sera aussi présentée.

5.1 Principe de la méthode de Pas de Temps Local

Nous avons vu que la méthode de Multirésolution permet d'avoir un maillage adaptatif où la taille des mailles est variable. L'idée principale de la méthode PTL est d'utiliser un pas de temps adapté localement au pas d'espace. Pour faire cela nous calculons, pour chaque niveau de discrétisation, un pas de temps correspondant tout en respectant la condition CFL locale pour assurer la stabilité du schéma numérique. On utilise donc un grand pas de temps pour une maille grossière, et un petit pas de temps pour une maille plus fine. On commence par faire évoluer la solution sur les mailles de petites tailles et on synchronise toutes les mailles à chaque grand pas de temps.

Pour clarifier cette idée, nous allons tout d'abord présenter la méthode PTL sur une grille adaptative de deux niveaux puis la généraliser sur une hiérarchie de grilles dyadiques quelconque.

5.1.1 Méthode PTL sur deux niveaux

Nous souhaitons faire évoluer la solution du temps t^n au temps t^{n+1} sur une grille adaptative de deux niveaux 0 et 1. Dans le but de généraliser la méthode PTL pour un nombre de niveaux quelconque dans la section suivante, nous allons désigner ces niveaux par les indices k et $k+1$. Le niveau $k=0$ est le niveau le plus grossier et $k+1=1$ est le niveau le plus fin. Dans la Fig. 5.2, on utilise les indices d'espace utilisés dans l'hiérarchie de grilles dyadiques uniformes (voir section 4.1.1 chaptire 4). Notons que ici la notation $\mathbb{U}^{n,k+1}_{2i-1,1}$ représente la solution discrétisée au temps intermédiaire $t^n_1 = t^n + \Delta t^{n,k+1}_0$ sur la maille numéro $2i-1$ de niveau $k+1$ de la grille adaptative. Les solutions discrétisées $\mathbb{U}^{n,k+1}_{2i-1}$ et $\mathbb{U}^{n,k+1}_{2i}$ sur les mailles de petit pas d'espace sont mises à jour en deux étapes :

- faire évoluer $\mathbb{U}^{n,k+1}_{2i-1}$ (resp. $\mathbb{U}^{n,k+1}_{2i}$) à $\mathbb{U}^{n,k+1}_{2i-1,1}$ (resp. $\mathbb{U}^{n,k+1}_{2i,1}$) en utilisant un pas de temps $\Delta t^{n,k+1}_0$,
- faire évoluer $\mathbb{U}^{n,k+1}_{2i-1,1}$ (resp. $\mathbb{U}^{n,k+1}_{2i,1}$) à $\mathbb{U}^{n+1,k+1}_{2i-1}$ (resp. $\mathbb{U}^{n+1,k+1}_{2i}$) en utilisant un pas de temps $\Delta t^{n,k+1}_1$.

Les solutions discrétisées $\mathbb{U}^{n,k}_{i-1}$ et $\mathbb{U}^{n,k}_{i-2}$ ayant un grand pas d'espace seront mises à jour avec un grand pas de temps $\Delta t^{n,k}_0 = \Delta t^{n,k+1}_0 + \Delta t^{n,k+1}_1$. Au temps t^{n+1}, toutes les solutions sont alors synchronisées.

On appelle $\Delta t^{n,k+1}_0$ et $\Delta t^{n,k+1}_1$ les micros pas de temps car ils sont associés aux petites mailles du niveau $k+1$. En revanche, $\Delta t^{n,k}_0$ est appelé le macro pas de temps car il est utilisé pour les grandes mailles du niveau k. Ces pas de temps sont calculés de manière à satisfaire les conditions de stabilité locales à chaque interface. Avant de préciser les calculs des pas de temps, nous allons tout d'abord présenter le schéma numérique correspondant en détaillant les flux numériques utilisés.

Fig. 5.2. Méthode de pas de temps local pour deux niveaux de raffinement. La solution discrétisée sur le niveau $k+1$ est mise à jour en deux étapes (deux micros pas de temps), tandis que celle sur le niveau k est mise à jour en une seule fois (avec un macro pas de temps).

5.1.1.1 Calculs des flux numériques et synchronisation en temps

L'une des difficultés majeures dans la méthode PTL est de savoir calculer et utiliser les flux numériques aux interfaces, notamment aux interfaces entre deux mailles de niveaux différents. En effet, lorsque les solutions discrétisées ne sont pas encore synchronisées (calculées) au même moment, il faut évaluer les flux numériques à partir des solutions existantes afin de les faire évoluer aux micros pas de temps suivants. Il existe dans la littérature plusieurs manières pour faire cela : celle de Oshers-Sanders [78] présentée dans la Fig. 5.3 et celle de Müller-Stiriba [75] présentée dans la Fig. 5.4.

Fig. 5.3. Calcul des flux suivant Oshers-Sanders sur un maillage adaptatif en espace et en temps.

La première méthode [78] consiste à utiliser « naïvement » les solutions discrétisées disponibles au pas de temps précédent. Plus précisément, au premier micro pas de temps, on calcule tous les flux et met à jour uniquement les solutions discrétisée à niveau fin $k+1$ ($\mathbb{U}_{2i-1}^{n,k}$ et $\mathbb{U}_{2i}^{n,k}$). Au deuxième micro pas de temps, aux interfaces où il y un changement de niveau ($i-1/2$ ou $2i-3/2$), on recalcule les flux en utilisant la solution à l'instant t^n sur la maille grossière de niveau k ($\mathbb{U}_{i-1}^{n,k}$) et celle à l'instant $t^n + \Delta t_0^{n,k+1}$ sur la maille fine de niveau $k+1$ ($\mathbb{U}_{2i-1,1}^{n,k}$). Dans [89], il est démontré que cette méthode n'est pas consistante.

Fig. 5.4. Calcul des flux suivant Müller-Stiriba sur un maillage adaptatif en espace et en temps.

Dans ce rapport, nous utilisons la deuxième méthode qui donne un flux précis en calculant au pas de temps intermédiaire les solutions discrétisées sur les mailles grossières de niveau k adjacentes à une maille plus fine de niveau $k+1$. Au premier micro pas de temps, ces solutions sont mises à jour avec un petit pas de temps $\Delta t_0^{n,k+1}$. Au moment de la synchronisation (au deuxième micro pas de temps), elles seront toutes « corrigées » en utilisant les vrais flux aux interfaces. Avant de décrire le schéma numérique correspondant, nous introduisons d'abord quelques ensembles suivants en reprenant les notation de [75] :

- l'ensemble des mailles du niveau l faisant partie de la grille adaptative

$$\widetilde{\mathcal{I}}^l = \{(i,l) \text{ t.q. } (i,l) \in \widetilde{\mathcal{S}}_\varepsilon^{n+1}\}, \tag{5.1}$$

- l'ensemble $(\mathcal{C}^l)_{l=k,k+1}$ des indices des mailles de niveau l qui peuvent évoluer avec *un* pas de temps $(\Delta t_p^{n,l})_{0\le p\le k-l}$,

$$\mathcal{C}^l := \Big\{(i,l) \in \widetilde{\mathcal{S}}_\varepsilon^{n+1} \text{ t.q. } \nexists (2i+m, l+1) \in \widetilde{\mathcal{S}}_\varepsilon^{n+1}, -2s-1 \le m \le 2s\Big\}, \tag{5.2}$$

  où s dépend du nombre de points nécessaires ou « stencil » pour calculer les flux. Un schéma utilisant des flux à deux points correspond à prendre $s=1$.

- l'ensemble $(\bar{\mathcal{C}}^l)_{l=k,k+1}$ le complément de $(\mathcal{C}^l)_{l=k,k+1}$ que l'on appelle les zones de transitions,

$$\bar{\mathcal{C}}^l := \widetilde{\mathcal{I}}^l \backslash \mathcal{C}^l. \tag{5.3}$$

Nous rappelons que dans (5.2) et (5.3), $\widetilde{\mathcal{S}}_\varepsilon^{n+1}$ est la grille adaptative graduelle définie par (4.19). Notons que l'ensemble $\mathcal{C}^{k+1}$ contient toutes les mailles du niveau le plus fin $k+1$, aucune maille de ce niveau est exclue. Dans notre exemple nous avons $\mathcal{C}^{k+1} = \{(2i-1, k+1), (2i, k+1)\}$ et la zone de transition associée au niveau $k+1$ est vide, *i.e.* $\bar{\mathcal{C}}^{k+1} = \emptyset$. Les ensembles $\mathcal{C}^k$ et $\bar{\mathcal{C}}^k$ associés au niveau k sont : $\mathcal{C}^k = \{(i-2, k)$ et $\bar{\mathcal{C}}^k = \{(i-1, k)\}$.

L'évolution de la solution se décompose alors en deux étapes. Dans un premier temps, on fait évoluer les solutions discrétisées sur toutes les mailles de $\mathcal{C}^{k+1}$ et celles de $\bar{\mathcal{C}}^k$ en utilisant un micro pas de temps $\Delta t_0^{n,k+1}$. Cela se traduit par

$$\mathbb{U}_{i,1}^{n,l} = \mathbb{U}_i^{n,l} - \frac{\Delta t_0^{n,k+1}}{\Delta x_i}\left(\mathbb{F}_{i+1/2,0}^{n,l} - \mathbb{F}_{i-1/2,0}^{n,l}\right), \quad l = k, k+1, (i,l) \in \mathcal{C}^{k+1} \bigcup \bar{\mathcal{C}}^k. \tag{5.4}$$

Fig. 5.5. Deux étapes de calcul : les arcs correspondent aux mailles où on met à jour la solution. En haut : mise à jour avec (5.4). En bas : mise à jour avec (5.5).

L'étape suivante consiste à mettre à jour les solutions discrétisées sur toutes les mailles de $\mathcal{C}^{k+1}$ et de $\bar{\mathcal{C}}^k$ avec un pas de temps $\Delta t_1^{n,k+1}$ à partir des solutions calculées à l'étape précédente, et celles de $\mathcal{C}^k$ avec le macro pas de temps $\Delta t_0^{n,k}$ à partir des solutions discrétisées au temps t^n, à savoir

$$\begin{aligned}
\mathbb{U}_i^{n+1,l} &= \mathbb{U}_{i,1}^{n,l} - \frac{\Delta t_1^{n,l}}{\Delta x_i}\left(\mathbb{F}_{i+1/2,1}^{n,l} - \mathbb{F}_{i-1/2,1}^{n,l}\right), \quad l = k, k+1, (i,l) \in \mathcal{C}^{k+1} \bigcup \bar{\mathcal{C}}^k, \\
\mathbb{U}_i^{n+1,k} &= \mathbb{U}_i^{n,k} - \frac{\Delta t_0^{n,k}}{\Delta x_i}\left(\mathbb{F}_{i+1/2,0}^{n,k} - \mathbb{F}_{i-1/2,0}^{n,k}\right), \quad (i,k) \in \mathcal{C}^k.
\end{aligned} \tag{5.5}$$

Dans (5.4) et (5.5), Δx_i est la taille de la maille M_i contenant la solution discrétisée $\mathbb{U}_i^n$ et $\mathbb{F}_{i+1/2,p}^{n,l}$ est le flux calculé à l'interface $i+1/2$ entre deux solutions $\mathbb{U}_{i,p}^{n,l}$ et $\mathbb{U}_{i+1,p}^{n,l}$.

Le schéma de mise à jour de la solution est illustré dans la Fig. 5.5. On remarque que l'on ne calcule les flux entre deux grosses mailles qu'une seule fois, *i.e.* $\mathbb{F}_{i-5/2,0}^{n,k}$ reste inchangé et $\mathbb{F}_{i-3/2,1}^{n,k} = \mathbb{F}_{i-3/2,0}^{n,k}$. De plus, par convention nous avons $\mathbb{F}_{i-1/2,p}^{n,k} = \mathbb{F}_{2i-3/2,p}^{n,k+1}$, pour $p = 0, 1$ aux interfaces entre deux niveaux.

En injectant (5.5) dans (5.4) nous obtenons une formulation conservative des flux pour chaque pas de temps $\Delta t_p^{n,l}$, $l = k, k+1$ et $0 \leq p \leq k-l$:

$$\mathbb{U}_i^{n+1,l} = \mathbb{U}_i^{n,l} - \frac{\Delta t_p^{n,l}}{\Delta x_i}\left(\bar{\mathbb{F}}_{i+1/2,p}^{n,l} - \bar{\mathbb{F}}_{i-1/2,p}^{n,l}\right), \quad (i,l) \in \widetilde{\mathcal{S}}_\varepsilon^{n+1}. \tag{5.6}$$

avec les flux

Fig. 5.6. Conservation de flux à l'interface entre deux mailles de deux niveaux différents.

- sur l'interface des mailles de niveau fin $k+1$

$$\bar{\mathbb{F}}^{n,k+1}_{i+1/2,p} = \mathbb{F}^{n,k+1}_{i+1/2,p} \tag{5.7}$$

- sur l'interface $i+1/2$ où il y a un changement de niveaux, pour la maille de niveau grossier k

$$\bar{\mathbb{F}}^{n,k}_{i+1/2,p} = \frac{\Delta t_0^{n,k+1}}{\Delta t_0^{n,k}} \mathbb{F}^{n,k+1}_{i+1/2,0} + \frac{\Delta t_1^{n,k+1}}{\Delta t_0^{n,k}} \mathbb{F}^{n,k+1}_{i+1/2,1} \tag{5.8}$$

- sur l'interface entre deux mailles de niveau grossier k

$$\bar{\mathbb{F}}^{n,k}_{i+1/2,p} = \mathbb{F}^{n,k}_{i+1/2,0} \tag{5.9}$$

Par conséquent, nous pouvons appliquer la méthode PTL à notre schéma conservatif Lagrange-Projection en utilisant (5.6).

5.1.1.2 Calcul des pas de temps

Dans cette section, nous allons présenter comment obtenir les pas de temps dans le cas de la méthode de pas de temps local. Sur les mailles fines de niveau $k+1$, la solution est mise à jour en deux étapes en utilisant deux micros pas de temps. Ces derniers peuvent être calculés indépendamment l'un de l'autre pour respecter la critère de stabilité du schéma numérique utilisé. Notre stratégie d'adaptation du pas de temps diffère de celles actuellement illustrées dans la littérature [33, 43, 75]

Dans le cadre des simulations numériques des écoulements diphasiques utilisant le schéma Lagrange-Projection conservatif, nous rappelons que le pas de temps dépend des vitesses cractéristiques aux interfaces (on se réfère aux formules (3.46), (3.47) pour le cas uniforme et (4.29) pour le casMultirésolution). Notons $c^{n,l}_{i+1/2,p}$, $p=0,1$, $l=k,k+1$, la vitesse caractéristique maximale à l'interface $i+1/2$ au temps t^n_p, donnée par (voir formule (2.60) page 25)

$$c^{n,l}_{i+1/2,p} = \max\left(|v^* - a\tau^*_L|^{n,l}_{i+1/2,p}, |v^* + a\tau^*_R|^{n,l}_{i+1/2,p}\right), \tag{5.10}$$

On obtient les formules suivantes pour les pas de temps :

- le premier micro pas de temps

$$\Delta t_0^{n,k+1} = \mathrm{CFL}^{exp} \min_{\substack{(i,l)\in\widetilde{\mathcal{S}}^{n+1}_{\varepsilon},l=k,k+1 \\ \text{ou } (i+1,l)\in\widetilde{\mathcal{S}}^{n+1}_{\varepsilon}}} \frac{\min\left(\Delta x_i, \Delta x_{i+1}\right)}{c^{n,l}_{i+1/2,0}}, \tag{5.11}$$

- le deuxième micro pas de temps

$$\Delta t_1^{n,k+1} = \min\Big(\Delta t_0^{n,k+1}, \mathrm{CFL}^{exp} \min_{\substack{(i,l)\in\mathcal{C}^{k+1}\bigcup\bar{\mathcal{C}}^k, l=k,k+1 \\ \text{ou } (i+1,l)\in\mathcal{C}^{k+1}\bigcup\bar{\mathcal{C}}^k}} \frac{\min\left(\Delta x_i, \Delta x_{i+1}\right)}{c_{i+1/2,p}^{n,l}}\Big), \tag{5.12}$$

- le macro pas de temps

$$\Delta t_0^{n,k} = \Delta t_0^{n,k+1} + \Delta t_1^{n,k+1}. \tag{5.13}$$

On remarque que le premier micro pas de temps est calculé sur toutes les interfaces de la grille adaptative. Tandis que le deuxième micro pas de temps est calculé uniquement sur les interfaces des mailles du niveau $k+1$ ainsi que des mailles dans la zone de transition. De plus, les micros pas de temps ne font que décroître, tout en respectant la condition de stabilité locale. En effet, cette propriété permet d'assurer aussi la stabilité dans les zones contenant les mailles plus grossières.

5.1.2 Méthode PTL - Cas général

Considérons maintenant une grille adaptative contenant des mailles de $K+1$ niveaux différents. Les mailles les plus grossières sont associées au niveau 0, elles ont la longueur Δx^0. Les mailles les plus fines sont associées au niveau K, elles ont la longueur Δx^K. Nous rappelons que (voir section 4.1.1 chapitre 4) sur l'intervalle $[0, L]$, la taille des mailles de niveau k, $0 \le k \le K$, est donnée par

$$\Delta x^k = \frac{L}{N_0 2^k}, \tag{5.14}$$

où N_0 est le nombre de mailles sur la grille la plus grossière. Par convention, l'interface entre deux mailles consécutives de niveaux différents peut être représentée de deux façons : $x_{2i-1/2}^{k+1} = x_{i-1/2}^k$.

Nous souhaitons faire évoluer la solution discrétisée sur cette grille adaptative dont le pas de temps dépend localement de la taille des mailles, *i.e.* deux mailles de pas d'espace différents vont évoluer avec des pas de temps différents, quitte à synchroniser la solution à la fin d'un grand pas de temps. On définit le macro pas de temps $\Delta t_0^{n,0}$ celui associé aux mailles les plus grossières de niveau 0. Les mailles de niveau k, $0 < k \le K$, vont évoluer en 2^k pas de temps intermédiaires, noté $\Delta t_p^{n,k}$, $0 \le p \le 2^k - 1$. Les pas de temps associés aux mailles les plus fines K s'appellent les micros pas de temps. Nous introduisons alors les temps intermédiaires associés à l'évolution d'une maille de niveau $k > 0$, pour $p = 1, \ldots, 2^k$,

$$t_p^{n,k} = t_{p-1}^{n,k} + \Delta t_{p-1}^{n,k}. \tag{5.15}$$

On associe à cette notation celle pour la solution discrétisée sur une maille i du niveau k au temps intermédiaire $t_p^{n,k}$: $\mathbb{U}_{i,p}^{n,k}$. Par définition, au début et à la fin de chaque macro pas de temps nous avons pour tout niveau k

$$t_0^{n,k} = t^n \quad \text{et} \quad t_{2^k}^{n,k} = t^{n+1}. \tag{5.16}$$

De plus, nous pouvons établir une relation entre les pas de temps correspondant aux deux niveaux consécutifs, à savoir

$$\Delta t_p^{n,k} = \Delta t_{2p}^{n,k+1} + \Delta t_{2p+1}^{n,k+1}. \tag{5.17}$$

Pour clarifier ces notations, nous montrons dans la Fig. 5.7 les pas de temps intermédiaires et les temps intermédiaires sur une grille adaptative de 4 niveaux (0 à 3).

Fig. 5.7. Exemple de pas de temps intermédiaire sur grille de 4 niveaux.

5.1.2.1 Synchronisation en temps

Nous allons maintenant décrire l'algorithme pour mettre à jour la solution du temps t^n au temps t^{n+1} dans le cas général où il y a $K+1$ niveaux de raffinement. Le principe de cet algorithme consiste à faire évoluer d'abord la solution sur les mailles du niveau le plus fin, puis successivement celles sur les niveaux plus grossiers. Cela permet de calculer les flux aux interfaces de changement de niveaux suivant (5.8).

Au temps $t_p^{n,K}$, on définit le niveau de synchronisation k_p, $p = 0, \ldots, 2^K$ par

$$k_p := \min\left\{k, 0 \leq k \leq K \text{ t.q. } p \bmod 2^{K-k} = 0\right\}. \tag{5.18}$$

Au temps $t_p^{n,K}$, on fait évoluer la solution sur toutes les mailles de la grille adaptative de niveau compris entre k_p et K plus les mailles de $\bar{\mathcal{C}}^{k_p-1}$ (voir définition ci-après) afin d'assurer la synchronisation à chaque temps intermédiaire. Il est évident que $k_0 = k_{2^K} = 0$ puisqu'aux temps $t_0^{n,K}$ et $t_{2^K}^{n,K}$ toutes les solutions de niveau K à 0 doivent évoluer.

Au micro pas de temps intermédiaire p, $p \in \{0, \ldots, 2^K - 1\}$, on associe l'indice des pas des temps intermédiaires correspondant aux niveaux concernés par cette étape, c'est-à-dire les niveaux compris entre k_p et K. On peut vérifier que cet indice est donné par la fonction $q(p,k)$ suivante

$$q(p,k) = \left\lfloor \frac{p}{2^{K-k}} \right\rfloor, \tag{5.19}$$

où $\lfloor p \rfloor$ dénote la partie entière de p.

Comme nous avons vu précédemment, au début de chaque macro pas de temps nous devons définir les ensembles $\mathcal{C}^l$, $l = 0, \ldots, K$, des mailles de niveau l qui peuvent évoluer avec *un* pas de temps $\Delta t_p^{n,l}$, $p = 0, \ldots, 2^l - 1$,

$$\begin{aligned} \mathcal{C}^K &:= \left\{(i,K) \text{ t.q. } (i,K) \in \widetilde{\mathcal{S}}_\varepsilon^{n+1}\right\}, \\ \mathcal{C}^l &:= \left\{(i,l) \in \widetilde{\mathcal{S}}_\varepsilon^{n+1} \text{ t.q. } \nexists (2i+m, k+1) \in \widetilde{\mathcal{S}}_\varepsilon^{n+1}, -2s-1 \leq m \leq 2s\right\}, \quad l < K, \end{aligned} \tag{5.20}$$

ainsi que les ensembles complémentaires $\bar{\mathcal{C}}^l$ représentant les zones de transition

$$\bar{\mathcal{C}}^l := \widetilde{\mathcal{I}}^l \backslash \mathcal{C}^l, \tag{5.21}$$

où l'ensemble $\widetilde{\mathcal{I}}^l$ est défini par (5.1).

Nous introduisons en plus les ensembles $\mathcal{F}_p$, $p = 0, \ldots, 2^K - 1$, des indices des flux à calculer au début du temps intermédiaire $t_p^{n,K}$

$$\mathcal{F}_p := \left\{ (i+1/2, k) \text{ t.q. } \left((i,k) \in \widetilde{\mathcal{S}}_\varepsilon^{n+1} \text{ ou } (i+1,k) \in \widetilde{\mathcal{S}}_\varepsilon^{n+1} \right) \text{ et } k \geq k_p \right\}. \tag{5.22}$$

Nous montrons dans la Fig. 5.8 les étapes de la mise à jour de la solution discrétisée au cours de chaque micro pas de temps. Au temps intermédiaire $t_p^{n,K}$, on doit calculer l'ensemble des flux $\mathcal{F}_p$ qui sont représentés par les flèches pleines. Afin de mettre à jour la solution dans les mailles de niveaux supérieur ou égal au niveau de synchronisation courant et celles dans les zones de transition, nous utilisons ces flux ou ceux calculés précédemment. Ces derniers sont représentés par les flèches creuses.

Rappelons que suivant la méthode de Multirésolution, la grille adaptative évolue au cours de chaque pas de temps afin de suivre le déplacement des singularités. Dans la méthode PTL, il est montré dans [75] que prédire l'évolution de la grille d'un pas de temps macro à l'autre conduisait à des grilles adaptatives beaucoup trop raffinées. Une singularité pouvant se déplacer de 2^K petites mailles. Il est donc nécessaire d'opérer des remaillages partiels, aux pas de temps intermédiaires. Plus précisément, tous les deux pas de temps intermédiaires, le niveau de synchronisation k_p est strictement inférieur à K et on remaille partiellement la grille pour les niveaux $k = k_p + 1$ à K.

Fig. 5.8. Évolution de la solution au cours de 4 premiers pas de temps intermédiaires $p = 1, \ldots, 4$ sur une grille adaptative de 4 niveaux (0 à 3). Au temps $t_p^{n,K}$, k_p est le niveau de synchronisation et $\mathcal{F}_{p-1}$ est l'ensemble les flux à calculer. a) Premier micro pas de temps b) Deuxième micro pas de temps c) Troisième micro pas de temps d) Quatrième micro pas de temps.

Algorithme 8 : Synchronisation de la solution au temps intermédiaire $t_{p+1}^{n,K}$, $p = 0, \ldots, 2^K - 1$

Données : Les solutions discrétisées mises à jour après le temps intermédiaire précédent $t_p^{n,K}$. Remaillage et recalcul des ensembles $\mathcal{C}^l$,$\bar{\mathcal{C}}^l$, $l = 0, \ldots, K$ si p est pair.
Résultat : Les solutions discrétisées au temps intermédiaire $t_{p+1}^{n,K}$
début
  1. Calcul des flux
  Calculer l'ensemble $\mathcal{F}_p$ suivant (5.22)
  si $p = 0$ **alors**
    Calculer $\mathbb{F}_{i+1/2,0}^{n,k} \in \mathcal{F}_p$
  fin
  sinon
    pour *niveau* k *décroissant de* K *à* k_p **faire**
      pour i *croissant de* 0 *à* $N_0 2^k + 1$ **faire**
        si $(i + 1/2, k) \in \mathcal{F}_p$ **alors**
          si $k \geq k_{p-1}$ **alors**
            Calculer $\mathbb{F}_{i+1/2,p}^{n,k}$ à partir des données au temps intermédiaire précédent
          fin
          sinon
            $\mathbb{F}_{i+1/2,p}^{n,k} = \mathbb{F}_{i+1/2,p-1}^{n,k}$
          fin
        fin
      fin
    fin
  fin
  2. Évolution en temps de la solution
  pour *niveau* k *décroissant de* K *à* k_{p+1} **faire**
    pour $(i,k) \in \mathcal{C}^k$ **faire**
      $$\mathbb{U}_{i,p+1}^{n,k} = \mathbb{U}_{i,p}^{n,k} - \frac{\Delta t_{q(p,k)}^{n,k}}{\Delta x_i} \left(\mathbb{F}_{i+1/2,p}^{n,k} - \mathbb{F}_{i-1/2,p}^{n,k} \right)$$
    fin
    pour $(i, k-1) \in \bar{\mathcal{C}}^{k-1}$ **faire**
      $$\mathbb{U}_{i,p+1}^{n,k-1} = \mathbb{U}_{i,p}^{n,k-1} - \frac{\Delta t_{q(p,k)}^{n,k}}{\Delta x_i} \left(\mathbb{F}_{i+1/2,p}^{n,k-1} - \mathbb{F}_{i-1/2,p}^{n,k-1} \right)$$
    fin
  fin
fin

5.1.2.2 Calculs des pas de temps

De la même manière que dans le cas à deux niveaux (voir 5.1.1.2), nous devons calculer les pas de temps intermédiaires de manière à satisfaire les conditions de stabilité locale. Dénotons $c_{i+1/2,p}^{n,k}$, pour $p \in \{0, \ldots, 2^K - 1\}$, la vitesse caractéristique maximale sur l'interface $i + 1/2$ au temps $t_p^{n,k}$, donnée par

$$c_{i+1/2,p}^{n,k} = \max \left(|v^* - a\tau_L^*|_{i+1/2,p}^{n,k}, |v^* + a\tau_R^*|_{i+1/2,p}^{n,k} \right), \tag{5.23}$$

avec $v_{i+1/2,p}^{n,k,*}$ et $\tau_{i+1/2,p}^{n,k,*}$ sont la vitesse et le volume spécifique dans la solution du problème de Riemann à l'interface $i + 1/2$ (voir (2.60) page 25).

Alors les micros pas de temps sont données par

- le premier micro pas de temps

$$\Delta t_0^{n,K} = \mathrm{CFL}^{exp} \min_{\substack{(i+1/2,k)\in\mathcal{F}_0 \\ 0\le k\le K}} \frac{\min\left(\Delta x_i, \Delta x_{i+1}\right)}{c_{i+1/2,0}^{n,k}}, \tag{5.24}$$

- le p ième micro pas de temps

$$\Delta t_p^{n,K} = \min\left(\Delta t_{p-1}^{n,K}, \mathrm{CFL}^{exp} \min_{\substack{(i+1/2,k)\in\mathcal{F}_p \\ 0\le k\le K}} \frac{\min\left(\Delta x_i, \Delta x_{i+1}\right)}{c_{i+1/2,p}^{n,k}}\right), \tag{5.25}$$

- le pas de temps associé aux mailles de niveau k, $0 \le k \le K$,

$$\Delta t_p^{n,k} = \Delta t_{2p}^{n,k+1} + \Delta t_{2p+1}^{n,k+1}. \tag{5.26}$$

Les micros pas de temps décroissent au cours d'un macro pas de temps.

5.2 Résultats numériques

Dans cette partie nous allons montrer des résultats numériques obtenus avec la méthode de Pas de Temps Local (PTL) couplée avec la méthode de Multirésolution (MR). Tout d'abord, nous étudions un problème de tube à choc sur lequel nous comparons la précision et l'arbre de multirésolution entre les deux méthodes PTL et MR. Ensuite, nous envisageons un cas test d'écoulement diphasique où il y a des conditions aux limites à prendre en compte. Nous faisons une comparaison sur le pas de temps utilisé ainsi que le temps CPU entre les différentes méthodes d'adaptation. Finalement, nous illustrons la convergence numérique des deux méthodes PTL et MR.

Sur les figures, LTS est l'abréviation de Local Time Stepping (terme en anglais) dénotée par PTL dans le texte. Toutes les simulations numériques ont été faites avec un schéma Lagrange-Projection explicite sans reconstruction de pente. La condition CFL explicite est égale à 0.5.

5.2.1 Tube à choc

Tout d'abord nous testons la méthode PTL sur un problème de tube à choc. Il s'agit d'un problème de Riemann dans une conduite de 32 km de long, inclinée de 30 degrés vers le haut. Le mélange est initialement séparé en deux états au milieu de la conduite

$$\begin{pmatrix} \rho \\ Y \\ v \end{pmatrix}_L = \begin{pmatrix} 540 \\ 0.2 \\ -10 \end{pmatrix} \qquad \text{et} \qquad \begin{pmatrix} \rho \\ Y \\ v \end{pmatrix}_R = \begin{pmatrix} 400 \\ 0.4 \\ -10 \end{pmatrix}.$$

Pour ce cas-test, on utilise la loi de glissement Zuber-Findlay IFP. Le temps de simulation est de 42 secondes. Dans la Fig. 5.16, nous montrons les résultats numériques obtenus sur un maillage uniforme contenant 4096 mailles et sur un maillage adaptatif sur 5 niveaux de raffinement ayant 4096 mailles sur la grille la plus fine. La méthode adaptative utilise la stratégie PTL.

Sur la densité, la vitesse du mélange ainsi que la fraction massique et la pression, les résultats sont comparables tant sur les ondes lentes que sur les ondes rapides. Les croix correspondent aux niveaux

de raffinement du maillage adaptatif et représentent l'« arbre des niveaux de raffinement ». On constate qu'aux endroits où il y a une grande variation de la solution, on utilise des mailles de petites tailles. Par conséquent, la solution est calculée de manière précise puisqu'on la calcule localement sur une grille uniforme fine. En revanche, les niveaux grossiers sont utilisés là où la solution est régulière ce qui permet de réduire les calculs numériques.

Afin de comparer la méthode de Pas de Temps Local avec la méthode de Multirésolution, nous montrons dans la figure Fig. 5.17 les résultats numériques correspondants. On constate qu'ils sont très similaire. Dans cette figure, l'arbre des niveaux de la méthode MR est représenté par les □. Sur les images en bas de la Fig. 5.17 on montre l'agrandissement de l'onde lente au milieu (image gauche) et de l'onde rapide à droite (image droite). La qualité des résultats entre les deux méthodes est très comparable.

Fig. 5.9. Problème de tube à choc. Comparaison entre les simulations sur grille uniforme et sur grille adaptative avec stratégie PTL. Schéma explicite ordre 1, 4096 mailles, 5 niveaux de rafinement, $\varepsilon = 10^{-3}$.

Fig. 5.10. Comparaison des champs de densité calculés par les méthodes Multi-résolution et Pas de Temps Local avec en haut à gauche l'arbre MR et en haut à droite celui du PTL. Agrandissement sur l'onde lente (image en bas à gauche) et sur l'onde rapide vers la droite (image en bas à droite). Schéma explicite ordre 1, 4096 mailles, 5 niveaux de rafinement, $\varepsilon = 10^{-3}$.

Géométrie de la conduite :	- Longueur	:	4000 m, horizontal	
	- Diamètre	:	0.146 m	
	- Pas d'espace	:	15.625 m	
Discrétisation :	- CFL explicite	:	0.5	
Source et Ordre :	- Sans source, ordre 1 sans reconstruction de pente			
Modèles :	- Thermodynamique	:	liquide incompressible	
	- Hydrodynamique	:	glissement Zuber-Findlay IFP	
Conditions opératoires :	- Durée de la stabilisation			: 100 s
	- Durée de la rampe			: 250 s
	- Condition limite amont sur le débit de gaz			: 10 à 20 kg/m^2/s
	- Condition limite amont sur le débit de liquide			: 1000 kg/m^2/s
	- Condition limite aval sur la pression			: 10 bar

Fig. 5.11. Détails de l'expérience avec conditions aux limites pour la méthode PTL.

Fig. 5.12. Variation des débits à l'amont - Multirésolution (gauche) et Pas de Temps Local (droite). Schéma explicite ordre 1, 256 mailles, 5 niveaux de rafinement, $\varepsilon = 10^{-3}$

5.2.2 Avec conditions aux limites

Étudions maintenant un cas test avec conditions aux limites. Sur une conduite de longueur 4000 m, on injecte du gaz et du liquide à l'entrée. Leur débits sont maintenus constants pendant une période de stabilisation de 100 secondes. Ensuite on double le débit du gaz pendant 250 secondes (durée de la rampe) puis on le garde constant jusqu'à la fin de la simulation. La pression à la sortie de la conduite est maintenue constante à 10 bar. Les détails de ce cas test se trouvent à la Fig. 5.11

Les résultats numériques sont montrés dans la Fig. 5.12 où on compare les deux méthodes Multi-résolution et Pas de Temps Local. Tout d'abord on constate que les conditions aux limites sont bien respectées pour les débits du gaz et du liquide ainsi que pour la pression à l'aval. Les résultats sont comparables entre les deux méthodes, bien que ceux avec la méthode PTL soient un peu plus diffusifs. Afin de mieux juger la méthode PTL, nous devons faire une comparaison de performance en temps CPU.

Une des nouveautés que nous voulons montrer dans cette section est le traitement des pas de temps dans le cas de la méthode PTL. Pour le cas test avec conditions aux limites précédent, nous montrons dans la Fig. 5.13 l'évolution des pas de temps au cours du temps avec ou sans adaptation de maillage et en temps. Dans tous les cas, les pas de temps varient très peu pendant la période de stabilisation (100 secondes) puis décroissent pendant la rampe jusqu'à 350 secondes. Ils restent de nouveau « constants » puis croissent après 1100 secondes de simulation.

Fig. 5.13. Évolution du pas de temps au cours du temps avec différentes méthodes. L'agrandissement sur trois plages : 0 à 20 secondes, 180 à 200 secondes et 1500 à 1520 secondes.

Dans cette figure, nous faisons particulièrement attention à l'évolution du pas de temps de la méthode PTL. Pour cela, nous avons deux courbes à regarder : la première correspond à la méthode PTL utilisant un micro pas de temps fixe au cours d'un macro pas de temps, c'est la courbe représentée par la légende LTS-fix. La deuxième courbe, repérée par LTS-var, correspond à notre méthode PTL où le micro pas de temps varie au cours d'un macro pas de temps.

Dans la première stratégie, le micro pas de temps est calculé au début de chaque itération en temps, tout en respectant la condition CFL locale sur la solution courante. La courbe LTS-fix correspondante est composée de paliers dont l'origine est indiqué par un $\circ$. Ces derniers représentent en effet les macros pas de temps pendant lesquels le micro pas de temps reste constant. L'avantage de cette gestion de pas de temps est que l'on peut bénéficier directement de la propriété de conservativité des flux aux interfaces en utilisant des coefficients adéquats (voir [27, 34]), ce qui rend facile l'implémentation informatique. En revanche, dans notre cas où le pas de temps peut décroître, cette stratégie n'assure pas la stabilité du schéma à chaque pas de temps intermédiaire. Concrètement, le micro pas de temps calculé au premier pas de temps intermédiaire ne permet d'assurer que la stabilité présente. La solution évolue et il se peut que ce micro pas de temps viole la nouvelle condition de stabilité au temps intermédiaire suivant, comme dans la partie décroissante de la figure 5.13.

Pour régler ce problème, nous avons présenté au paragraphe 5.1.2 une nouvelle méthode où le micro pas de temps est recalculé à chaque pas de temps intermédiaire en fonction de la condition CFL locale au niveau courant. De plus, le micro pas de temps ne peut que décroître au cours d'un macro pas de temps pour assurer la stabilité au niveau supérieur. Les formules mathématiques correspondantes se trouvent dans (5.1.1.2), (5.12) et (5.13) et le résultat est représenté par la courbe LTS-var. Sur cette courbe, on voit bien qu'au cours d'un macro pas de temps (marqué par les $\times$), le micro pas de temps n'augmente pas. Lorsque le pas de temps doit décroître (de 180 à 200 secondes par exemple) la courbe LTS-var reste proche de la courbe des pas de temps de la méthode MR (stable), ce qui n'est pas le cas pour la courbe LTS-fix. On constate aussi que le micro pas de temps variable est toujours plus petit que le micro pas de temps fixe, ce qui explique la position en dessous de la courbe LTS-var par rapport à la courbe LTS-fix. Cette propriété explique aussi le retard de LTS-var vis à vis LTS-fix de 1500 à 1520 secondes à cause du temps cumulé dans la simulation. En ce qui concerne la courbe MR et U (pour le maillage uniforme), on voit qu'elles sont plus régulières comparées aux courbes LTS-fix et LTS-var.

***Remarque** 5.1.* Une autre stratégie utilisée dans la littérature [34, 75] consiste à utiliser un pas de temps constant pendant toute la simulation. En effet, ce pas de temps choisi égal au plus petit pas de temps utilisé pendant une simulation sur la grille uniforme fine. Ceci permet d'assurer la stabilité du schéma mais n'est évidemment pas très performant puisque le pas de temps n'est pas choisi de manière optimale.

5.3 Convergence et performance en temps CPU

Cette partie est consacrée à illustrer la convergence ainsi que la performance en temps CPU de la méthode de Pas de Temps Local. Tout d'abord, nous montrons la convergence de la méthode PTL par rapport au paramètre de seuillage ε. Pour cela, on considère le cas test du tube à choc précédent et on fait des simulations sur une grille uniforme (U), puis adaptative en espace (MR) et adaptative en espace et en temps (LTS). On utilise 4096 mailles pour la grille uniforme ainsi que pour la grille uniforme la plus fine lors de l'utilisation des méthodes adaptatives. Pour ces dernières, le nombre de niveaux de raffinement est égal à 5. Le paramètre de seuillage ε varie entre 10^{-1} à 10^{-14}. À partir de ces résultats numériques, on calcule les erreurs des solutions adaptatives induites par le seuillage, mesurées relativement à la solution uniforme. On trace les courbes des erreurs relatives de la densité et de la fraction massique du gaz en fonction de ε et on observe que ces erreurs tendent vers zéro.

Fig. 5.14. Convergence des erreurs relatives de la densité (gauche) et de la fraction massique du gaz (droite) en fontion du paramètre de seuillage ε.

Fig. 5.15. Erreurs relatives sur la densité en fonction des gains CPU pour la méthode adaptative en espace (MR) et adaptative en espace et en temps (LTS), pour différentes valeurs du paramètre de seuillage ε. Cas-test tube à choc.

Dans la Fig. 5.14, on constate que les deux méthodes Multirésolution et Pas de Temps Local sont convergentes, *i.e.* leur solutions convergent vers la solution uniforme calculée sur $\mathcal{S}^K$. On voit le même comportement de convergence numérique pour les deux méthodes MR et PTL, ce qui illustre bien le résultat théorique obtenu pour la méthode Multirésolution (voir (4.20)). Plus ε est petit, plus les erreurs sont faibles. Ceci est cohérent car pour une petite valeur de ε on seuille ou on néglige moins de détails et on utilise donc plus de petites mailles, la solution adaptative correspondante s'approche donc de la solution uniforme.

Sur ce même cas-test, nous voulons montrer la performance de la méthode PTL comparée à celle de la méthode MR. Cette performance est illustrée dans la Fig. 5.15 où on trace l'erreur relative de la densité en fonction du gain CPU entre les méthodes adaptative (adapt) et la méthode sans adaptation, *i.e.* sur une grille uniforme (unif). Cette performance dépend en effet de la valeur du paramètre de seuillage. Pour les deux méthodes adaptatives, plus ε est grand, plus le gain CPU est important. On constate que les gains CPU obtenus avec la méthode PTL sont nettement plus importants que ceux obtenus avec la méthode MR. Le facteur de gain entre les méthodes PTL et uniforme peut atteindre 40 ($\varepsilon = 10^{-1}$) pour une erreur relative acceptable, de l'ordre de 10^{-2}. Pour des valeurs pas trop petites de ε (supérieures à 10^{-5}), le facteur de gain entre la méthode PTL et la méthode MR varie entre 3 et 4

ce qui est très significatif. Ces gains étaient prévisibles du fait que la méthode PTL permet de réduire le nombre d'appels aux lois de fermeture et par conséquent de diminuer les calculs ce qui réduit le temps CPU.

Pour démontrer encore la performance de la méthode de Pas de Temps Local, on montre dans le tableau 5.1 les gains en temps CPU pour le cas-test avec des conditions aux limites présenté précédemment (voir les détails dans le Tab. 5.11, page 125 et les résultats numériques sur la page 127). Nous avons fait Les simulations avec deux lois hydrodynamiques différentes : une loi *simple* de type Zuber-Findlay et une loi *réaliste* (la loi synthétique présentée au chapitre 6)

Premier cas-test	Rapport UNI/MR				Rapport MR/PTL			
$N(K)$	Loi *simple*		Loi *réaliste*		Loi *simple*		Loi *réaliste*	
	rCPU	rNbLoi	rCPU	rNbLoi	rCPU	rNbLoi	rCPU	rNbLoi
256(5)	**2.98**	2.90	**3.17**	3.07	**6.61**	11.67	**10.41**	10.82
512(6)	**4.19**	4.64	**5.02**	4.85	**7.77**	18.79	**16.13**	17.06
1024(7)	**5.47**	6.99	**7.48**	7.13	**8.39**	29.20	**24.58**	25.88

Tableau 5.1 Rapport temps CPU et nombre d'appels aux lois d'état - Uniforme (U) versus Multirésolution (MR) et MR versus Pas de Temps Local (PTL). Schéma explicite ordre 1. Cas-test avec conditions aux limites.

Dans ce tableau, N est le nombre de mailles sur la grille uniforme la plus fine, et K est le nombre de niveau. rCPU désigne le rapport de temps CPU et rNbLoi désigne le rapport des nombres d'appels aux lois de fermeture. On remarque que le gain CPU croît lorsque l'on augmente le nombre de mailles sur la grille uniforme la plus fine tout en augmentant aussi le nombre de niveau de raffinement. Ceci est cohérent avec le fait que la méthode d'adaptation en espace est efficace pour un nombre de mailles et un nombre de niveaux suffisamment grands. De plus, on constate une meilleure performance de la méthode PTL par rapport à la méthode MR. On remarque aussi que, pour la loi simple de type Zuber-Findlay IFP, il n'y a pas toujours de corrélation entre le rapport de temps CPU et le rapport des appels aux lois de fermeture. Ceci est dû au fait que l'on perd du temps dans la gestion de la grille adaptative dans un ordre de grandeur comparable au temps passé dans le schéma numérique proprement dit. En revanche, pour une loi hydrodynamique plus *compliquée*, de type synthétique (voir chapitre 6), nous obtenons des gains plus importants qui sont bien corrélés avec le rapport des appels aux lois de fermeture. La raison est que dans ce cas, le temps passé dans le schéma volumes finis augmente beaucoup tandis que le coût spécifique à la gestion de la multirésolution reste identique.

5.4 Extension de la méthode de pas de temps local

Les résultats numériques que nous venons de voir sont obtenus avec le schéma Lagrange-Projection explicite d'ordre 1. Deux extensions du couplage PTL avec la Multirésolution sont actuellement en cours de validation. Il s'agit de l'application au schéma Lagrange-Projection semi-implicite et de la montée en ordre en espace. Nous présentons ici quelques premiers résultats préliminaires pour démontrer la faisabilité de ces extensions.

5.4.1 Pas de Temps Local pour le schéma semi-implicite d'ordre 1

L'avantage de travailler avec le schéma semi-implicite est que l'on peut utiliser un grand pas de temps lors de l'étape lagrangienne où les ondes acoustiques sont traitées de manière implicite. En conséquence, le temps de calcul est beaucoup moins important comparé à celui du schéma explicite. En revanche, la condition CFL beaucoup plus grande que 1 vis à vis des ondes acoustique nous oblige à travailler avec des zones de dépendance très larges lors de la construction du système linéaire à résoudre (voir section 3.5.4, chapitre 3). L'implémentation de la méthode PTL en semi-implicite nécessite donc des aménagements par rapport à la méthode proposée dans [75] et présentée dans ce chapitre. Les spécificités propres au cas semi-implicite sont présentées dans [34] où on peut également trouver plusieurs cas-tests numériques.

Dans la Fig. 5.16 et 5.17 on montre les résultats numériques du problème de tube à choc présenté précédemment. Ils sont obtenus avec le schéma Lagrange-Projection semi-implicite, en version uniforme (U) et multirésolution avec et sans stratégie pas de temps local (PTL et MR). Le nombre de mailles est égal à 4096 et on utilise 5 niveaux de grilles avec $\varepsilon = 10^{-3}$. Le macro pas de temps est pris constant $\Delta t = 0.065$. Ce pas de temps est en fait le plus petit pas de temps obtenu avec le maillage uniforme pour assurer la stabilité du schéma utilisé. Notons que pour ce schéma semi-implicite, nous n'avons pas encore implémenté la version micro pas de temps variable comme présenté dans la section 5.1.2. On constate que les résultats sont très similaires pour les deux méthodes MR et PTL (figure 5.17). Bien que la diffusion sur les ondes rapides soit plus importante pour la méthode PTL, on obtient la même précision sur l'onde lente pour les trois méthodes, uniforme, MR et LTS.

	Rapport UNI/MR				Rapport MR/PTL			
$N(K)$	Loi *simple*		Loi *réaliste*		Loi *simple*		Loi *réaliste*	
	rCPU	rNbLoi	rCPU	rNbLoi	rCPU	rNbLoi	rCPU	rNbLoi
4096(5)	**4.76**	5.41	**6.11**	5.30	**2.70**	3.47	**4.07**	3.59

Tableau 5.2 Rapport temps CPU et nombre d'appels aux lois d'état - Uniforme (U) versus Multirésolution (MR) et MR versus Pas de Temps Local (PTL). Schéma semi-implicite ordre 1. Problème du tube à choc.

Au niveau du temps de calcul, on obtient des gains en temps CPU significatifs pour la méthode MR et PTL. En utilisant une loi réaliste, la méthode adaptative avec stratégie pas de temps local est $6.11 \times 4.07 \backsimeq 24$ fois plus rapide que la méthode sans adaptation de maillage. De plus, ce gain est plus important que celui obtenu avec une loi simple (à l'ordre de $4.76 \times 2.7 \backsimeq 13$). On remarque que les gains avec la loi réaliste sont supérieurs au rapport des nombres d'appels aux lois de fermetures correspondant, tandis qu'avec la loi simple c'est le contraire. Ces résultats sont très encourageants puisque dans les applications réalistes on sera amené à utiliser des lois d'états très coûteuses.

Fig. 5.16. Problème de tube à choc pour la méthode de Pas de Temps Local. Schéma semi-implicite ordre 1, 4096 mailles, 5 niveaux de rafinement, $\varepsilon = 10^{-3}$.

Fig. 5.17. Comparaison des champs de densité calculés par les méthodes Multirésolution et Pas de Temps Local. Agrandissement sur l'onde lente au milieu (image en bas à gauche) et sur l'onde rapide vers la droite (image en bas à droite). Schéma semi-implicite ordre 1, 4096 mailles, 5 niveaux de rafinement, $\varepsilon = 10^{-3}$.

5.4.2 Pas de temps local pour le schéma explicite avec reconstruction de pente en espace

Dans le cadre de cette thèse, nous avons pu implémenter une première version de la méthode de pas de temps local pour un schéma explicite d'ordre supérieur à 1, *i.e.* avec reconstruction de pente en espace. Dans ce cas, nous devons augmenter le nombre de mailles dans les zones de transition à deux, contrairement à ce qu'on utilise pour l'instant (voir (5.20), page 120). Ceci a pour but de bien reconstruire les pentes dans la reconstruction linéaire (voir (4.27), page 100) afin d'obtenir des flux plus précis. Les résultats numériques correspondants sont présentés dans les Fig. 5.18 et 5.19. Les gains en temps CPU sont montrés dans le Tab. 5.3. Nous espérons améliorer cette version actuelle afin d'avoir de meilleurs gains.

Fig. 5.18. Problème de tube à choc pour la méthode de Pas de Temps Local. Schéma explicite ordre 2 en espace, 256 mailles, 5 niveaux de rafinement, $\varepsilon = 10^{-3}$.

	Rapport UNI/MR				Rapport MR/PTL			
$N(K)$	Loi *simple*		Loi *réaliste*		Loi *simple*		Loi *réaliste*	
	rCPU	rNbLoi	rCPU	rNbLoi	rCPU	rNbLoi	rCPU	rNbLoi
256(5)	**2.09**	2.24	**5.56**	2.22	**1.59**	1.53	**1.60**	1.50

Tableau 5.3 Rapport temps CPU et nombre d'appels aux lois d'état - Uniforme (U) versus Multirésolution (MR) et MR versus Pas de Temps Local (PTL). Schéma explicite avec reconstruction de pente.

Fig. 5.19. Comparaison des champs de densité calculés par les méthodes Multirésolution et Pas de Temps Local. Agrandissement sur l'onde lente au milieu (image en bas à gauche) et sur l'onde rapide vers la droite(image en bas à droite). Schéma explicite avec reconstruction de pente, 256 mailles, 5 niveaux de rafinement, $\varepsilon = 10^{-3}$.

5.5 Conclusion sur la méthode de pas de temps local

Nous venons de présenter le couplage de la stratégie de Pas de Temps Local avec la méthode Multirésolution. Dans la méthode PTL que nous présentons ici, nous voulons insister sur le fait qu'on peut travailler avec des micros pas de temps variables au cours d'un macro pas de temps. Le micro pas de temps décroît pendant un grand pas de temps tout en respectant la condition de stabilité du schéma numérique utilisé. À notre connaissance, cette gestion du pas de temps est une nouveauté.

Pour une qualité de résultats comparable avec celle obtenue avec la méthode MR ou sans adaptation de maillage, la méthode PTL nous donne des gains en temps CPU importants comme ce qu'on a prévu. Ces gains sont notamment intéressants lorsque l'on travaille avec des cas-tests réalistes et des lois hydrodynamiques très complexes. De plus dans ce cas, ils sont bien corrélés avec le nombre d'appels aux lois de fermeture car le coût du calcul numérique augmente beaucoup tandis que le coût spécifique à la gestion de la multirésolution, *i.e* l'opération de codage-seuillage-décodage (voir section 4.1.1.2 et 4.1.2 chapitre 4), reste identique. En conséquence, le temps CPU consacré à la gestion de la grille adaptative est négligeable devant le temps CPU des calculs numériques.

Tout comme la méthode Multirésolution, les gains CPU ainsi que la qualité des résultats obtenus avec la méthode PTL dépendent du paramètre de seuillage ε ainsi que du nombre de niveaux de raffinement K. Dans la configuration classique, on prend souvent $\varepsilon = 10^{-3}$ et $K = 5$ pour obtenir des gains intéressants. Nous avons illustré numériquement la convergence des méthodes PTL et MR en fonction du paramètre de seuillage, montrant aussi la robustesse de ces stratégies adaptatives.

Dans ce travail, nous avons pu aussi implémenter la méthode PTL pour le schéma Lagrange-Projection en semi-implicite ordre 1 et en explicite avec reconstruction de pente en espace. Les résultats préliminaires obtenues avec ces schémas sont prometteurs. Cependant, leur implémentation est à améliorer afin d'avoir de meilleures performances en temps de calcul.

Chapitre 6

Loi hydrodynamique Synthétique

Dans ce chapitre, nous examinons de plus près la loi de glissement *réaliste* déjà mentionnée dans les chapitres précédents pour les simulations. En effet, on constate que les lois de glissement *simples*, de type Zuber-Findlay CEA (voir section 1.4.2.2) ou Zuber-Findlay IFP (voir section 1.4.2.3) ne peuvent être utilisées que si l'on connaît a priori le régime de l'écoulement. Par exemple, une simulation d'un écoulement où apparaît l'état monophasique gaz ne peut pas utiliser la loi Zuber-Findlay CEA. Cela nous conduit donc à travailler avec une loi de glissement plus générale, appelée loi Synthétique, qui couvre l'ensemble des régimes d'écoulements. Cette loi permet de reconnaître non seulement les trois configurations d'écoulements de bases (dispersé, intermittent et séparé), mais aussi les régimes de « transition ».

6.1 Principes de la loi de glissement Synthétique

6.1.1 Un nouveau jeu de variables

Tout d'abord, rappelons quelques grandeurs :

- R_g (resp. R_l) la fraction surfacique du gaz (resp. du liquide),
- $u_g = R_g v_g$ (resp. $u_l = R_l v_l$) la vitesse superficielle du gaz (resp. du liquide),
- $u_s = u_g + u_l$ la vitesse superficielle du mélange,
- $\mathcal{T}$ l'inclinaison de la conduite,
- $\phi = v_l - v_g$ le glissement ou l'écart de vitesse entre deux phases.

Les autres grandeurs sont : ρ la densité du mélange, Y la fraction massique du gaz, P la pression du mélange.

En modélisation hydrodynamique, il est plus naturel de travailler avec le jeu de variables (R_g, P, u_s) plutôt que celui du cadre mathématique introduit dans le chapitre 1, à savoir (ρ, Y, v). Aussi, au lieu de formuler la fermeture en tant que « glissement » $v_l - v_g = \phi(\rho, Y, v)$, il est d'usage de l'exprimer sous la forme d'une fonction calculant u_g, c'est-à-dire

$$u_g = \Psi(R_g, P, \mathcal{T}, u_s). \tag{6.1}$$

Viviand [95, 96] a d'ailleurs montré que cette façon de voir la fermeture hydrodynamique est particulièrement adaptée au modèle dit NPW (No Pressure Wave) [79, 93] qui constitue une approximation hydrostatique de notre modèle DFM.

La fonction Ψ doit vérifier les axiomes mathématiques et physiques suivants :

- Régularité : Ψ doit être au moins C^0 par rapport à R_g et u_s.
- Cohérence avec les états monophasiques gaz, c'est-à-dire :

$$\Psi(R_g = 0, P, \mathcal{T}, u_s) = 0 \quad \text{et} \quad \Psi(R_g = 1, P, \mathcal{T}, u_s) = u_s. \tag{6.2}$$

  En d'autres termes, v_g reste borné quand $R_g \to 0$, et v_l reste borné quand $R_l \to 0$.
- Symétrie d'invariance par rapport à l'orientation de la conduite, c'est-à-dire :

$$\Psi(R_g, P, -\mathcal{T}, -u_s) = -\Psi(R_g, P, \mathcal{T}, u_s). \tag{6.3}$$

  Autrement dit, si l'on « passe de l'autre côté » de la conduite et on observe l'écoulement, on doit constater la même relation algébrique.

6.1.2 Une nouvelle fonction Ψ

La fonction Ψ préconisée par la loi Synthétique est une combinaison de trois lois de fermeture hydrodynamique de base. Sur l'intervalle $R_g \in [0, 1]$, nous distinguons donc trois zones, chacune correspond à une loi spécifique, à savoir :

- Écoulement dispersé : $R_g \in [0, R_g^\flat]$ avec (voir 1.4.2.1)

$$u_g^d = R_g(u_s + \delta), \tag{6.4}$$

  où $\delta = 1.53 \sqrt[4]{g\varrho/\rho_l} \sin\mathcal{T}$ est la vitesse de dérive.
- Écoulement intermittent : $R_g \in [R_g^\flat, R_g^\sharp]$ avec

$$u_g^i = u_g^\flat + v_g^t(R_g - R_g^\flat) \tag{6.5}$$

  où $v_g^t = c_0 u_s + c_1$ est la vitesse de montée d'une bulle de gaz et $c_0 = 1.05 + 0.15\sin^2\mathcal{T}$ et $c_1 = 0.35\sqrt{gD}\sin\mathcal{T}$. On remarque que la constante c_0 est différente de celle utilisée dans la loi Zuber-Findlay IFP (voir 1.4.2.3), la différence étant $0.05 + 0.05\sin^2\mathcal{T}$.
- Écoulement séparé : sur $R_g \in [R_g^\sharp, 1]$, u_g^s est solution d'un bilan hydrostatique

$$\beta_g(R_g)G(u_g) + \beta_l(R_g)G(u_g - u_s) + \beta_i(R_g)G(u_g - R_g u_s) = \gamma(R_g, \mathcal{T}), \tag{6.6}$$

  où $G(x) = \frac{1}{2}x|x|$ et $\beta_g, \beta_l, \beta_i$ sont fonctions de R_g représentant des coefficients de frottement, et γ représente la différence des densités de chaque phase. Ces fonctions sont choisies de telle manière que ce régime soit valide au voisinage de $R_g = 1$. Notons que le « séparé » est déjà une synthèse entre deux lois : stratifiée et annulaire.

Chacune des briques élémentaires (6.4), (6.5) et (6.6) vérifie les axiomes (6.2) et (6.3). À u_s et $\mathcal{T}$ fixés, la relation (6.4) est représentée par une droite passant par 0 de pente u_s et (6.5) est une droite de pente v_g^t. En revanche, l'allure de la courbe (6.6) n'est pas connue.

Comme illustre la Fig. 6.1, $R_g^\flat$ est le seuil de R_g qui définit le passage du régime dispersé vers le régime intermittent. Il est donc l'intersection entre les courbes (6.4) et (6.5), tandis que $R_g^\sharp$, intersection entre (6.5) et (6.6), est le seuil de changement entre le régime intermittent et séparé. Notons que $R_g^\flat$ et $R_g^\sharp$ dépendent tous les deux de $(P, \mathcal{T}, u_s)$ mais pas de R_g. En pratique, à u_s et $\mathcal{T}$ fixés, $R_g^\flat$ est une constante donnée par une formule empirique. Le calcul de $R_g^\sharp$ est plus difficile, et dépend de la position du point $u_g^i(R_g = 1)$ par rapport à $u_g^s(R_g = 1)$. En général, les courbes (6.5) et (6.6) se coupent, mais au cas où elles ne se coupent pas, on décide de remplacer la droite (6.5) par la tangente $\tilde{u}_g^i$ à (6.5) issue du point $(R_g^\flat, u_g^\flat)$, et $R_g^\sharp$ est alors le point de contact (voir Fig. 6.2). On peut démontrer que ce processus ne fait pas apparaître de discontinuité, d'où l'intérêt de la loi Synthétique.

Fig. 6.1. Loi Synthétique - Vitesse superficielle du gaz u_g en fonction de R_g - Configuration 1 : les deux coubes u_g^s et u_g^i se coupent au point $(R_g^\sharp, u_g^\sharp)$.

Fig. 6.2. Loi Synthétique - Vitesse superficielle du gaz u_g en fonction de R_g - Configuration 2 : les deux coubes u_g^s et u_g^i ne se coupent pas. La droite u_g^i est remplacé par la tangente $\tilde{u}_g^i$ à u_g^s.

***Remarque** 6.1.* Considérons maintenant le régime intermittent défini par (6.5) sur une conduite horizontale ($\mathcal{T} = 0$). Dans ce cas, $v_g^t = c_0 u_s$ et $u_g^\flat = R_g^\flat u_s$. Après quelques calculs, on obtient les vitesses absolues du gaz et du liquide pour ce régime d'écoulement

$$v_g^i = u_s \frac{c_0 R_g - (c_0 - 1) R_g^\flat}{R_g} \quad \text{et} \quad v_l^i = u_s \frac{1 - (c_0 R_g - (c_0 - 1) R_g^\flat)}{1 - R_g}. \tag{6.7}$$

Dans ce cas, le glissement ϕ est donné par

$$\phi = u_s \frac{(c_0 - 1)(R_g - R_g^\flat)}{R_g (1 - R_g)} \tag{6.8}$$

Pour une constante c_0 différent de 1, nous avons par conséquent un glissement non nul. Sur l'intervalle $[R_g^\flat, R_g^\sharp]$ il s'agit en fait de la loi Zuber-Findlay IFP à une constante près ($\zeta = 0.05 + 0.05 \sin^2 \mathcal{T}$), et dans ce cas ϕ dépend de R_g. Par conséquent, les résultats numériques d'une même expérience obtenus par ces deux lois sont différents.

Fig. 6.3. Fonction à annuler $F(\phi)$ - Un cas particulier.

6.1.3 Méthode de résolution

À partir de (6.1), on peut revenir à la formulation « glissement » par la relation

$$R_g(v - X\phi) - \Psi\left(R_g, P, \mathcal{T}, v - (XR_g - YR_l)\phi\right) = 0, \tag{6.9}$$

dont la nature implicite $F(\phi) = 0$ nous oblige à mettre en œvre un processus itératif. Plus la loi hydrodynamique Ψ est complexe, plus la résolution de (6.9) s'avère longue et difficile.

Discutons maintenant comment trouver le *zéro* de (6.9). Supposons que la fonction $F(\phi)$ est suffisamment régulière et monotone, une méthode itérative de type Newton est alors suffiante pour résoudre (6.9). La fonction $\Psi(R_g, P, \mathcal{T}, u_s)$ étant définie de manière complexe. L'allure de la fonction $F(\phi)$ peut parfois être pathologique comme en témoign la Fig. 6.3. Par conséquent la méthode de Newton est incapable de trouver ce *zéro*. Dans ce cas, on utilise la méthode de Dichotomie comme alternative, tout en supposant que la fonction $F(\phi)$ admet une solution unique.

***Remarque** 6.2.* Pour que la méthode Newton marche, nous devons assurer que la dérivée $F'(\phi)$ soit non nulle. Il faut donc avoir une condition supplémentaire sur Ψ :

$$F'(\phi) = XR_g + (XR_g - YR_l)\Psi_{u_s} \neq 0, \iff \Psi_{u_s} \neq \frac{\rho_l}{1 - \frac{\rho_g}{\rho_l}} \tag{6.10}$$

De plus, $F'(\phi) = 0$ lorsque $R_g = 0$ ou $R_l = 0$, nous devons avoir dans ce cas un traitement spécial.

6.2 Résultats numériques avec la loi Synthétique

Passons en revue maintenant quelques résultats numériques obtenus avec la loi Synthétique. Nous utilisons le schéma Lagrange-Projection semi-implicite avec une recontruction de pente en espace et un coefficient d'implicitation θ égal à 0.7. Les cas-tests sont ceux avec conditions aux limites que l'on a déjà présentés dans les chapitres précédents.

6.2.1 Conduite inclinée de 30 degrés

Le premier cas-test consiste à simuler l'expérience 3.4.1.2 présentée dans le chapitre 3. La configuration d'écoulement est la même mais cette fois-ci la conduite est inclinée de 30 degrés afin de bien prendre en compte le glissement. On utilise un maillage uniforme où la taille de maille est constante $\Delta x = 39$ m (256 mailles). Pour ce cas-test nous allons comparer les résultats obtenus avec la loi simple Zuber-Fidlay et ceux obtenus avec la loi réaliste Synthétique. Sur la Fig. 6.4, on montre l'évolution au

cours du temps de la fraction massique du gaz, du débit du gaz et du liquide, et de la pression. Les courbes correspondent à cette évolution à 0 m (entrée de la conduite), 1000 m, 2000 m, 3000 m et à 4000 m (sortie de la conduite).

On constate tout d'abord une différence sur la fraction massique du gaz pour les deux résultats. Ceci est dû au fait que les deux lois de glissement sont différentes et leurs paramètres ne sont pas les mêmes. Dans les deux cas, on observe un phénomène de transport se propageant au cours du temps, de l'entrée de la conduite à 1000 m et jusqu'à la sortie. Les conditions aux limites sont bien respectées, à savoir le changement de $0.2\times$Area à 0.4 kg/m^3/s pour le débit du gaz à 1000 seconde et le débit du liquide est maintenu constant à 20 kg/m^3/s. Ici Area est l'aire de la section de la conduite. Pour les deux lois, ces débits sont comparables, bien qu'il y ait une légère différence sur leur amplitude. Sur la pression, on obtient de résultats très similaires.

Nous ne montrons pas les résultats obtenus avec la méthode Multirésolution mais nous avons obtenus des gains en temps de calcul significatif en utilisant cette méthode pour une même qualité de résultats. Ces gains sont montrés dans le Tab. 6.1 où N est le nombre de maille sur la grille uniforme la plus fine et K est le nombre de niveaux de raffinement. On constate que le gain croît quand le nombre de maille augmente et il est meilleur avec la loi réaliste qu'avec la loi simple.

	Rapport UNI/MR			
$N(K)$	Loi *simple*		Loi *réaliste*	
	rCPU	rNbLoi	rCPU	rNbLoi
128(5)	**1.93**	2.01	**2.11**	2.08
256(5)	**3.34**	3.20	**3.37**	3.31
512(5)	**4.90**	5.01	**5.24**	5.16

Tableau 6.1 Rapport temps CPU et nombre d'appels aux lois d'état - Uniforme (UNI) versus Multirésolution (MR) avec 5 niveaux de raffinement et $\varepsilon = 10^{-3}$. Schéma Lagrange-Projection semi-implicite avec reconstruction de pente.

Fig. 6.4. Simulation à 8000 secondes sur une conduite inclinée de 30 degrés. Loi de glissement Zuber-Findlay IFP versus loi de glissement Synthétique. Maillage uniforme avec 256 mailles.

Fig. 6.5. Simulation à 2000 secondes sur une V-conduite. Loi de glissement Zuber-Findlay IFP. 256 mailles, 5 niveaux de raffinement, $\varepsilon = 10^{-3}$.

6.2.2 Conduite en V

Pour le deuxième cas-test, nous reprenons celui déjà présenté dans la section 4.3.2 chapitre 4. Rappelons qu'il s'agit d'une conduite dont la première moitié est penchée de 30 degrés vers le bas et l'autre moitié est penchée de 30 degrés vers le haut. Les détails de la simulation se trouve dans la Fig. 4.5 où on fait varier le débit du gaz pendant 30 secondes à partir de 100 secondes de simulation. Le débit du liquide à l'entrée et la pression du mélange à la sortie de la conduite sont maintenus constants.

Nous montrons les résultats numériques après 2000 secondes de simulation en utilisant la méthode de Multirésolution, *i.e.* avec un maillage adaptative. Le nombre de niveau de raffinement est 5 et le paramètre de seuillage est égal à 10^{-3}. Afin de voir la différence entre deux types de loi de glissement simple et réaliste, nous montrons sur la Fig. 6.5 ceux obtenus avec la loi Zuber-Findlay IFP et sur la Fig. 6.6 ceux obtenus avec la loi Synthétique. Sur ces figures, on regarde la densité, la vitesse, la pression et le débit total du mélange le long de la conduite. On constate qu'il y a une différence sur l'amplitude de la vitesse et du débit total. On observe aussi un décalage sur la densité entre deux lois. Dans les deux cas, l'allure des courbes sont similaires.

Avec la méthode de Multirésolution, nous avons un maillage adaptatif où on utilise des mailles de tailles différentes. Ces dernières sont fines (de niveau 5) au milieu et à la sortie de la conduite car il y a une forte variation de la solution. Dans les zones où la solution est plus régulière, on utilise

Fig. 6.6. Simulation à 2000 secondes sur une V-conduite. Loi de glissement Synthétique. 256 mailles, 5 niveaux de raffinement, $\varepsilon = 10^{-3}$.

	Rapport UNI/MR			
$N(K)$	Loi *simple*		Loi *réaliste*	
	rCPU	rNbLoi	rCPU	rNbLoi
256(5)	**1.69**	1.62	**1.71**	1.53
512(5)	**2.82**	2.84	**2.97**	2.61

Tableau 6.2 Rapport temps CPU et nombre d'appels aux lois d'état - Uniforme (UNI) versus Multirésolution (MR) avec 5 niveaux de raffinement et $\varepsilon = 10^{-3}$. Schéma Lagrange-Projection semi-implicite avec reconstruction de pente.

des mailles plus grossières (du niveau 2, 3 et 4). Par conséquent, nous obtenons des gains significatifs au niveau de temps de calcul entre une méthode sans adaptation de maillage (uniforme) et avec adaptation de maillage (Multirésolution). On montre ces gains dans le Tab. 6.2. On constate que le gain CPU est variable suivant le nombre de mailles sur la grille uniforme la plus fine N : plus N est grand, plus le gain en temps CPU est important. De plus, ce gain est meilleur avec la loi Synthétique. On peut avoir un gain très significatif, de l'ordre de 3, entre le schéma uniforme et le schéma adaptatif.

6.3 Conclusion sur la loi Synthétique

Dans ce chapitre, nous avons présenté la loi Synthétique, appréciée pour son « réalisme » par les équipes IFP. Nous avons montré quelques résultats numériques en application de cette loi, bien plus chère que les lois « académiques » simples. À cause du surcoût d'un appel à la loi hydrodynamique, le gain en temps CPU entre le schéma uniforme et adaptatif est meilleur avec la loi Synthétique qu'avec la loi simple. Ces résultats sont très encourageants en vue du développement de la méthode de Pas de Temps Local appliquée au schéma Lagrange-Projection semi-implicite d'ordre supérieur à 1 où nous aurons certainement des gains encore plus significatifs.

Conclusion et perspectives

Un schéma relaxation-Lagrange-Projection consistant et conservatif

Nous présentons dans cette recherche de doctorat une nouvelle méthode numérique pour l'approximation des solutions de systèmes d'EDP non-linéaires pour les simulations des écoulements diphasiques en conduites pétrolières. Le Drift-Flux modèle utilisé est fermé par deux lois de fermeture thermodynamique (loi de pression) et hydrodynamique (glissement entre les phases) qui sont fortement non-linéaires. Afin de traiter de manière efficace ces non-linéarités, nous utilisons la méthode de relaxation qui rend le système plus facile à résoudre avec plusieurs propriétés intéressantes.

Contrairement au schéma de relaxation en eulérien semi-implicite qui ne permet pas de garantir la positivité de certaines variables physiques, nous proposons dans ce travail un schéma de relaxation en Lagrange-Projection positif en explicite comme en semi-implicite. Basé sur la décomposition ALE, le schéma Lagrange-Projection semi-implicite possède de nombreux avantages :

1. Par construction, ce schéma est localement conservatif.

2. Il est consistant grâce au critère de stabilité du schéma.

3. Le système linéaire à résoudre à chaque itération est tridiagonal par blocs, la taille des blocs étant 2×2 au lieu de 5×5. Ainsi, on gagne en temps CPU.

4. On est capable d'assurer la stabilité, la positivité des densités et le principe du maximum sur les fractions massiques sous une condition CFL permettant de calculer le pas de temps.

Une nouveauté de cette recherche sur laquelle nous voulons insister est l'estimation a priori du pas de temps dans le cas semi-implicite. Plus précisément, le pas de temps est calculé explicitement tout en prenant en compte les informations à l'intérieur comme aux bords du domaine, ce qui permet de garantir la positivité du schéma. À notre connaissance, ce type d'estimation n'existe pas dans la littérature.

Au cours de ce travail, nous avons dû faire face à un certain nombre d'aspects techniques délicats supplémentaires, comme la prise en compte des termes sources et la gestion des conditions aux limites. Grâce à l'expérience acquise lors des travaux précédents, ces aspects ont pu être traités de façon satisfaisante. En effet, le schéma Lagrange-Projection semi-implicite permet de simuler beaucoup de cas-tests « réels » dont les résultats sont comparables avec ceux obtenues avec le schéma eulérien. Au niveau de la précision, les résultats numériques ont montré un accord parfait entre les deux schémas semi-implicites (eulérien et Lagrange-Projection) tant sur les amplitudes que sur les positions des ondes. Nous obtenons une même précision pour un temps de calcul moins important avec le schéma L-P. Ce dernier a été démontré dans la troisième chapitre : on gagne un facteur 1.4 en temps CPU en semi-implicite.

Des méthodes adaptatives en temps et en espace efficaces

La mise en œuvre du schéma Lagrange-Projection, en tant que nouvelle technique de semi-implicitation sur une grille uniforme, est un point de passage important de la thèse. Suite aux résultats favorables de ce schéma, nous avons intégré la méthode Multirésolution afin de travailler avec une grille adaptative. Étant une sur-couche indépendante du schéma numérique employé, la méthode MR permet d'accélérer les simulations grâce à la réduction du nombre d'appels aux lois de fermeture très coûteuses. De plus, le schéma Lagrange-Projection utilisant la méthode MR est capable de traiter des problèmes complexes, à savoir celui du severe slugging.

Nous avons ensuite mis au point la méthode de Pas de Temps Local couplé avec la méthode Multirésolution. Cette méthode PTL consiste à utiliser les pas de temps variables en fonction du pas d'espace du maillage adaptatif. L'originalité de la méthode PTL que nous proposons ici est que les micros pas de temps sont variables au cours d'un grand pas de temps, contrairement à l'utilisation des méthodes PTL classique où ils sont constants. Cette nouvelle stratégie permet de garantir la stabilité du schéma au cours du temps. Nous avons implémenté la méthode PTL pour le schéma L-P explicite d'ordre 1 en temps et en espace où nous obtenons des résultats très prometteurs, notamment sur le modèle avec la loi de glissement réaliste Synthétique. Pour ce dernier, nous avons des gains CPU très intéressants en faveur de la méthode PTL par rapport à la méthode MR. De plus, ces gains sont corrélés avec le rapport du nombre d'appels aux lois de fermeture, ce qui n'est pas toujours le cas pour une loi simple. La convergence numérique des méthodes PTL et MR ont étés aussi montrés.

Perspectives

En ce qui concerne le schéma numérique, bien que le traitement des conditions aux limites (CL) actuel soit satisfaisant il existe des possibilités d'amélioration de cet aspect, notamment le passage à l'ordre 2 aux limites du domaine. Pour les méthodes adaptatives, il est important de pouvoir travailler avec un schéma semi-implicite d'ordre 2 en temps et en espace. Pour cela, nous avons implémenté une première version de la méthode de Pas de Temps Local pour le schéma Lagrange-Projection explicite d'ordre 2 en espace et celui en semi-implicite d'ordre 1. Les résultats sont prometteurs mais nous espérons avoir des gains encore meilleurs en améliorant certains aspects techniques et informatiques. Nous résumons dans le Tab 6.3 le travail de cette thèse : ce qui ont été faits (désignés par un $\surd$) et ce qui reste à améliorer/faire.

	Positivité $\rho > 0, Y \in [0,1]$	Glissement non nul		Ordre 1	Ordre 2
L-P explicite	$\surd$	$\surd$	MR	$\surd$	$\surd$
			PTL	$\surd$	à améliorer
			CL	$\surd$	à faire
L-P semi-implicite	$\surd$	$\surd$	MR	$\surd$	$\surd$
			PTL	à améliorer	à faire
			CL	$\surd$	à faire

Tableau 6.3 Récapitulatif sur le schéma Lagrange-Projection en utilisation avec les méthodes adaptatives Multirésolution (MR) et Pas de Temps Local (PTL).

Bibliographie

[1] R. Abgrall and A. Harten, *Multiresolution representation in unstructured meshes*, SIAM J. Numer. Anal., 35 (1998), pp. 2128–2146.

[2] N. Andrianov, F. Coquel, M. Postel, and Q. H. Tran, *A linearly semi-implicit AMR method for 1-d gas-liquid flows*, in Proceedings of the 3rd International Conference on Computational Methods in Multiphase Flow, Portland, Maine, October 2005, A. A. Mammoli and C. A. Brebbia, eds., no. III in Computational Methods in Multiphase Flows, Berlin, 2005, WIT Press, pp. 295–304.

[3] N. Andrianov, F. Coquel, M. Postel, and Q. H. Tran, *Semi-implicit multiresolution for multiphase flows*, in Proceedings of the 6th European Conference on Numerical Mathemat ics and Advanced Applications, Santiago de Compostela, July 2005, A. Bermúdez de Castro, D. Gómez, and P. Quintela, P. an d Salgado, eds., vol. 6 of Numerical Mathematics and Advanced Applications, Berlin, 2006, Springer, pp. 814–821.

[4] N. Andrianov, F. Coquel, M. Postel, and Q. H. Tran, *A relaxation multiresolution scheme for accelerating realistic two-phase flows calculations in pipelines*, Int. J. Numer. Math. Fluids, 54 (2007), pp. 207–236.

[5] M. Baudin, *Schémas de relaxation explicites et semi-implicites pour la simulation des écoulements diphasiques dans TACITE*, Rapport IFP 56762, Institut Français du Pétrole, 2002.

[6] ——, *Méthodes de relaxation pour la simulation des écoulements polyphasiques dans les conduites pétrolières*, Thèse de doctorat, Université Pierre et Marie Curie (Paris 6), 2003.

[7] M. Baudin, C. Berthon, F. Coquel, R. Masson, and Q. H. Tran, *A relaxation method for two-phase flow models with hydrodynamic closure law*, Numer. Math., 99 (2005), pp. 411–440.

[8] M. Baudin, F. Coquel, and Q. H. Tran, *A semi-implicit relaxation scheme for modeling two-phase flow in a pipeline*, SIAM J. Sci. Comput., 27 (2005), pp. 914–936.

[9] M. Baudin, F. Daude, and Q. H. Tran, *Schémas de relaxation explicite pour l'équation d'énergie dans TACITE*, Rapport IFP 56989, Institut Français du Pétrole, 2002.

[10] M. Baudin, A. Fornel, and Q. H. Tran, *Quelques résultats avec la méthode de relaxation pour les écoulements diphasiques dans les conduites*, Rapport IFP 57777, Institut Français du Pétrole, 2003.

[11] M. Baudin, V. Martin, and Q. H. Tran, *Méthode de relaxation pour la simulation des écoulements diphasiques dans TACITE*, Rapport IFP 56023, Institut Français du Pétrole, 2001.

[12] A. Benabdallahh and D. Serre, *Problèmes aux limites pour des systèmes hyperboliques non linéaires de deux équations à une dimension d'espace*, C. R. Acad. Sci. Paris Sér I, 305 (1987), pp. 677–680.

[13] M. J. Berger and P. Collela, *Local adaptive mesh refinement for shock hydrodynamic*, J. Comput. Phys., 82 (1989), pp. 64–84.

[14] M. J. Berger and J. Oliger, *Adaptive mesh refinement for hyperbolic partial differential equations*, J. Comput. Phys., 53 (1984), pp. 484–512.

[15] B. L. Bihari and A. Harten, *Multiresolution schemes for the numerical solution of 2-D conservation laws. I*, SIAM J. Sci. Comput., 18 (1997), pp. 315–354.

[16] F. Bouchut, *Nonlinear stability of finite volume methods for hyperbolic conservation laws, and well-balanced schemes for sources*, Frontiers in Mathematics, Birkhäuser, 2004.

[17] F. Bramkamp, P. Lamby, and S. Müller, *An adaptive multiscale finite volume solver for unsteady and steady state flow computations*, J. Comput. Phys., 197 (2004), pp. 460–490.

[18] R. Bürger, R. Ruiz, K. Schneider, and M. A. Sepúlveda, *Fully adaptive multiresolution schemes for strongly degenerate parabolic equations with discontinuous flux*, J. Engrg. Math., 60 (2008), pp. 365–385.

[19] C. Chalons, *Bilans d'entropie discrets dans l'approximation numérique des chocs non-classiques. Application aux équations de Navier-Stokes multi-pression 2-D et à quelques systèmes visco-capillaires*, Thèse de doctorat, Université Pierre et Marie Curie, 2002.

[20] G. Chen and T. P. Liu, *Zero relaxation and dissipation limits for hyperbolic conservation laws*, Comm. Pure Appl. Math, 46 (1993), pp. 755–781.

[21] G. Q. Chen, C. D. Levermore, and T. P. Liu, *Hyperbolic conservation laws with stiff relaxation terms and entropy*, Comm. Pure Appl. Math., 47 (1994), pp. 787–830.

[22] G. Chiavassa and R. Donat, *Numerical experiments with multilevel schemes for conservation laws*, in Godunov methods (Oxford, 1999), Kluwer/Plenum, New York, 2001, pp. 155–160.

[23] G. Chiavassa and R. Donat, *Point value multiscale algorithms for 2D compressible flows*, SIAM J. Sci. Comput., 23 (2001), pp. 805–823.

[24] G. Chiavassa, R. Donat, and A. Marquina, *Fine-mesh numerical simulations for 2D Riemann problems with a multilevel scheme*, in Hyperbolic problems: theory, numerics, applications, Vol. I, II (Magdeburg, 2000), vol. 141 of Internat. Ser. Numer. Math., 140, Birkhäuser, Basel, 2001, pp. 247–256.

[25] A. Cohen, *Numerical Analysis of Wavelet Methods*, vol. 32 of Studies in Mathematics and its Applications, North-Holland, Amsterdam, 2003.

[26] A. Cohen, N. Dyn, S. M. Kaber, and M. Postel, *Multiresolution schemes on triangles for scalar conservation laws*, J. Comput. Phys., 161 (2000), pp. 264–286.

[27] A. Cohen, S. M. Kaber, S. Müller, and M. Postel, *Fully adaptive multiresolution finite volume schemes for conservation laws*, Math. Comp., 72 (2003), pp. 183–225.

[28] A. Cohen, S. M. Kaber, and M. Postel, *Multiresolution analysis on triangles: application to gas dynamics*, in Hyperbolic problems: theory, numerics, applications, Vol. I, II (Magdeburg, 2000), H. Freistühler and G. Warnecke, eds., vol. 140/141 of Internat. Ser. Numer. Math., Basel, 2001, Birkhäuser, pp. 257–266.

[29] ——, *Adaptive multiresolution for finite volume solutions of gas dynamics*, Computer and Fluids, 32 (2003), pp. 31–38.

[30] F. Collino, T. Fouquet, and P. Joly, *Conservative space-time mesh refinement methods for the FDTD solution of Maxwell's equations*, J. Comput. Phys., 211 (2006), pp. 9–35.

[31] F. Coquel, Q. L. Nguyen, M. Postel, and Q. H. Tran, *Lagrange-projection scheme for two-phase flows : 1. Application to realistic test cases*, in Proceedings of the 6th International Congress on Industrial and Applied Mathematics, Zürich, 2007.

[32] ——, *Lagrange-projection scheme for two-phase flows : 2. Boundary conditions and second-order enhancement*, in Proceedings of the 6th International Congress on Industrial and Applied Mathematics, Zürich, 2007.

[33] ——, *Local time stepping applied to implicit-explicit methods for hyperbolic systems.* Soumis 2007, prépublication LJLL R07058, 2007.

[34] ——, *Local time stepping for implicit-explicit methods on time varying grids*, in Proceedings of the 7th European Conference on Numerical Mathematics and Advanced Applications (ENUMATH), Graz, K. Kunisch, G. Of, and O. Steinbach, eds., Berlin, 2007, Springer-Verlag, pp. 257–264.

[35] ——, *Performances de la multirésolution avec ou sans pas de temps local sur des cas tests réalistes d'écoulements diphasiques*, in Proceedings du 39ème Congrès National d'Analyse Numérique (CANUM), Saint Jean de Monts, 2008.

[36] ——, *Time-varying grids for gas dynamics*, in Proceedings of the 14th European Conference on Mathematics for Industry, Madrid, July 2006, L. L. Bonilla, M. Moscoso, G. Platero, and J. M. Vega, eds., vol. 12 of Mathematics in Industry, Berlin, 2008, Springer-Verlag, pp. 887–891.

[37] F. Coquel and B. Perthame, *Relaxation of energy and approximate Riemann solvers for general pressure laws in fluid dynamics*, SIAM J. Numer. Anal., 35 (1998), pp. 2223–2249.

[38] F. Coquel, M. Postel, N. Poussineau, and Q. H. Tran, *Multiresolution technique and explicit-implicit scheme for multicomponent flows*, J. Numer. Math., 14 (2006), pp. 187–216.

[39] C. Dafermos, *Hyperbolic Conservation Laws in Continuum Physics*, vol. 325 of Grundlehren der mathematischen Wissenschaften, Springer-Verlag, Berlin, 2000.

[40] C. Dawson and R. Kirby, *High resolution schemes for conservation laws with locally varying time steps*, SIAM J. Sci. Comput., 22 (2000), pp. 2256–2281.

[41] B. Després, *Invariance properties of Lagrangian systems of conservation laws, approximate Riemann solvers and the entropy condition*, Numer. Math., 89 (2001), pp. 99–134.

[42] B. Després and C. Mazeran, *Lagrangian gas dynamics in two-dimensions and Lagrangian systems*, Arch. Ration. Mech. Anal., 178 (2005), pp. 327–372.

[43] M. O. Domingues, S. M. Gomes, O. Roussel, and K. Schneider, *An adaptive multiresolution scheme with local time stepping for evolutionary PDEs*, J. Comput. Phys., 227 (2008), pp. 3758–3780.

[44] J. Donea, A. Huerta, J. P. Ponthot, and A. Rodríguez-Ferran, *Arbitrary Lagrangian-Eulerian methods*, in Encyclopedia of Computational Mechanics, E. Stein, R. de Borst, and T. Hughes, eds., vol. 1, John Wiley & Sons, 2004, ch. 14, pp. 413–437.

[45] F. Dubois, *Volumes finis pour la dynamique des gaz*, Rapport scientifique, 2002.

[46] F. Dubois and P. LeFloch, *Boundary conditions for nonlinear hyperbolic systems of conservation laws*, J. Diff. Eq., 71 (1988), pp. 93–122.

[47] E. Duret, *Dynamique et contrôle des écoulements polyphasiques*, Thèse de doctorat, École des Mines de Paris, 2005.

[48] S. Evje and T. Flatten, *Weakly implicit numerical schemes for a two-fluid model*, SIAM J. Sci. Comput., 26 (2005), pp. 1449–1484.

[49] I. Faille and E. Heintzé, *A rough finite volume scheme for modeling two phase flow in a pipeline*, Computers and Fluids, 28 (1999), pp. 213–241.

[50] T. Flatten and S. T. Munkejord, *The approximate Riemann solver of Roe applied to a drift-flux two-phase flow model*, Modél. Math. Anal. Numér., 40 (2006), pp. 735–764.

[51] T. Gallouët and J. M. Masella, *A rough Godunov scheme*, C. R. Acad. Sci. Paris, (1996), p. 77.

[52] E. Godlewski and P. A. Raviart, *Numerical Approximation of Hyperbolic Systems of Conservation Laws*, vol. 118 of Applied Mathematical Sciences, Springer-Verlag, New York, 1996.

[53] A. Harten, *Discrete multi-resolution analysis and generalized wavelets*, Appl. Numer. Math., 12 (1993), pp. 153–192. Special issue to honor Professor Saul Abarbanel on his sixtieth birthday (Neveh, 1992).

[54] ——, *Multi-resolution analysis for ENO schemes*, in Algorithmic trends in computational fluid dynamics (1991), ICASE/NASA LaRC Ser., Springer, New York, 1993, pp. 287–302.

[55] ——, *Adaptive multiresolution schemes for shock computations*, J. Comput. Phys., 115 (1994), pp. 319–338.

[56] ——, *Multiresolution algorithms for the numerical solutions of hyperbolic conservation laws*, Comm. Pure Appl. Math., 48 (1995), pp. 1305–1342.

[57] ——, *Multiresolution representation of data: a general framework*, SIAM J. Numer. Anal., 33 (1996), pp. 1205–1256.

[58] A. Harten, P. D. Lax, and B. van Leer, *On upstream differencing and Godunov-type schemes for hyperbolic conservation laws*, SIAM Review, 25 (1983), pp. 35–61.

[59] C. W. Hirt, A. A. Amsden, and J. L. Cook, *An arbitrary Lagrangian-Eulerian computing method for all flow speeds*, J. Comput. Phys., 14 (1974), pp. 227–253.

[60] S. Jin and M. Slemrod, *Regularization of the burnett equations for rapid granular flows via relaxation*, Physica D, 150 (2001), pp. 207–218.

[61] S. Jin and Z. P. Xin, *The relaxation schemes for systems of conservation laws in arbitrary space dimension*, Comm. Pure Appl. Math., 48 (1995), pp. 235–276.

[62] H. O. Kreiss, *Initial boundary value problems for hyperbolic equations*, Comm. Pure Appl. Math., 14 (1970), pp. 98–112.

[63] P. Lamby, S. Müller, and Y. Stiriba, *Solution of shallow water equations using fully adaptive multiscale schemes*, Int. J. Numer. Meth. Fluids, 49 (2005), pp. 417–437.

[64] B. Larrouturou, *How to preserve the mass fraction positivity when computing multi-component flows*, J. Comput. Phys., 95 (1991), pp. 59–84.

[65] R. J. LeVeque, *Numerical Methods for Conservation Laws*, Lectures in Mathematics, ETH Zürich, Birkhäuser Verlag, Berlin, 1992.

[66] ——, *Finite Volume Methods for Hyperbolic Problems*, vol. 31 of Cambridge Texts in Applied Mathematics, Cambridge University Press, Cambridge, 2002.

[67] L. Linise, *Nouvelles techniques numériques pour la simulation des écoulements diphasiques dans les conduites pétrolières*, Rapport de stage (non référencé), Institut Français du Pétrole, 2004.

[68] J. L. Lions, Y. Maday, and G. Turinici, *Résolution d'EDP par un schéma en temps pararéel*, C. R. Acad. Sci. I Math., (2001), p. 332.

[69] T. P. Liu, *Hyperbolic conservation laws with relaxation*, Comm. Math. Phys., 108 (1987), pp. 153–175.

[70] T. P. Liu, J. Wang, and T. Yang, *Stability for a relaxation model with a nonconvex flux*, SIAM. J. Math. Anal., 29 (1998), pp. 19–29.

[71] J. M. Masella, *Quelques méthodes numériques pour les écoulements diphasiques bifluides en conduites pétrolières*, Thèse de doctorat, Université Pierre et Marie Curie (Paris 6), 1997.

[72] J. M. Masella, I. Faille, and T. Gallouët, *On an approximate Godunov scheme*, Int. J. Comput. Fluid Dynam., 12 (1999), pp. 133–149.

[73] S. Müller, *Adaptive Multiscale Schemes for Conservation Laws*, vol. 27 of Lecture Notes on Computational Science and Engineering, Springer, Berlin, 2003.

[74] S. Müller, P. Helluy, and J. Ballmann, *Numerical simulation of cavitation bubbles by compressible two-phase fluids*. Report No. 273, IGPM, RWTH Aachen, 2007.

[75] S. Müller and Y. Stiriba, *Fully adaptive multiscale schemes for conservation laws employing locally varying time stepping*, J. Sci. Comput., 30 (2007), pp. 493–531.

[76] S. M. N. Hovhannisyan, *On the stability of fully adaptive multiscale schemes for conservation laws using approximate flux and source reconstruction strategies*, Rapport 284, IGPM-RWTH Aachen, 2008.

[77] Q. L. Nguyen, *Adaptation dynamique de maillage pour les écoulements diphasiques en conduites pétrolières : application à la simulation des phénomènes de terrain-slugging et severe-slugging*, Rapport IFP 60001, Institut Français du Pétrole, 2007.

[78] S. Osher and R. Sanders, *Numerical approximations to nonlinear conservation laws with locally varying time and space grids*, Math. Comp., 41 (1983), pp. 321–336.

[79] S. Patault and Q. H. Tran, *Modèle et schéma numérique du code TACITE-NPW*, Rapport IFP 42415, Institut Français du Pétrole, 1996.

[80] C. Pauchon, H. Dhulesia, G. Binh-Cirlot, and J. Fabre, *TACITE: A transient tool for multiphase pipeline and well simulation*, SPE Annual Technical Conference and Exhibition, New Orleans, September 1994, 1994. SPE Paper 28545.

[81] N. Poussineau, *Analyse multi-résolution des écoulements diphasiques dans les conduites pétrolières*, Rapport IFP 58272, Institut Français du Pétrole, 2004.

[82] O. Roussel, K. Schneider, and M. Farge, *Coherent vortex extraction in 3D homogeneous turbulence: comparison between orthogonal and biorthogonal wavelet decompositions*, J. Turbul., 6 (2005), pp. Paper 11, 15 pp.

[83] B. Scheurer, *Introduction à l'analyse des écoulements réactifs dans le code KIVA*, Rapport IFP 38045, Institut Français du Pétrole, 1990.

[84] N. Seguin, *Modélisation et simulation numérique des écoulements diphasiques*, Thèse de doctorat, Université de Provence, 2002.

[85] D. Serre, *Systèmes de Lois de Conservation I & II*, Collection Fondations, Diderot Éditeur Arts et Sciences, Paris, 1996.

[86] ——, *Relaxation semi-linéaires et cinétique des systèmes de lois de conservation*, Ann. Inst. Henri Poincaré, 17 (2000), pp. 169–192. Analyse non linéaire.

[87] J. Smoller, *Shock Waves and Reaction-Diffusion Equations*, vol. 258 of Grundlehren der mathematischen Wissenschaften, Springer-Verlag, New York, 1994.

[88] P. K. Sweby, *High resolution schemes using flux limiters for hyperbolic conservation laws*, SIAM J. Numer. Anal., 21 (1984), pp. 995–1011.

[89] H. Z. Tang and G. Warnecke, *High resolution schemes for hyperbolic conservation laws and convection-diffusion equations with varying time and space grids*, J. Comput. Math., 24 (2006), pp. 121–140.

[90] E. F. Toro, *Riemann Solvers and Numerical Methods for Fluid Dynamics: A Practical Introduction*, Springer-Verlag, Berlin, 1997.

[91] Q. H. Tran, *Schémas de type multidimensionnel en maillage déformé structuré pour l'advection scalaire linéaire VI: Intégration du schéma d'Iserles dans la phase transport d'un code d'écoulement fluide non-linéaire 1-D*, Rapport IFP 59638, Institut Français du Pétrole, 2007.

[92] ——, *Second-order slope limiters for the simultaneous linear advection of (not so) independent variables*, Comme. Math . Sci., 6 (2008), pp. 569–593.

[93] Q. H. Tran, I. Faille, C. Pauchon, and F. Willien, *The No Pressure Wave (NPW) model: Application to oil and gas transport*, in Proceedings of the 3rd Symposium on Finite Volumes for Complex Applications, Porquerolles, June 2002, R. Herbin and D. Kröner, eds., vol. 3 of Finite Volumes for Complex Applications, London, 2002, Hermes Penton Science, pp. 687–694.

[94] B. Van Leer, *Towards the ultimate conservative difference scheme V: A second-order sequel to Godunov's method*, J. Comput. Phys., 32 (1979), pp. 101–136.

[95] H. Viviand, *Analyse de modèles simplifiés d'écoulements diphasiques transitoires*, Rapport d'étude Principia IFP.RET.43.004.01, Principia Recherche Développement S.A., La Seyne-sur-Mer, 1994.

[96] ——, *Modèles simplifiés d'écoulements dans les conduites pétrolières*, Rapport d'étude Principia IFP.RET.43.069.02, Principia Recherche Développement S.A., La Seyne-sur-Mer, 1996.

[97] D. H. Wagner, *Equivalence of the Euler and Lagrangian equations of gas dynamics for weak solutions*, J. Diff. Eq., 68 (1987), pp. 118–136.

[98] G. B. Whitham, *Linear and Nonlinear Waves*, vol. 258 of Pure and Applied Mathematics, Wiley-Interscience, New York, 1974.

[99] N. Zuber and J. Findlay, *Average volumetric concentration in two-phase flow systems*, J. Heat Transfer, 87 (1965), pp. 453–458.

Annexe A

Détails de majoration de $|v^*_{i+1/2}|$

Cette annexe technique fournit les détails des 4 étapes évoquées en 3.5.6.1 pour la mojoration de $|v^*_{i+1/2}|$.

A.1 Modification du système

Si, pour le calcul effectif d'une itération, on rédout le système linéaire (3.135), pour l'estimation de Δt on néglige l'effet de $(\partial_v P)_i$ ainsi que $(\partial_Y P)_i$ dans l'implicitation de la projection à l'équilibre $\widetilde{\Pi}_i = P(\widetilde{\tau}_i, \widetilde{Y}_i, \widetilde{v}_i)$. C'est-à-dire que $\delta\Pi_i = (\partial_\tau P)_i \delta\tau_i$ ne dépend que de l'incrément $\delta\tau_i$. De plus, vu le contenu du système par blocs 2×2, on décide de remplacer tous les $(\partial_\tau P)_i$ par $-a^2$, qui en est un majorant d'après la condition de Whitham (2.46). L'intérêt de cette opération de sur-dissipation est de simplifier la structure du problème. Le système (3.107) devient maintenant

$$\left[\begin{array}{cc|cc|cc} -\mu_i\frac{a}{2} & \frac{\mu_i}{2} & 1+\mu_i a & 0 & -\mu_i\frac{a}{2} & -\frac{\mu_i}{2} \\ \mu_i\frac{a^2}{2} & -\mu_i\frac{a}{2} & 0 & 1+\mu_i a & -\mu_i\frac{a^2}{2} & -\mu_i\frac{a}{2} \end{array}\right] \cdot \left[\begin{array}{c} \delta\tau_{i-1} \\ \hline \delta v_{i-1} \\ \delta\tau_i \\ \hline \delta v_i \\ \delta\tau_{i+1} \\ \hline \delta v_{i+1} \end{array}\right] = \left[\begin{array}{c} (\widetilde{\mathcal{R}}_\tau)^n_i \\ (\widetilde{\mathcal{R}}_v)^n_i \end{array}\right]. \tag{A.1}$$

La matrice au premier membre se compose de 3 blocs 2×2. Les premier et troisième blocs sont diagonalisables dans la même base et que les vecteurs propres sont très simples grâce à la modification du système. On remarque aussi que le deuxième bloc est proportionel à la matrice d'identité. Ceci facilite davantage la résolution comme on verra plus loin.

A.2 Changement de variables

En introduisant maintenant deux nouvelles variables

$$\begin{aligned} \delta C_i^+ &= \frac{\delta v_i + a\delta\tau_i}{2}, \\ \delta C_i^- &= \frac{\delta v_i - a\delta\tau_i}{2}, \end{aligned} \tag{A.2}$$

on peut transformer facilement (A.1) en un système matriciel sur $\delta C_i^{\pm}$, à savoir

$$\begin{aligned}
\left[\begin{array}{ccc} 0 & 1+a\mu_i & -a\mu_i \end{array}\right] \cdot \left[\begin{array}{c} \delta C_{i-1}^{+} \\ \delta C_i^{+} \\ \delta C_{i+1}^{+} \end{array}\right] &= (\mathcal{R}_{\mathcal{C}^+})_i^n, \\
\left[\begin{array}{ccc} -a\mu_i & 1+a\mu_i & 0 \end{array}\right] \cdot \left[\begin{array}{c} \delta C_{i-1}^{-} \\ \delta C_i^{-} \\ \delta C_{i+1}^{-} \end{array}\right] &= (\mathcal{R}_{\mathcal{C}^-})_i^n.
\end{aligned} \tag{A.3}$$

avec

$$\begin{aligned}
(\mathcal{R}_{\mathcal{C}^+})_i^n &= \frac{(\widetilde{\mathcal{R}}_v)_i^n + a(\widetilde{\mathcal{R}}_\tau)_i^n}{2}, \\
(\mathcal{R}_{\mathcal{C}^-})_i^n &= \frac{(\widetilde{\mathcal{R}}_v)_i^n - a(\widetilde{\mathcal{R}}_\tau)_i^n}{2}.
\end{aligned} \tag{A.4}$$

A.2.1 Conditions limites pour le système $\delta C^{\pm}$

Que se passe-t-il pour $i = 1$ et $i = N+1$? Est-ce que l'on a toujours les relations $\delta\Pi_0 = -a^2\delta\tau_0$ et $\delta\Pi_{N+1} = -a^2\delta\tau_{N+1}$? Étudions maintenant les traitements aux limites pour ce systèmes de $\delta C_i^{\pm}$.

À l'amont, la suppression d'onde $\Pi + a^2\tau$ donne :

$$\delta\Pi_0 + a^2\delta\tau_0 = \delta\Pi_1 + a^2\delta\tau_1 = 0 \quad \text{car} \quad \delta\Pi_1 = -a^2\delta\tau_1, \tag{A.5}$$

d'où $\delta\Pi_0 = -a^2\delta\tau_0$. En conséquence, la relation $\delta\Pi_0 - a\delta v_0 = \delta\Pi_1 - a\delta v_1$ (suppression de l'onde $\Pi - av$) devient

$$\begin{aligned}
-a^2\delta\tau_0 - a\delta v_0 &= -a^2\delta\tau_1 - a\delta v_1 \\
\Leftrightarrow \qquad \delta v_0 + a\delta\tau_0 &= \delta v_1 + a\delta\tau_1 \\
\Leftrightarrow \qquad \delta C_0^+ &= \delta C_1^+
\end{aligned} \tag{A.6}$$

De plus, en remplaçant $\delta\tau_0 = \dfrac{\delta C_0^+ - \delta C_0^-}{a}$ et $\delta v_0 = \delta C_0^+ + \delta C_0^-$ dans la condition (3.113), nous avons

$$\begin{aligned}
q_T\left(\frac{\delta C_0^+ - \delta C_0^-}{a}\right) - \theta(\delta C_0^+ + \delta C_0^-) &= -\tau_0\delta q_T \\
\Leftrightarrow \qquad (\theta + \frac{q_T}{a})\delta C_0^- + (\theta - \frac{q_T}{a})\delta C_0^+ &= \tau_0\delta q_T
\end{aligned} \tag{A.7}$$

De même à l'aval, on peut vérifier que $\delta\Pi_{N+1} = -a^2\delta\tau_{N+1}$ (suppression d'onde $\Pi + a^2\tau$) d'où

$$\delta\Pi_{N+1} = -a(\delta C_{N+1}^+ - \delta C_{N+1}^-). \tag{A.8}$$

L'élimination de l'onde $\Pi + av$ nous donne

$$\begin{aligned}
\delta\Pi_{N+1} + a\delta v_{N+1} &= \delta\Pi_N + a\delta v_N \\
-a^2\delta\tau_{N+1} + a\delta v_{N+1} &= -a^2\delta\tau_N + a\delta v_N \\
\Leftrightarrow \qquad \delta v_{N+1} - a\delta\tau_{N+1} &= \delta v_N - a\delta\tau_N \\
\Leftrightarrow \qquad \delta C_{N+1}^- &= \delta C_N^-
\end{aligned} \tag{A.9}$$

Les conditions limites pour le système $\delta C^{\pm}$ sont donc

$$\begin{aligned} \delta C_0^+ &= \delta C_1^+, \\ (\theta + \frac{q_T}{a})\delta C_0^- + (\theta - \frac{q_T}{a})\delta C_0^+ &= \tau_0 \delta q_T, \\ \delta C_{N+1}^- &= \delta C_N^-, \\ \delta C_{N+1}^- - \delta C_{N+1}^+ &= \frac{\delta \Pi_{N+1}}{a}. \end{aligned} \tag{A.10}$$

A.2.2 Systèmes définitifs en $\delta C^{\pm}$

Le système entier se décompose alors en deux sous-systèmes presque découplés, c'est-à-dire

$$\begin{bmatrix} (*) & 0 & \cdots & \cdots & \cdots & \cdots & \cdots & 0 \\ -a\mu_1 & 1+a\mu_1 & \ddots & & & & & \vdots \\ 0 & -\alpha\mu_2 & 1+a\mu_2 & \ddots & & & & \vdots \\ \vdots & \ddots & \ddots & \ddots & \ddots & & & \vdots \\ \vdots & & \ddots & \ddots & \ddots & \ddots & & \vdots \\ \vdots & & & \ddots & -a\mu_{N-1} & 1+a\mu_{N-1} & \ddots & \vdots \\ \vdots & & & & \ddots & -a\mu_N & 1+a\mu_N & 0 \\ 0 & \cdots & \cdots & \cdots & \cdots & \cdots & -1 & 1 \end{bmatrix} \cdot \begin{bmatrix} \delta C_0^- \\ \delta C_1^- \\ \delta C_2^- \\ \cdots \\ \cdots \\ \delta C_{N-1}^- \\ \delta C_N^- \\ \delta C_{N+1}^- \end{bmatrix} = \begin{bmatrix} (\mathcal{R}_{\mathcal{C}^-})_0^n \\ (\mathcal{R}_{\mathcal{C}^-})_1^n \\ (\mathcal{R}_{\mathcal{C}^-})_2^n \\ \cdots \\ \cdots \\ (\mathcal{R}_{\mathcal{C}^-})_{N-1}^n \\ (\mathcal{R}_{\mathcal{C}^-})_N^n \\ (\mathcal{R}_{\mathcal{C}^-})_{N+1}^n \end{bmatrix}, \tag{A.11}$$

et

$$\begin{bmatrix} 1 & -1 & 0 & \cdots & \cdots & \cdots & \cdots & 0 \\ 0 & 1+a\mu_1 & -a\mu_1 & \ddots & & & & \vdots \\ \vdots & \ddots & 1+a\mu_2 & -a\mu_1 & \ddots & & & \vdots \\ \vdots & & \ddots & \ddots & \ddots & \ddots & & \vdots \\ \vdots & & & \ddots & \ddots & \ddots & \ddots & \vdots \\ \vdots & & & & \ddots & 1+a\mu_{N-1} & -a\mu_{N-1} & 0 \\ \vdots & & & & & \ddots & 1+a\mu_N & -a\mu_N \\ 0 & \cdots & \cdots & \cdots & \cdots & \cdots & 0 & (*) \end{bmatrix} \begin{bmatrix} \delta C_0^+ \\ \delta C_1^+ \\ \delta C_2^+ \\ \cdots \\ \cdots \\ \delta C_{N-1}^+ \\ \delta C_N^+ \\ \delta C_{N+1}^+ \end{bmatrix} . = \begin{bmatrix} (\mathcal{R}_{\mathcal{C}^+})_0^n \\ (\mathcal{R}_{\mathcal{C}^+})_1^n \\ (\mathcal{R}_{\mathcal{C}^+})_2^n \\ \cdots \\ \cdots \\ (\mathcal{R}_{\mathcal{C}^+})_{N-1}^n \\ (\mathcal{R}_{\mathcal{C}^+})_N^n \\ (\mathcal{R}_{\mathcal{C}^+})_{N+1}^n \end{bmatrix}. \tag{A.12}$$

Les termes $(*)$ proviennent du fait que les deux systèmes (A.12) et (A.11) sont couplés. On constate que la matrice à gauche de (A.12) est bidiagonale supérieure et celle de (A.11) est bidiagonale inférieure. On utilisera donc une méthode itérative avec des algorithmes de descente et de montée (balayage) pour les résoudre (voir (A.20))

Remarque *A.1.* Pour un problème avec des conditions limites quelconques nous pouvons choisir un point fixe de telle manière que la méthode converge en deux itérations seulement. Par contre, pour la condition limite de type Neumann, un seul balayage suffira.

A.3 Résolution à la main du système $\delta C^{\pm}$

L'idée est d'effectuer le double balayage $(*)$ à la main afin d'obtenir des formules explicites pour la solution. Ces formules explicites font intervenir les grandeurs suivantes.

Définition *A.2.*

$$e_i = \frac{a\mu_i}{1+a\mu_i} \quad pour\ i = 1,..,N, \tag{A.13}$$

$$E^i_j = \begin{cases} \prod_{k=j}^{i} e_k & si\ j \le i, \\ 1 & si\ j > i, \end{cases} \tag{A.14}$$

$$\Xi = \frac{1}{1+\Theta(E^N_1)^2}, \tag{A.15}$$

$$\Theta = \frac{\theta - \dfrac{q_T}{a}}{\theta + \dfrac{q_T}{a}}. \tag{A.16}$$

Étant donné δC^-_0, à partir du système (A.11) nous avons pour $i = 1,..,N$

$$\begin{aligned} & -a\mu_i \delta C^-_{i-1} + (1+a\mu_i)\delta C^-_i = (\mathcal{R}_{\mathcal{C}^-})^n_i \\ \Leftrightarrow \quad & \delta C^-_i = \frac{a\mu_i}{1+a\mu_i}\delta C^-_{i-1} + \frac{1}{1+a\mu_i}(\mathcal{R}_{\mathcal{C}^-})^n_i \\ \Leftrightarrow \quad & \delta C^-_i = e_i \delta C^-_{i-1} + (1-e_i)(\mathcal{R}_{\mathcal{C}^-})^n_i \end{aligned} \tag{A.17}$$

On peut calculer donc δC^-_N ce qui permet de calculer δC^-_{N+1} puis δC^+_{N+1} et δC^+_N grâce au (A.10). Notons que la dernière formule justifie le nom de « coefficient local de propagation » pour e_i.

Le système (A.12) nous donne

$$\delta C^+_{i-1} = e_{i-1}\delta C^+_i + (1-e_{i-1})(\mathcal{R}_{\mathcal{C}^+})^n_{i-1}. \tag{A.18}$$

On peut donc calculer δC^+_0 puis le nouveau δC^-_0 grâce à la relation du couplage (A.7)

$$\delta C^-_0 = -\Theta \delta C^+_0 + \frac{1+\Theta}{2\theta}\delta q_T \tag{A.19}$$

$$\begin{array}{ccccccccccc} \boxed{F(\delta C^-_0)\,|\,\delta C^-_0} & \xrightarrow{(A.17)} & \delta C^-_1 & \xrightarrow{(A.17)} & \dots & \delta C^-_i & \dots & \xrightarrow{(A.17)} & \delta C^-_N & \xrightarrow{(A.9)} & \delta C^-_{N+1} \\ \uparrow{\scriptstyle (A.7)} & & & & & & & & & & \downarrow{\scriptstyle (A.8)} \\ \delta C^+_0 & \xleftarrow[(A.6)]{} & \delta C^+_1 & \xleftarrow[(A.18)]{} & \dots & \delta C^+_i & \dots & \xleftarrow[(A.18)]{} & \delta C^+_N & \xleftarrow[(A.18)]{} & \delta C^+_{N+1} \end{array} \tag{A.20}$$

Remarque *A.3.* En pratique, nous utilisons relations de récurrence suivantes de e_i pour calculer les $\delta C^{\pm}_i$, pour $i = 1,..,N$

$$\begin{aligned} \delta C^-_i &= E^i_1 \delta C^-_0 \quad + \sum_{k=1}^{i} (E^i_{k+1} - E^i_k)(\mathcal{R}_{\mathcal{C}^-})^n_k \\ \delta C^+_i &= E^N_i \delta C^+_{N+1} + \sum_{k=i}^{N} (E^{k-1}_i - E^k_i)(\mathcal{R}_{\mathcal{C}^+})^n_k \end{aligned} \tag{A.21}$$

A.3.1 Balayage gauche-droite

On commence par évaluer successivement les δC_i^-, en supposant δC_0^- connu

$$\begin{aligned}
\delta C_0^- &= -\Theta\delta C_1^+ + \frac{1+\Theta}{2\theta}\delta q_T \\
&= -\Theta\left[E_1^N\delta C_{N+1}^+ + \sum_{k=1}^{N}(E_1^{k-1}-E_1^k)(\mathcal{R}_{\mathcal{C}^+})_k^n\right] + \frac{1+\Theta}{2\theta}\delta q_T \\
&= -\Theta\left[E_1^N(\delta C_{N+1}^- - \frac{\delta\Pi_{N+1}}{a}) + \sum_{k=1}^{N}(E_1^{k-1}-E_1^k)(\mathcal{R}_{\mathcal{C}^+})_k^n\right] + \frac{1+\Theta}{2\theta}\tau_0\delta q_T \\
&= -\Theta\left[E_1^N\left[E_1^N\delta C_0^- + \sum_{k=1}^{N}(E_{k+1}^N-E_k^N)(\mathcal{R}_{\mathcal{C}^-})_k^n\right] + \sum_{k=1}^{N}(E_1^{k-1}-E_1^k)(\mathcal{R}_{\mathcal{C}^+})_k^n\right] \\
&\quad + \frac{1+\Theta}{2\theta}\tau_0\delta q_T + \Theta E_1^N\frac{\delta\Pi_{N+1}}{a}
\end{aligned} \tag{A.22}$$

d'où

$$\begin{aligned}
(1+\Theta(E_1^N)^2)\delta C_0^- &= -\Theta\left[E_1^N\sum_{k=1}^{N}(E_{k+1}^N-E_k^N)(\mathcal{R}_{\mathcal{C}^-})_k^n + \sum_{k=1}^{N}(E_1^{k-1}-E_1^k)(\mathcal{R}_{\mathcal{C}^+})_k^n\right] \\
&\quad + \frac{1+\Theta}{2\theta}\tau_0\delta q_T + \Theta E_1^N\frac{\delta\Pi_{N+1}}{a} \\
\Longrightarrow \qquad \delta C_0^- &= -\Theta\Xi\left[E_1^N\sum_{k=1}^{N}(E_{k+1}^N-E_k^N)(\mathcal{R}_{\mathcal{C}^-})_k^n + \sum_{k=1}^{N}(E_1^{k-1}-E_1^k)(\mathcal{R}_{\mathcal{C}^+})_k^n\right] \\
&\quad + \frac{(1+\Theta)\Xi}{2\theta}\tau_0\delta q_T + \frac{\Theta\Xi E_1^N}{a}\delta\Pi_{N+1}
\end{aligned} \tag{A.23}$$

En remplaçant (A.23) dans (A.21) on a

$$\begin{aligned}
\delta C_i^- &= -\Theta\Xi\left[E_1^iE_1^N\sum_{k=1}^{N}(E_{k+1}^N-E_k^N)(\mathcal{R}_{\mathcal{C}^-})_k^n + E_1^i\sum_{k=1}^{N}(E_1^{k-1}-E_1^k)(\mathcal{R}_{\mathcal{C}^+})_k^n\right] \\
&\quad + \frac{(1+\Theta)\Xi E_1^i}{2\theta}\tau_0\delta q_T + \frac{\Theta\Xi E_1^iE_1^N}{a}\delta\Pi_{N+1} + \sum_{k=1}^{i}(E_{k+1}^i-E_k^i)(\mathcal{R}_{\mathcal{C}^-})_k^n \\
&= -\Theta\Xi E_1^iE_1^N\sum_{k=1}^{N}(E_{k+1}^N-E_k^N)(\mathcal{R}_{\mathcal{C}^-})_k^n + \sum_{k=1}^{N}(E_{k+1}^i-E_k^i)(\mathcal{R}_{\mathcal{C}^-})_k^n \\
&\quad -\Theta\Xi E_1^i\sum_{k=1}^{N}(E_1^{k-1}-E_1^k)(\mathcal{R}_{\mathcal{C}^+})_k^n + \frac{(1+\Theta)\Xi E_1^i}{2\theta}\tau_0\delta q_T + \frac{\Theta\Xi E_1^iE_1^N}{a}\delta\Pi_{N+1}
\end{aligned} \tag{A.24}$$

d'où

$$\delta C_i^- = \sum_{k=1}^{N}\alpha_{i,k}^-(\mathcal{R}_{\mathcal{C}^-})_k^n + \sum_{k=1}^{N}\beta_{i,k}^-(\mathcal{R}_{\mathcal{C}^+})_k^n + \gamma_{i,k}^-\delta q_T + \chi_{i,k}^-\delta\Pi_{N+1}, \tag{A.25}$$

avec

$$\begin{aligned}
\alpha_{i,k}^- &= -\Theta\Xi E_1^iE_1^N(E_{k+1}^N-E_k^N) + (E_{k+1}^i-E_k^i), \\
\beta_{i,k}^- &= -\Theta\Xi E_1^i(E_1^{k-1}-E_1^k), \\
\gamma_{i,k}^- &= \frac{(1+\Theta)\Xi E_1^i}{2\theta}\tau_0, \\
\chi_{i,k}^- &= \frac{\Theta\Xi E_1^iE_1^N}{a}.
\end{aligned} \tag{A.26}$$

A.3.2 Balayage droite-gauche

De la même façon, après quelques calculs nous avons

$$\begin{aligned}\delta C^+_{N+1} &= \delta C^-_{N+1} - \frac{\delta\Pi_{N+1}}{a}\\ &= \delta C^-_N - \frac{\delta\Pi_{N+1}}{a}\\ &= \sum_{k=1}^{N}\alpha^-_{N,k}(\mathcal{R}_{\mathcal{C}^-})^n_k + \sum_{k=1}^{N}\beta^-_{N,k}(\mathcal{R}_{\mathcal{C}^+})^n_k + \gamma^-_{N,k}\delta q_T + \chi^-_{N,k}\delta\Pi_{N+1} - \frac{\delta\Pi_{N+1}}{a}\end{aligned} \tag{A.27}$$

avec

$$\begin{aligned}\alpha^-_{N,k} &= -\Theta\Xi E^N_1 E^N_1(E^N_{k+1} - E^N_k) + (E^N_{k+1} - E^N_k)\\ &= (1-\Theta\Xi(E^N_1)^2)(E^N_{k+1} - E^N_k)\\ &= \Xi(E^N_{k+1} - E^N_k),\\ \beta^-_{N,k} &= \Theta\Xi E^N_1(E^{k-1}_1 - E^k_1),\\ \gamma^-_{N,k} &= \frac{(1+\Theta)\Xi E^N_1}{2\theta}\tau_0,\\ \chi^-_{N,k} &= \frac{\Theta\Xi(E^N_1)^2}{a}\\ \chi^-_{N,k} - \frac{1}{a} &= -\frac{\Xi}{a}.\end{aligned} \tag{A.28}$$

En remplaçant (A.27) dans la deuxième équation de (A.21) nous avons

$$\begin{aligned}\delta C^+_i &= E^N_i\left[\sum_{k=1}^{N}\alpha^-_{N,k}(\mathcal{R}_{\mathcal{C}^-})^n_k + \sum_{k=1}^{N}\beta^-_{N,k}(\mathcal{R}_{\mathcal{C}^+})^n_k + \gamma^-_{N,k}\delta q_T + \chi^-_{N,k}\delta\Pi_{N+1} - \frac{\delta\Pi_{N+1}}{a}\right]\\ &\qquad + \sum_{k=i}^{N}(E^{k-1}_i - E^k_i)(\mathcal{R}_{\mathcal{C}^+})^n_k\\ &= E^N_i\left[\sum_{k=1}^{N}\alpha^-_{N,k}(\mathcal{R}_{\mathcal{C}^-})^n_k + \sum_{k=1}^{N}\beta^-_{N,k}(\mathcal{R}_{\mathcal{C}^+})^n_k + \gamma^-_{N,k}\delta q_T + \chi^-_{N,k}\delta\Pi_{N+1} - \frac{\delta\Pi_{N+1}}{a}\right]\\ &\qquad + \sum_{k=1}^{N}(E^{k-1}_i - E^k_i)(\mathcal{R}_{\mathcal{C}^+})^n_k\end{aligned} \tag{A.29}$$

d'où

$$\delta C^+_i = \sum_{k=1}^{N}\alpha^+_{i,k}(\mathcal{R}_{\mathcal{C}^-})^n_k + \sum_{k=1}^{N}\beta^+_{i,k}(\mathcal{R}_{\mathcal{C}^+})^n_k + \gamma^+_{i,k}\delta q_T + \chi^+_{i,k}\delta\Pi_{N+1}, \tag{A.30}$$

avec

$$\begin{aligned}\alpha^+_{i,k} &= E^N_i\alpha^-_{N,k} &&= \Xi E^N_i(E^N_{k+1} - E^N_k),\\ \beta^+_{i,k} &= E^N_i\beta^-_{N,k} + (E^{k-1}_i - E^k_i) &&= -\Theta\Xi E^N_1 E^N_i(E^{k-1}_1 - E^k_1) + (E^{k-1}_i - E^k_i),\\ \gamma^+_{i,k} &= E^N_i\gamma^-_{N,k} &&= \frac{(1+\Theta)\Xi E^N_1 E^N_i}{2\theta}\tau_0,\\ \chi^+_{i,k} &= E^N_i(\chi^-_{N,k} - \frac{1}{a}) &&= -\frac{\Xi E^N_i}{a}.\end{aligned} \tag{A.31}$$

A.3.3 Valeur de la vitesse aux arêtes

On a $\delta v^*_{i+1/2} = \delta C^-_i + \delta C^+_{i+1}$ donc d'après (A.25) et (A.30) nous avons

$$\delta v^*_{i+1/2} = \sum_{k=1}^{N} \alpha_{i+1/2,k} (\mathcal{R}_{\mathcal{C}^-})^n_k + \sum_{k=1}^{N} \beta_{i+1/2,k} (\mathcal{R}_{\mathcal{C}^+})^n_k + \gamma_{i+1/2,k} \delta q_T + \chi_{i+1/2,k} \delta \Pi_{N+1}. \tag{A.32}$$

Nous allons maintenant calculer les coefficients $\alpha_{i+1/2,k}, \beta_{i+1/2,k}, \gamma_{i+1/2,k}, \chi_{i+1/2,k}$. Par définition

$$\begin{aligned} \alpha_{i+1/2,k} &= \alpha^-_{i,k} + \alpha^+_{i+1,k} \\ &= -\Theta\Xi E^i_1 E^N_1 (E^N_{k+1} - E^N_k) + (E^i_{k+1} - E^i_k) + \Xi E^N_{i+1} (E^N_{k+1} - E^N_k) \\ &= \Xi (E^N_{k+1} - E^N_k)(E^N_{i+1} - \Theta E^i_1 E^N_1) + (E^i_{k+1} - E^i_k) \end{aligned} \tag{A.33}$$

On distingue donc deux configurations

- Si $k \leq i$ on a $E^N_{k+1} - E^N_k = (E^i_{k+1} - E^i_k) E^N_{i+1}$ et dans ce cas

$$\begin{aligned} \alpha_{i+1/2,k} &= (E^i_{k+1} - E^i_k) \left[\Xi (E^N_{i+1})^2 - \Theta\Xi E^N_{i+1} E^i_1 E^N_1 + 1 \right] \\ &= (E^i_{k+1} - E^i_k) \left[\Xi (E^N_{i+1})^2 + (1 - \Theta\Xi (E^N_1)^2) \right] \\ &= (E^i_{k+1} - E^i_k) \left[\Xi (E^N_{i+1})^2 + \Xi \right] \\ &= \Xi (E^i_{k+1} - E^i_k) \left[1 + (E^N_{i+1})^2 \right] \end{aligned} \tag{A.34}$$

- Si $k > i$ on a $E^i_{k+1} - E^i_k = 0$ et dans ce cas

$$\begin{aligned} \alpha_{i+1/2,k} &= \Xi (E^N_{k+1} - E^N_k)(E^N_{i+1} - \Theta E^i_1 E^i_1 E^N_{i+1}) \\ &= \Xi E^N_{i+1} (E^N_{k+1} - E^N_k) \left[1 - \Theta (E^i_1)^2 \right] \end{aligned} \tag{A.35}$$

Donc

$$\alpha_{i+1/2,k} = \left\{ \Xi (E^i_{k+1} - E^i_k) \left[1 + (E^N_{i+1})^2 \right] \right\} \mathbb{1}_{\{k | 1 \leq k \leq i\}} + \left\{ \Xi E^N_{i+1} (E^N_{k+1} - E^N_k) \left[1 - \Theta (E^i_1)^2 \right] \right\} \mathbb{1}_{\{i+1 \leq k \leq N\}}. \tag{A.36}$$

Le coefficient $\beta_{i+1/2,k}$ est calculé de la même manière. on a

$$\begin{aligned} \beta_{i+1/2,k} &= \beta^-_{i,k} + \beta^+_{i+1,k} \\ &= -\Theta\Xi E^i_1 (E^{k-1}_1 - E^k_1) - \Theta\Xi E^N_1 E^N_{i+1} (E^{k-1}_1 - E^k_1) + (E^{k-1}_{i+1} - E^k_{i+1}) \\ &= -\Theta\Xi (E^{k-1}_1 - E^k_1)(E^i_1 + E^N_1 E^N_{i+1}) + (E^{k-1}_{i+1} - E^k_{i+1}) \end{aligned} \tag{A.37}$$

Deux cas possibles

- Si $k \leq i$ on a $E^{k-1}_{i+1} - E^k_{i+1}$ et dans ce cas

$$\begin{aligned} \beta_{i+1/2,k} &= -\Theta\Xi (E^{k-1}_1 - E^k_1)(E^i_1 + E^i_1 E^N_{i+1} E^N_{i+1}) \\ &= -\Theta\Xi E^i_1 (E^{k-1}_1 - E^k_1) \left[1 + (E^N_{i+1})^2 \right] \end{aligned} \tag{A.38}$$

- Si $k > i$ on a $E_1^{k-1} - E_1^k = E_1^i(E_{i+1}^{k-1} - E_{i+1}^k)$ et dans ce cas

$$\begin{aligned}\beta_{i+1/2,k} &= (E_{i+1}^{k-1} - E_{i+1}^k)\left[1 - \Theta\Xi E_1^i(E_1^i + E_1^N E_{i+1}^N)\right] \\ &= (E_{i+1}^{k-1} - E_{i+1}^k)\left[(1 - \Theta\Xi(E_1^N)^2) - \Theta\Xi(E_1^i)^2\right] \\ &= (E_{i+1}^{k-1} - E_{i+1}^k)\left[\Xi - \Theta\Xi(E_1^i)^2\right] \\ &= \Xi(E_{i+1}^{k-1} - E_{i+1}^k)\left[1 - \Theta(E_1^i)^2\right]\end{aligned} \tag{A.39}$$

Donc

$$\beta_{i+1/2,k} = -\Theta\Xi E_1^i(E_1^{k-1} - E_1^k)\left[1 + (E_{i+1}^N)^2\right]\mathbb{1}_{\{k|1\le k\le i\}} + \Xi(E_{i+1}^{k-1} - E_{i+1}^k)\left[1 - \Theta(E_1^i)^2\right]\mathbb{1}_{\{k|i+1\le k\le N\}}. \tag{A.40}$$

Pour les deux autres coefficients

$$\begin{aligned}\gamma_{i+1/2,k} &= \gamma^-_{i,k} + \gamma^+_{i+1,k} \\ &= \frac{(1+\Theta)\Xi E_1^i}{2\theta}\tau_0 + \frac{(1+\Theta)\Xi E_1^N E_{i+1}^N}{2\theta}\tau_0 \\ &= \frac{(1+\Theta)\Xi E_1^i\left[1 + (E_{i+1}^N)^2\right]}{2\theta}\tau_0 \\ \chi_{i+1/2,k} &= \chi^-_{i,k} + \chi^+_{i+1,k} \\ &= \frac{\Theta\Xi E_1^i E_1^N}{a} - \frac{\Xi E_{i+1}^N}{a} \\ &= \frac{\Xi E_{i+1}^N\left[\Theta(E_1^i)^2 - 1\right]}{a}\end{aligned} \tag{A.41}$$

A.4 Majoration de $|\delta v^*_{i+1/2}|$

On va maintenant majorer les coefficients $\alpha_{i+1/2,k}, \beta_{i+1/2,k}, \gamma_{i+1/2,k}, \chi_{i+1/2,k}$. Nous savons que $0 \le \Xi, \le 1$ et $E_1^i \le E_k^i \le 1$ et $1 + (E_i^N)^2 \le 2$, $\forall 1 \le k \le i \le N$, donc

$$\begin{aligned}|\alpha_{i+1/2,k}| &\le 2(1 - E_1^i)\mathbb{1}_{\{k|1\le k\le i\}} + E_{i+1}^N(1 - E_{i+1}^N)\mathbb{1}_{\{k|i+1\le k\le N\}}, \\ |\beta_{i+1/2,k}| &\le 2\Theta E_1^i(1 - E_1^i)\mathbb{1}_{\{k|1\le k\le i\}} + (1 - E_{i+1}^N)\mathbb{1}_{\{k|i+1\le k\le N\}}, \\ |\gamma_{i+1/2,k}| &\le \frac{(1+\Theta)E_1^i}{\theta}\tau_0, \\ |\chi_{i+1/2,k}| &\le \frac{E_{i+1}^N}{a}.\end{aligned} \tag{A.42}$$

En conséquence $|\delta v^*_{i+1/2}|$ est majoré par

$$\begin{aligned}|\delta v^*_{i+1/2}| \le\ & 2(1 - E_1^i)\max_{1\le k\le i}|(\mathcal{R}_{\mathcal{C}^-})_k^n| + E_{i+1}^N(1 - E_{i+1}^N)\max_{i+1\le k\le N}|(\mathcal{R}_{\mathcal{C}^-})_k^n| \\ &+ 2\Theta E_1^i(1 - E_1^i)\max_{1\le k\le i}|(\mathcal{R}_{\mathcal{C}^+})_k^n| + (1 - E_{i+1}^N)\max_{i+1\le k\le N}|(\mathcal{R}_{\mathcal{C}^+})_k^n| \\ &+ \frac{2(1+\Theta)E_1^i}{2\theta}\tau_0|\delta q_T| + \frac{E_{i+1}^N}{a}|\delta\Pi_{N+1}|\end{aligned} \tag{A.43}$$

Or d'après (3.158) nous avons

$$\begin{aligned} |(\mathcal{R}_{\mathcal{C}^+})^n_k| &\leq \Delta t|(\mathcal{R}^{(1)}_{C^-})^n_k| + (\Delta t)^2|(\mathcal{R}^{(2)}_{C^-})^n_k|, \\ |(\mathcal{R}_{\mathcal{C}^-})^n_k| &\leq \Delta t|(\mathcal{R}^{(1)}_{C^+})^n_k| + (\Delta t)^2|(\mathcal{R}^{(2)}_{C^+})^n_k|. \end{aligned} \tag{A.44}$$

Par conséquent (A.43) s'écrit

$$\begin{aligned} |\delta v^*_{i+1/2}| \leq \sum_{p=1}^{2} (\Delta t)^p \Bigg\{ & 2(1-E^i_1) \max_{1\leq k\leq i} |(\mathcal{R}^{(p)}_{C^-})^n_k| \quad + E^N_{i+1}(1-E^N_{i+1}) \max_{i+1\leq k\leq N} |(\mathcal{R}^{(p)}_{C^-})^n_k| \\ & + \quad 2\Theta E^i_1(1-E^i_1) \max_{1\leq k\leq i} |(\mathcal{R}^{(p)}_{C^+})^n_k| + (1-E^N_{i+1}) \max_{i+1\leq k\leq N} |(\mathcal{R}^{(p)}_{C^+})^n_k| \Bigg\} \\ & + \quad \frac{2(1+\Theta)E^i_1}{2\theta}\tau_0|\delta q_T| \quad + \frac{E^N_{i+1}}{a}|\delta \Pi_{N+1}| \end{aligned} \tag{A.45}$$

Le second-membre apparaît comme une fonction quadratique de deux variables scalaires E^N_1 et E^N_{i+1}. Plus précisément, c'est la somme de 2 fonctions de E^N_1 et de 2 fonctions de E^N_{i+1}. Chacune de ces 4 fonctions est de la forme (3.165).

Proposition *A.1. Soit la fonction* $f : \xi \in [0,1] \longmapsto \mathbf{R}_+$ *définie par*

$$f(\xi) = (1-\xi)A + \xi B + \xi(1-\xi)D \quad \textit{avec } A, B, D \geq 0. \tag{A.46}$$

Alors son maximum est donné par

$$\max_{\xi\in[0,1]} f(\xi) = \varphi(A,B;D) := \begin{cases} \max(A;B) & \textit{si } D \leq |B-A|, \\ \dfrac{A+B}{2} + \dfrac{D}{4} + \dfrac{(B-A)^2}{4D} & \textit{si } D > |B-A|. \end{cases} \tag{A.47}$$

Démonstration Considérons $f(\xi)$ comme un polynôme du seconde degré en ξ alors

$$f(\xi) = -D\xi^2 + (D+B-A)\xi - A \tag{A.48}$$

Les dérivées

$$\begin{aligned} f'(\xi) &= -2D\xi + D + B - A, \\ f''(\xi) &= -2D < 0 \quad \forall D > 0 \end{aligned} \tag{A.49}$$

donc $f(\xi)$ est concave sur $\xi \in [0,1]$.

Sur l'intervalle $[0,1]$, $f(\xi)$ est monotone si et seulement si $f'(0)f'(1) \geq 0$, c'est-à dire

$$\begin{aligned} & (D+B-A)(-D+B-A) \geq 0 \\ \Longleftrightarrow \quad & C \leq |B-A| \end{aligned} \tag{A.50}$$

Donc si $D \leq |B-A|$, $\sup\limits_{\xi\in[0,1]} f(\xi) = \max(f(0);f(1)) = \max(A;B)$

Dans le cas contraire, un maximum $f(\xi^*)$ existe sur $]0,1[$ si et seulement si

$$f'(\xi^*) = 0 \Longleftrightarrow \xi^* = \frac{B+D-A}{2D} \tag{A.51}$$

et dans ce cas

$$f(\xi^*) = \frac{A+B}{2} + \frac{D}{4} + \frac{(B-A)^2}{4D} \tag{A.52}$$

d'où la preuve. □

Annexe B

Analyse de temps CPU

B.1 Introduction

Dans cette annexe, nous allons étudier la complexité des algorithmes du schéma Lagrange-Projection et la comparer avec celle du schéma eulérien. Nous dénombrerons les opérations nécessaires pour chaque étape de résolution afin de mieux voir la différence des deux méthodes et de montrer combien le schéma L-P est plus rapide que le schéma eulérien en analysant le temps CPU obtenu. Rappelons que la complexité des algorithmes dépend de :

1. la taille du système étudié (ici c'est le modèle diphasiques DFM avec trois équations),
2. le nombre de mailles N dans la discrétisation du domaine d'études (ici c'est la longueur du pipeline) sachant que le temps CPU est une fonction quadratique du nombre de mailles,
3. le schéma numérique (explicite, semi-implicite...) et la méthode utilisés (relaxation, L-P...),
4. les lois de fermetures (thermodynamiques et hydrodynamiques) qui sont très coûteuses en temps de calcul.

Nous allons détailler l'analyse des complexités algorithmiques pour deux schémas :

1. schéma explicite : fondé sur le calcul des flux numériques sur chaque arrête,
2. schéma semi-Implicite : fondé sur la construction et la résolution du système linéaire constitué de blocs de matrices 5×5 pour le schéma eulérien, et sur l'approximation a priori de pas de temps et la résolution du système linéaire par blocs de tailles 2×2 pour le schéma Lagrange-Projection.

Nous convenons d'appeler « opérations » les opérateurs $+, -, *, /, \sqrt{}, \max(,), \min(,)$ et « évaluations » les appels à une boîte « noire » pour avoir $P, \partial_\tau P, \partial_Y P, \partial_v P, \sigma$ et $\partial_Y \sigma$.

B.2 Analyse algorithmique pour les schémas explicites

On s'intéresse dans cette partie au calcul de la complexité des algorithmes des schémas eulérien et Lagrange-Projection explicites. Tout d'abord, nous comparons les calculs de coefficients de relaxation des deux schémas. Le schéma L-P utilise la même méthode de calcul pour les coefficients de relaxation utilisée dans [5, 6]. En revanche, nous ne devons pas ici ajouter des opérations supplémentaires sur a et b pour assurer la positivité de la densité et le principe du maximum sur la fraction massique du gaz. Les détails de calcul sont présentés dans les figures (B.1) et (B.2). Le schéma eulérien nécessite 30 opérations tandis que le schéma L-P ne nécessite que 12 opérations. Tous les deux schémas utilisent 10 évaluations de bases (calcul des dérivées).

Étape	**Nombre d'opérations**
1. Calcul de $(a^\flat, b^\flat)$ de base (voir (2.46) et (2.47)) $A_\kappa = -(\partial_\tau P)_\kappa + (\partial_v P)^2_\kappa, \qquad B_\kappa = (\partial_Y \sigma)^2_\kappa, \qquad \kappa = L, R$ $a^\flat = \sqrt{\max(A_L, A_R)}, \quad b^\flat = \sqrt{\max(B_L, B_R)}$	**12 opérations** **10 évaluations**
2. Calcul du $a^\sharp$ assurant $\tau^*_L > 0, \tau^*_R > 0$ (voir (2.41) $\tau_\kappa = 1/\rho_\kappa, \qquad v_\kappa = q_\kappa/\rho_\kappa, \qquad \kappa = L, R$ $\alpha = \min(\tau_L, \tau_R), \qquad \beta = v_L - v_R, \qquad \gamma = \lvert P_R - P_L \rvert$ $\Delta = \beta^2 + 8\alpha\gamma, \qquad a^\sharp = \max(\frac{\beta + \sqrt{\Delta}}{4\alpha}, 1)$	**12 opérations**
3. Calcul du $b^\sharp$ assurant $Y^* \in [0,1]$ (voir (2.41)) $(\rho\phi)_\kappa = \rho_\kappa \phi_\kappa, \qquad \kappa = L, R$ $b^\sharp = \max(\max((\rho\phi)_L, (\rho\phi)_R), 1)$	**4 opérations**
4. Calcul de (a, b) final (voir (2.44) et (2.45)) $a = \max(a^\flat, a^\sharp), \qquad b = \max(b^\flat, b^\sharp)$	**2 opérations**
Total	**30 opérations** **10 évaluations**

Fig. B.1. Calcul des coefficients de relaxation pour le schéma eulérien.

Contrairement au schéma eulérien qui évalue les flux aux interfaces entre les mailles une seule fois à chaque pas de temps, le schéma L-P nous oblige à calculer ces flux en deux étapes différentes : les flux pour la phase Lagrange et ceux pour la phase Projection. Les détails du calcul de flux des deux schémas sont montrés dans les figures (B.3) et (B.4). Le nombre d'opérations total consommées par le schéma eulérien est égal à 98 et celui du schéma L-P est plus petit : 62 opérations.

Notons que dans le cas du schéma L-P l'évaluation de la solution intermédiaire après la phase Lagrange est nécessaire pour la mise à jour des flux de l'étape Projection, même si ces flux correspondent à la formulation conservative, *i.e.* la solution finale est calculée à partir de la solution initiale à l'instant n et pas à partir de la solution intermédiaire. Il faut remarquer aussi que les deux schémas réalisent le même nombre d'évaluations de fonctions de bases pour leur calcul de (a, b) et de flux (10 évaluations).

Les deux méthodes utilisent des algorithmes de calcul des éléments propres dont la complexité est la même (en $O(N)$). En théorie, on devrait avoir un rapport de coût entre le schéma L-P et eulérien explicite à l'ordre de $98/60 \approx 1.63$. En combinant avec le nombre d'évaluations (rapport 1), le gain du schéma L-P par rapport au schéma eulérien en explicite d'ordre 1 en temps et en espace est

$$Gain = \zeta + 1.63(1 - \zeta), \qquad \zeta \in [0, 1]. \tag{B.1}$$

Étape	**Nombre d'opérations**
1. Calcul de $(a^\flat, b^\flat)$ de base $A_\kappa = -(\partial_\tau P)_\kappa + (\partial_v P)^2_\kappa, \qquad B_\kappa = (\partial_Y \sigma)^2_\kappa, \qquad \kappa = L, R$ $a^\flat = \sqrt{\max(A_L, A_R)}, \quad b^\flat = \sqrt{\max(B_L, B_R)}$	**12 opérations** **10 évaluations**
Total	**12 opérations** **10 évaluations**

Fig. B.2. Calcul des coefficients de relaxation pour le schéma Lagrange-Projection.

Étape	**Nombre d'opérations**
1. Calcul de (a, b)	**30 opérations** **10 évaluations**
2. Calcul des états intermédiaires dans le problème de Riemann (voir (2.60)) $\tau_\kappa = 1/\rho_\kappa, \quad Y_\kappa = \chi_\kappa/\rho_\kappa, \quad v_\kappa = q_\kappa/\rho_\kappa, \quad \kappa = L, R$ $Y^* = \frac{1}{2}\left[Y_L + Y_R + \frac{1}{b}(\Sigma_R - \Sigma_L)\right], \quad v^* = \frac{1}{2}\left[v_R + v_L - \frac{1}{a}(\Pi_R - \Pi_L)\right]$ $\Pi^* = \frac{1}{2}\left[\Pi_R + \Pi_L - a(v_R - v_L)\right], \quad \Sigma^* = \frac{1}{2}\left[\Sigma_L + \Sigma_R + b(Y_R - Y_L)\right]$ $\tau_L^* = \tau_L + \frac{\Pi_L - \Pi^*}{a^2}, \quad \tau_R^* = \tau_R + \frac{\Pi_R - \Pi^*}{a^2}$	**38 opérations**
3. Calcul des valeurs propres (voir (2.23)) $\omega = (v_L - a\tau_L, v^* - b\tau_L^*, v^*, v^* + b\tau_R^*, v_R + a\tau_R)$	**8 opérations**
4. Solution du problème de Riemann $\rho = 1/\tau_\mathfrak{R}, \quad \chi = \rho Y_\mathfrak{R}, \quad q = \rho v_\mathfrak{R}, \quad (\rho\Pi) = \rho\Pi_\mathfrak{R}, \quad (\rho\Sigma) = \rho\Sigma_\mathfrak{R}$ avec $(\tau, Y, v, \Pi, \Sigma)_\mathfrak{R}$ la solution du problème de Riemann.	**5 opérations**
5. Évaluation du flux numérique (voir (2.36)) $v = q/\rho, \quad \Pi = (\rho\Pi)/\rho, \quad \Sigma = (\rho\Sigma)/\rho$ $\mathbb{G} = (q, \chi v - \Sigma, \rho v^2 + \Pi, q\Pi + a^2 v, q\Sigma - b^2\chi/\rho)^T$	**17 opérations**
Total	**98 opérations** **10 évaluations**

Fig. B.3. Calcul du flux pour le schéma eulérien explicite.

Étape	**Nombre d'opérations**
1. Calcul de (a, b)	**12 opérations** **10 évaluations**
2. Calcul des états intermédiaires dans le problème de Riemann $$v^* = \frac{1}{2}\left[v_R + v_L - \frac{\Pi_R - \Pi_L}{a}\right]$$ $$\Pi^* = \frac{1}{2}\left[\Pi_R + \Pi_L - a(v_R - v_L)\right], \quad \Sigma^* = \frac{1}{2}\left[\Sigma_L + \Sigma_R + b(Y_R - Y_L)\right]$$ $$\rho_L^* = \frac{\rho_L a^2}{a^2 + \rho_L(\Pi_L - \Pi^*)}, \quad \rho_R^* = \frac{\rho_R a^2}{a^2 + \rho_R(\Pi_R - \Pi^*)}$$	**26 opérations**
3. Calcul des valeurs propres $$\omega = (v_L - a/\rho_L, v^*, v_R + a/\rho_R)$$	**4 opérations**
4. Flux numérique Lagrange (voir (3.23)) $$\mathbb{F}^{Lagrange} = (v^*, -\Sigma^*, \Pi^*)^T$$	**0 opérations**
5. Évaluation de la solution lagrangienne $$\mu_i = \frac{\Delta t}{\rho_i \Delta x_i}$$ $$\widetilde{\tau_i} = \tau_i + \mu_i(v^*_{i+1/2} - v^*_{i-1/2}), \qquad \widetilde{\rho_i} = 1/\widetilde{\tau_i}$$ $$\widetilde{Y_i} = Y_i + \mu_i(\Sigma^*_{i+1/2} - \Sigma^*_{i-1/2}), \qquad (\widetilde{\rho Y})_i = \widetilde{\rho_i}\widetilde{Y_i}$$ $$\widetilde{v_i} = v_i - \mu_i(\Pi^*_{i+1/2} - \Pi^*_{i-1/2}), \qquad (\widetilde{\rho v})_i = \widetilde{\rho_i}\widetilde{v_i}$$	**12 opérations**
6. Flux numérique Projection (voir (3.34)) $$(\mathbb{F}_{\mathbb{X}})_{i+1/2} = (v^*_{i+1/2})^+(\widetilde{\rho\mathbb{X}})_i + (v^*_{i+1/2})^-(\widetilde{\rho\mathbb{X}})_{i+1} + \mathbb{H}^*_{i+1/2}$$ avec $\mathbb{X} = (1, Y, v)^T$ et $\mathbb{H} = (0, -\Sigma, \Pi)^T$	**6 opérations**
Total	**60 opérations** **10 évaluations**

Fig. B.4. Calcul du flux pour le schéma Lagrange-Projection explicite.

Étape	**Nombre d'opérations**
1. Étape physique (voir (2.52))	**113N opérations**
1.1 Calcul des flux numériques	98N opérations
$$\mathbb{G}_{i+1/2} = \mathbb{G}(\mathbb{W}_i, \mathbb{W}_{i+1})$$	10N **évaluations**
1.2 Mise à jour de $\mathbb{W}_i^\sharp$	15N opérations
$$\mu_i = \frac{\Delta t}{\Delta x_i}$$ $$\mathbb{W}_i^\sharp = \mathbb{W}_i - \mu_i \left(\mathbb{G}_{i+1/2} - \mathbb{G}_{i-1/2}\right)$$	
2. Étape mathématique (voir (2.53))	**404N opérations**
2.1 Calculs préliminaires	2N opérations
$$\mu_i = \Delta t/(2\Delta x_i), \quad (\mathbb{I})_{ij}\delta_{ij}$$ où $\delta_{ij} = 1$ si $i = j$ et $\delta_{ij} = 0$ sinon	
2.2 Construction du premier membre	399N opérations
(a) Calcul des matrices Jacobiennes modifiées	106N opérations
$$\nabla\mathcal{G}_\kappa = \nabla\mathcal{G}\left(\mathbb{W}_\kappa\right), \quad \kappa = L, R$$	
(b) Calcul de la matrice de diffusion (voir (**??**))	143N opérations
$$\left\|\widetilde{\mathcal{A}}^{Roe}\right\| = \left\|\widetilde{\mathcal{A}}^{Roe}(\mathbb{W}_L, \mathbb{W}_R)\right\|$$	
(c) Calcul des dérivées (voir (2.74))	
$$(\partial_\rho P)_i, (\partial_{\rho Y} P)_i \text{ et } (\partial_{\rho v} P)_i$$	3N évaluations
(d) Calcul des matrices $\mathbf{C}_i, \mathbf{D}_i, \mathbf{E}_i$ (voir (2.50b) et (**??**))	150N opérations
$$\mathbf{C}_i = -\mu_i \left(\nabla\mathcal{G}_L + \left\|\widetilde{\mathcal{A}}^{Roe}\right\|\right), \quad \mathbf{E}_{i+1} = \mu_i \left(\nabla\mathcal{G}_R - \left\|\widetilde{\mathcal{A}}^{Roe}\right\|\right)$$ $$\mathbf{D}_i = \mathbb{I} - \mathbf{C}_i - \mathbf{E}_i$$	
2.3 Construction du seconde membre (voir (2.53))	5N opérations
$$\mathcal{R}_i = \mathbb{W}_i^\sharp - \mathbb{W}_i$$	
3. Résolution du système linéaire	**384N opérations**
Total	**901N opérations** **13N évaluations**

Fig. B.5. Détails du schéma eulérien semi-implicite (blocs de taille 5, demi largeur bande 9).

Étape	**Nombre d'opérations**
1. Étape préliminaire : approximation de Δt	**$128N$ opérations**
1.1 Calcul de (a,b) + états intermédiaires (Riemann) + valeurs propres	$40N$ opérations **$10N$ évaluations**
1.2 Calcul des résidus (voir (3.138)) $\alpha_i = 1/\rho_i \Delta x_i \quad$ et $\quad \beta_i = \theta\alpha_i/2$ $(\mathcal{R}_\tau^{(1)})_i = \alpha_i(v^*_{i+1/2} - v^*_{i-1/2}), \quad (\mathcal{R}_v^{(1)})_i = -\alpha_i(\Pi^*_{i+1/2} - \Pi^*_{i-1/2})$ $(\mathcal{R}_Y)_i = \alpha_i(\Sigma^*_{i+1/2} - \Sigma^*_{i-1/2})$ $(\mathcal{R}_\tau^{(2)})_i = \beta_i \dfrac{(\partial_Y P)_{i+1}(\mathcal{R}_Y)_{i+1} - 2(\partial_Y P)_i(\mathcal{R}_Y)_i + (\partial_Y P)_{i-1}(\mathcal{R}_Y)_{i-1}}{a}$ $(\mathcal{R}_v^{(2)})_i = \beta_i\left((\partial_Y P)_{i+1}(\mathcal{R}_Y)_{i+1} - (\partial_Y P)_{i-1}(\mathcal{R}_Y)_{i-1}\right)$	$22N$ opérations N évaluations
1.3 Calculs intermédiaires pour $p = 1, 2$ (voir (3.159)) $\max_i \left\lvert(\mathcal{R}_{C^+}^{(p)})_i\right\rvert \quad$ et $\quad \max_i \left\lvert(\mathcal{R}_{C^-}^{(p)})_i\right\rvert$	$16N$ opérations
1.4 Calcul des bornes $M^1_{i+1/2}$ et $M^2_{i+1/2}$ (voir (3.169) et (3.170))	$50N$ opérations
2. Étape Lagrange : résolution du système linéaire (voir (3.5.4))	**$96N$ opérations**
2.1 Construction de matrice par blocs (sur une ligne) : $\gamma_i = \mu_i/2$ $\left[\begin{array}{cc\|cc\|cc} \gamma_i\dfrac{(\partial_\tau P)_{i-1}}{a} & \gamma_i & 1-2\gamma_i\dfrac{(\partial_\tau P)_i}{a} & 0 & \gamma_i\dfrac{(\partial_\tau P)_{i+1}}{a} & -\gamma_i \\ -\gamma_i(\partial_\tau P)_{i-1} & -\gamma_i a & 0 & 1+2a\gamma_i & \gamma_i(\partial_\tau P)_{i+1} & -a\gamma_i \end{array}\right]$	$20N$ opérations
2.2 Construction de seconde membre (voir (3.138)) $(\widetilde{\mathcal{R}}_\tau)_i = \Delta t(\mathcal{R}_\tau^{(1)})_i - (\Delta t)^2(\mathcal{R}_\tau^{(2)})_i$ $(\widetilde{\mathcal{R}}_v)_i = \Delta t(\mathcal{R}_v^{(1)})_i - (\Delta t)^2(\mathcal{R}_v^{(2)})_i$	$8N$ opérations
2.3 Résolution du système linéaires d'incréments sur τ et v	$60N$ opérations
2.4 Mise à jour de $\widetilde{\mathbb{U}}$ $\widetilde{\rho}_i = \dfrac{1}{1/\rho_i + \delta\tau_i}, \quad \widetilde{Y_i} = Y_i + \Delta t(\mathcal{R}_Y)_i, \quad \widetilde{v_i} = v_i + \delta v$ $(\widetilde{\rho Y})_i = \widetilde{\rho}_i\widetilde{Y}_i \quad$ et $\quad (\widetilde{\rho v})_i = \widetilde{\rho}_i\widetilde{v}_i$	$8N$ opérations
3. Étape Projection : mise à jour du flux numérique $(\mathbb{F}_\mathbb{X})_{i+1/2} = (v^*_{i+1/2})^+(\widetilde{\rho\mathbb{X}})_i + (v^*_{i+1/2})^-(\widetilde{\rho\mathbb{X}})_{i+1} + \mathbb{H}^*_{i+1/2}$ avec $\mathbb{X} = (1, Y, v)^T$ et $\mathbb{H} = (0, -\Sigma, \Pi)^T$	**$6N$ opérations**
Total	**$230N$ opérations** **$11N$ évaluations**

Fig. B.6. Détails du schéma Lagrange-Projection semi-implicite (blocs de taille 2, demi largeur bande 5).

B.3 Analyse algorithmique pour les schémas semi-implicites

Dans cette section, nous nous intéressons à la comparaison des complexités algorithmiques du schéma semi-implicite d'ordre 1 en temps et en espace, sans terme source ni conditions aux limites. Rappelons que pour le schéma Lagrange-Projection, nous avons 2 étapes de résolution. La première nécessite de résoudre un système linéaire par blocs dont le temps de calcul n'est pas négligeable. Néanmoins, on constate que, dans l'étape Lagrange, l'opération qui consomme le plus de temps de calcul est l'approximation a priori du pas de temps. Cette méthode d'approximation est assez compliquée avec une complexité en $O(N)$. Elle nous permet de choisir un pas de temps convenable à notre schéma L-P semi-implicite. Notons qu'il est très difficile, voire impossible de prédire la grandeur de Δt pour le schéma L-P semi-implicite car d'une part on utilise une méthode d'approximation tout à fait différente de celle du schéma eulérien et d'autre part on utilise un coefficient de relaxation uniforme (global) sur tout le domaine de calcul. Les expériences nous ont démontré la validité et l'efficacité de notre méthode d'approximation.

En ce qui concerne le système linéaire, nous rappelons que la résolution d'un système linéaire dont la matrice à gauche ayant la largeur de bande l nécessite $(4l^2 + 6l + 6)N$ opérations en utilisant l'algorithme de Gauss. Le schéma eulérien nécessitera donc $384N$ opérations (figure (B.6)) pour la résolution du système linéaire par blocs de taille 5×5 tandis que le schéma L-P aura besoin juste de $60N$ opérations pour cette étape. Le schéma eulérien semi-implicite utilise $901N$ opérations et $13N$ opération. En revanche, le schéma L-P semi-implicite n'utilise que $230N$ opérations et $11N$ évaluations. Le gain en nombre d'opérations entre les deux schémas est égal à $901/230 \approx 3.92$ tandis que le gain en nombre d'évaluations est égal à $13/11 \approx 1.18$.

En théorie, le facteur global entre les deux méthodes semi-implicite d'ordre 1 en temps et en espace est une combinaison entre le coût de l'évaluation des fonctions de bases et le nombre d'opérations :

$$Gain = 1.18\zeta + 3.92(1 - \zeta), \qquad \zeta \in [0, 1]. \tag{B.2}$$

Notons qu'il est encore possible de diminuer le coût de calcul si on néglige les termes $(\partial_Y P)_i$: il y aura N évaluations (étape 1.2) et 14 opérations (étape 1.2 et 2.2) de moins. Ceci est faisable en vérifiant les résultats obtenus après l'élimination de ces termes.

B.4 Conclusion

Dans ce chapitre, nous avons étudié et comparé les schémas L-P et eulérien du point de vue de la complexité algorithmique et du temps CPU. Nos remarques sont les suivantes :

1. En explicite, le temps CPU est un peu plus important avec le schéma eulérien qu'avec le schéma L-P.

2. En semi-implicite, le schéma eulérien est plus gourmand en nombre d'opérations à cause de son gros système linéaire. Le schéma L-P utilise une estimation du pas de temps optimal, laquelle conduit parfois à un pas de temps plus petit que celui obtenu avec le schéma eulérien. Dans la plupart des cas, les deux schémas donnent des pas de temps Δt comparables et par conséquent le schéma Lagrange-Projection est plus rapide que le schéma eulérien en semi-implicite.

3. Le schéma L-P semi-implicite nous permet d'obtenir en pratique un gain entre 1.5 et 1.6 en temps CPU pour la simulation des cas-tests réels (voir les chapitres précédents).

Ces résultats sur les cas-tests réalistes (avec les conditions aux limites et source et glissement) nous confirment l'amélioration en temps CPU que l'on désirait.

Annexe C

Structure du code informatique

Ce chapitre est consacré à la description de la structure du code développé au cours de la thèse. Son objectif étant de documenter tous les futurs développeurs qui prendront le relais sur ce sujet, il nous a paru judicieux de le rédiger en anglais.

Le plan du chapitre est le suivant. Tout d'abord, nous faisons une brève introduction au programme implémenté, suivie par une présentation des classes principales. Ensuite, nous détaillons de manière précise les différents types de classes : la représentation de la solution, le modèle, les lois de fermetures et les schémas. La dernière partie consiste à montrer quelques diagrammes de collaboration des classes afin de mieux voir les relations entre elles.

C.1 Introduction

The code that we implemented is called AMRelax code. It takes in consideration not only the model and the physical settings of the problem but also the numerical schemes using the relaxation method and the adaptive methods (in space and time). The philosophy of the AMRelax code that we want to emphasize here consists of separating the mesh refinement technique and the numerical method. Indeed, the mesh refinement criterion is based solely on the smoothness of the solution, and it does not matter which numerical method was used to obtain this solution. Inversely, given a mesh (uniform or adaptive), we can use any suitable numerical method in order to advance the solution on the next time step. In the current version of the code, there exist two types of schemes : an Euler scheme and a Lagrange-Projection scheme, both of which use the relaxation method. However, one can employ a different method such as the VFRoe currently used in TACITE.

The AMRelax code was implemented in the oriented object C++ programming language which seems to suit best our needs. In this context, the idea of separating the numerical scheme and the grid management is obvious. When we deal with the mesh refinement we can see only the solution representation, *i.e.* the interface, and we cannot see how it was computed, *i*.e. the implementation. Inversely, when we deal with the solution update, we do not care how the current mesh was obtained, only the values of the solution can be seen on the given mesh.

The outline of this chapter is as follows. First, we will give a quick start guidelines of the main classes in order to have an overview of the AMRelax code. Then the rest of the chapter will describe the class documentation.

C.2 Quick start

The code main file Main.cpp creates the instances of the following classes :

Parameter Reads the parameters from an external or data file (*.dat).

Thermo Defines a particular pressure law or equation of state (compressible or uncompressible), specified by Parameter.

Hydro Defines a particular hydrodynamic law or slip law, specified by Parameter.

Model Defines a particular model depending on the corresponding Parameter entry. In our work, we implement the Drift-Flux model with pipeline inclination $\mathcal{T}$ and the source terms $\mathcal{S}$. Model defines also the properties of the relaxation method, *i*.e. how to set the state in the equilibrium and how to compute the relaxation coefficients. It has access to the instances of Parameter : Thermo and Hydro, which are passed as pointer arguments in its constructor.

The derived class LPRelaxModel and RelaxModel correspond respectively to the models used by the Euler relaxation scheme and the Lagrange-Projection scheme. In these classes, we compute the physical values at cell interfaces while using the finite volume method. In LPRelaxModel, we implement methods to compute global relaxation coefficients. Additional model (potential model for example) can be defined as a derived class of Model.

BC Defines a particular type of boundary conditions. The BC constructor takes a pointer to Model as parameter. This is needed to compute the ghost cells.

The derived classes BCNeumann, BCRelax and BCRelaxLP inherit from BC. The first one is used for simulations with shock tubes, the second and the third ones are used for real two-phase flow simulations.

Solution Defines a solution representation on either Uniform or Adaptive grids. Independently of the choice of grid, the access to the values of the solution in grid cells is the same: it is given by the overload operator [ ], defined differently in Uniform and Adaptive. This allows to use the same numerical scheme Scheme for all types of grid. The Solution's constructor receives a pointer from BC as a parameter in order to compute the interface values at the boundary and adapts the grid near this place.

The classes Uniforme1, Uniforme2, Adaptive1, Adaptive2, Adapt1LTS, Adapt1LTSVar and Adapt2LTS inherit from Uniform and Adaptive. They define a first or second order representation of the solution by evaluating its left and right values at each cell interface. These values are used in Scheme to compute the numerical flux function.

Scheme Initializes the internal representation of the solution, *i.e.* the Riemann test-cases and the computation of the stationary solution with boundary conditions problems.

The inherited classes RelaxScheme and LPRelaxScheme define the procedure to compute the solution up to the final time of simulation. In each of these classes, this procedure consists of computing the flux function and updating the solution after each time step. The details of the corresponding methods are re-defined in their derived classes where we distinguish several modus operandi: RelaxExplicit, RelaxSemiImplicit for the relaxation scheme and LPExplicit,LPImplicit,LPLTSExp and LPLTSImp for the Lagrange-Projection scheme.

In the next section, we will follow closely the above order in the documentation of the AMRelax code.

C.3 Documentation of the code

C.3.1 Pointwise solution representation

The value of the solution at each position x and time t is represented in one form of the physical *state*, *i.e.* its density, velocity, pressure, pipeline inclination and various derivatives. In order to represent the principal physical values of the solution, we make use of the Vector representation. To reflect the variety of the pointwise solution representation, we introduce a hierarchy of classes, describing different aspects of the physical state.

C.3.1.1 Vector, Vector2 and VectorAMR

The basic class of the solution representation is Vector whose internal representation is valarray<double>. In the general case, Vector has the dimension DIM= 5 and its components are addressed via the overload subscript operator [ ].

We can be interested in using two-component vectors created by Vector2 while working with the Larange-projection scheme. The usual componentwise operations (addition, subtraction ...) are also defined for these two classes. We note that Vector does not have any information on the meaning of its components.

The class VectorAMR inherits from Vector and adds to this some components which will be used to estimate the norm of the corresponding Detail in the multiresolution representation of the solution. For the moment, only the pipeline inclination $\mathcal{T}$ is added. It is subject to arithmetical operations as the other components of Vector.

C.3.1.2 Matrix and Matrix2

The class Matrix is used for the implementation of the implicit version of the Euler relaxation method. A Matrix is internally represented by a linear valarray[DIM×DIM]. Due to the low-level valarray operations like slice, the code allows a better compiler optimization.

Matrix2 is a class used for the semi-implicit version of the Lagrange-Projection scheme. It uses the Vector2 as well as the corresponding operations.

C.3.1.3 State, StateCons, StatePrim and StatePrimReconstr

The State is derived from VectorAMR. The subscript operator [ ] is inherited from Vector so it still addresses the DIM= 5 components. In addition to these Vector components, State possesses various other variables such as the position x, the associated cell size Δx and other variables related to the pressure and slip laws.

The three classes StateCons, StatePrim and StatePrimRecons are derived from State. These classes give in fact the physical meanings to the Vector components:

StateCons	Interprets its components as conservative variables $(\rho, \rho Y, \rho v, \rho\Pi, \rho\Sigma)^T$.
StatePrim	Interprets its components as primitive variables $(\rho, Y, v, \Pi, \Sigma)^T$.
StatePrimRecons	Interprets its components as primitive reconstruction ones $(p, Y, v, \Pi, \Sigma)^T$.

Depending on the procedure of the scheme, we can use one of the three derived classes. For example, in the L-P scheme, we use StatePrimRecons in the Lagrange step and StateCons in the Projection step for the reconstruction by MUSCL. Note that we can transform easily one class to another by using simple linear operations.

C.3.2 Physical closures

C.3.2.1 Thermo

The class Thermo and its derived classes ThermoCompressible and ThermoIncompressible define the state laws when using compressible or incompressible liquid. They include the computation of the pressure as well as its derivatives (function of the mixture density and the gas mass fraction). The two methods of these clases are:

Pressure()	Calculates the pressure in fun ction of the density and the gas mass fraction $p = p(\rho, Y)$.
Density()	Calculates the density in function of the pressure and the gas mass fraction $\rho = \rho(p, Y)$.

To compute these values, the methods of Thermo can take either StateCons, StatePrim or StatePrim-Recons as arguments.

C.3.2.2 Hydro

The computation of the slip law is one of the major difficulties while working with the Drift-Flux model. As mentioned in the technical section, it consists of computing the difference of phase velocities. In this work, we take in consideration two types of slip laws:

Simplified	Defines *simple* a slip law. Its expression of the slip is explicitly known.
Synthetic	The so-called slip law cannot be written in a close-form algebraic expression. Rather, it can be represented as a sophisticated if-then-else operator, which switches between the different flow regimes. Since there are three distinct flow regimes (stratified, intermittent and slug), the difference of phase velocities will be calculated in a different way, depending on a particular regime. Inside these regimes, an approximate Newton's method is used in order to solve a corresponding system of nonlinear equations, yielding the difference of phase velocities and its derivatives with respect to the input variables.

To work with each of these laws, we define two derived classes from Hydro:

HydroSimplified	Calls the corresponding Fortran procedures of the code of M. Baudin.
HydroSynthetic	Calls the code provided by the department of Mechanics.

In these classes, the method SlipLaw(State) (defined virtual in Hydro) updates the fields of a State related to the difference of phase veolicities and its derivatives.

C.3.3 Mathematical model

C.3.3.1 Model

The class Model describes the properties related to the system of equations of the Drift-Flux model. In particular, it provides two important methods:

Theta(x)	Returns the pipeline inclination $\mathcal{T}$ at the position x.
Source($\mathcal{T}$)	Returns the value of the source term for a given State.
Equilibrium(S)	Sets a State S to equilibrium and add to this some additional fields related to the slip law

In order to compute the pressure and slip laws, the pointers to the instances of Thermo and Hydro are passed to the Model's constructor.

Fig. C.1. Inheritance diagram for the class Model.

C.3.3.2 RelaxModel and LPRelaxModel

The class LPRelaxModel is derived from Model and is used for Lagrange-Projection scheme. The class RelaxModel is used for the Euler relaxation scheme. It is itself derived from LPRelaxModel due to the common use of BCRelaxLP (see section C.3.4.2). In each of these classes, there are two main methods:

Flux()	This method permits compute the flux function at a cell interface.
CoefRelax()	This method defines how to compute the relaxation coefficients for the Riemann problem when giving two states to the left and right of an interface. In the case of Lagrange-Projection scheme, the conditions on a and b to ensure the positivity of the density and the maximum principal of the gas mass fraction are dismissed.

In the LPRelaxModel we introduce the UpdateGlobalAB() method to calculate the global (maximum) relaxation coefficient a and b. Note that all of these methods are implemented in two versions: one using StateCons and one using StatePrimReconstr.

C.3.4 Boundary conditions

A correct implementation of the boundary conditions plays a crucial role as they mainly govern the flow behavior. In the current version, we use only one ghost cell at each boundary. In order to implement second order methods, also one can set at the boundaries more than one ghost cell by incrementing the parameter Nghost in Parameter.

C.3.4.1 BC

The class BC defines the interface for the implementation of the boundary conditions. Its main methods are:

UpdatePhysicalBC()	Updates the boundary values of the total inflow q_t, the gas inflow q_g and the outflow pressure P in function of time t.
DeltaBC()	Computes the variation of q_t (total mass flow rate), q_g (gas mass flow rate) at the input and the pressure P at the output between two instants.
ComputeGhostCellsInflow() ComputeGhostCellOutflow()	Virtual methods defined in the derived classes, depending on the type of the chosen boundary conditions.

Note that BC's constructor receives a pointer to the Model which means that its methods have access to the properties of the Drift-Flux model, such as the setting to equilibrium of a State.

Fig. C.2. Inheritance diagram for the class BC.

C.3.4.2 BCNeumann , BCRelax and BCRelaxLP

The class BCNeumann derived from BC is used for Riemann problems. It consists of setting each ghost cell equal exactly to the adjacent internal one.

The class BCRelax and BCRelaxLP are also two derived classes of BC. They are implemented so that they satisfy the *kill-wave* method as explained in the technical part. Initially, BCRelax is used for the Euler relaxation scheme but due to the different versions of killing waves at the boundaries, experiences show that it's better to use BCRelaxLP devoted for the LP scheme. Note that the BCRelax and BCRelaxLP constructors receive also a pointer to the Model instance. However, their methods are designed for the relaxation model RelaxModel and LPRelaxModel respectively. In consequence, their constructors must convert the Model pointer into the corresponding one by using the dynamic_cast operator.

C.3.5 Grid solution representation

C.3.5.1 Solution

The class Solution defines the representation of the solution on a grid. Its core is valarray<StateCons> U, which holds the internal representation of the solution in conservative variables. Solution is a pure abstract class, all of its methods are virtual and are defined in derived classes. Among these classes, we can find the important methods:

Adapt()	Adapts the underlying grid.
InterfaceStates()	Computes the states at the left and right of each cell interface.

Note that the Solution receives a pointer to BC as the methods of BC are used to compute the ghost cells, which are in turn used to compute the interface states at the boundary. Moreover, in the case of adaptive grid the time varying boundary conditions may require some preventive refinement of the grid near the boundary.

Fig. C.3. Inheritance diagram for the class Solution.

C.3.5.2 Uniform

The class Uniform is derived from Solution. It describes the access to the values of the solution on the uniform underlying grid. The subscript operator [ ] is defined accordingly: it returns the corresponding value of the internal representation U, computed with respect to the number of ghost cells Nghost. Consequently, the method Adapt() is empty. We note that the choice of the boundary conditions is independent of the grid type.

C.3.5.3 Uniform1 and Uniform2

There are two classes derived from Uniform:

Uniform1 — Defines the cell interface values with the corresponding piecewise constant values of the solution to the left and to the right of this interface on the uniform grid. It corresponds to the first order numerical scheme.

Uniform2 — Ddefines a piecewise linear representation of the solution on the uniform grid. It corresponds to the second order numerical scheme. The interface values in the domain are obtained via the MUSCL technique using the methods SlopeLimiter() and NewSlopeLimiter(). On the contrary, the interface values on the boundaries are computed differently for each time step: the previous ghost cell values are used to obtain the internal interface boundary values which are then used to compute external interface boundary values as well as new ghost cell values.

Each of these classes defines their own versions of the interface state computation thanks to the method InterfaceStates().

C.3.5.4 Adaptive

The class Adaptive derived from Solution provides the access to the values of the solution on the adaptive grid via the overloaded operator [ ]. It adapts the underlying grid by providing an implementation of the method Adapt() whose algorithm is the following:

PredictValue() — Predict the value $\widehat{\mathbb{U}}^k_{2i}$.

EncodeActive() — Obtain values at active cells. A cell is active if we are about to reconstruct the value of the solution there. The indicator Cell.Active() shows if the corresponding cell is active or not. Initially, all cells are set to be active. Therefore, when called for the first time, EncodeActive() provides a total encoding of the solution.

Threshold() — Mark the cells with details greater than the threshold parameter (see....) and predicts the set of significant details at next time step. This is done by Cell.Marked().

ActivateCells() — Activate cells around the marked ones so that the tree is graded (see...), *i.e.* set the corresponding Cell.Active() using....

DecodeActive() — Obtain values in the active cells.

CoverageActive() — Construct the adaptive grid using.... This construction consists of updating the array adaptive[j], which returns the index in the internal representation of the solution U, given the index j of a cell on adaptive grid.

In fact the adaptive mesh, provided by Adapt(), uses the multiresolution representation of the solution, given by the class Multiresolution and related classes. Inside of Adaptive, the access to the

elements of the internal representation of the solution is done via the instances u of the class Value and d of the classe Detail. For a cell i on a grid of level k, $u(i,k)$ (resp. $d(i,k)$) returns a reference to the corresponding value of the solution (resp. detail).

C.3.5.5 Adaptive1 and Adaptive2

Like the uniform case, there are two classes derived from Adaptative in order to work with schemes with different order:

Adaptive1	Defines a piecewise constant representation of the solution at the interfaces between cells over the adaptive grid. It corresponds to the first order scheme.
Adaptive2	Defines a piecewise linear representation of the solution and corresponds to the second order scheme when using adaptive mesh refinement.

C.3.5.6 Adapt1LTS and Adapt1LTSVar

Similarily to Adaptive1, the class Adapt1LTS and Adapt1LTSVar are derived from Adaptive but they are used for the Local Time Stepping application coupling with the Multiresolution strategy:

Adapt1LTS	Devoted to working with a constant micro time step along one macro time step.
Adapt1LTSVar	Permits to work with the variable micro time step strategy along the simulation.

Both of these two classes have their own InterfaceStates() method, the implementation of which is a little bit complicated as we have to pay attention to the indexes of cells, specially in the transition zones.

As the Local Time Stepping method needs to update the solution only on part of the grid after each intermediate time step, we have to localize the indexes of the cells where the solution can be changed. Consequently, we have to define two following methods in theses classes:

GetIndexFP()	Provides the indexes of fluxes nearby the discretized solution that will be updated. The parameter NbFluxes represents the number of flux functions to be calculated at each intermediate time step.
PartialUpdate()	Computes these solutions in consideration.

C.3.6 Scheme

Scheme defines the class of numerical schemes. Its principal methods are:

Init()	Computes the initial solution. It calls either the InitRiemann() method for Riemann problems or the InitStationary() method for problems with boundary conditions. This procedure must be done at the beginning of the simulation without knowing the specific numerical scheme we want to work with.
Solve()	Pure abstract function which will be defined in derived classe. It designs the solver containing the loop in time.
UpdateFluxes() UpdateSolution()	Pure abstract functions defined in the derived classes. Their role is to update the flux at interfaces cells then update the solution after each time step.

Note that to compute the solution Scheme needs to access to all properties of Thermo, Hydro, Model and BC. Consequently, Scheme possesses four protected instances of these corresponding classes, which are passed by pointer in the constructor of Scheme.

C.3.6.1 LPScheme and RelaxScheme

The LPScheme and RelaxScheme are derived from Scheme. Each one correponds to a particular scheme: Lagrange-Projection and pure Euler relaxation:

RelaxScheme	Represents the Euler relaxation scheme using the Relaxmodel.
	Solve() is implemented here while UpdateFluxes() and UpdateSolution() methods are defined in the derived classes (RelaxExplicit and RelaxSemiImplicit).
LPScheme	Represents the Lagrange-Projection scheme using the LPRelaxmodel.
	The Solve() method corresponding to the L-P scheme is also defined.

In order to distinguish two trategies of calculating the time step, we introduce two classes derived from LPScheme:

LPGTS	Used for global time stepping, *i.e* we do call the Local Time Stepping strategy.
LPLTS	Used for local time stepping, *i.e* we employ the Local Time Stepping strategy.

In this case, the loop in time management Solve() will call the corresponding method SolveTS() which computes the solution in one global time step (for uniform and MR case) and in one macro time step (LTS case). Obviously, SolveTS() is defined differently in each of these two classes.

C.3.6.2 LPExplicit, LPImpBC, LPLTSExp, LPLTSExpVar, LPLTSImp

At the lowest level in the hierarchy of LPScheme, we can find many classes used for different type of Lagrange-Projection schemes. The three folling classes are derived from LPGTS and contain methods computing the flux functions UpdateFluxesLagrange() and UpdateFluxesprojection():

LPExplicit	Explicite L-P scheme.
LPImplicit	L-P semi-implicite scheme used for Riemann problems.
LPImpBC	L-P semi-implicite scheme use for test-cases with boundary conditions.

In parallel, the are two other classes derived from LPLTS when we work with the Local Time Stepping strategy:

LPLTSExp	Explicite L-P scheme with constant micro time step.
LPLTSExpVar	Explicite L-P scheme with variable micro time step.
LPLTSImp	Semi-implicite L-P scheme with constant micro time step.

Both of these two clases contain methods computing the flux functions, *i.e* UpdateFluxesLagrangeLTS() and UpdateFluxesProjectionLTS().

C.3.6.3 RelaxExplicit and RelaxSemiImplicit

The class RelaxExplicit corresponds to the explicit Euler relaxation scheme and RelaxSemiImplicit corresponds to the implicit version. They are derived directly from RelaxScheme as they do not use the local time stepping strategy.

Fig. C.4. Inheritance diagram for the class Scheme.

C.3.7 Some collaboration diagrams of the main classes

In this part, we show some collaboration diagrams of some main classes: RelaxScheme and LPScheme (Fig. C.5, Uniform and Adaptive (Fig. C.6).

Fig. C.5. Collaboration diagram for RelaxScheme and LPScheme classes.

In these diagrams, the relation with a full line arrows represents the inheritance between two classes, *i.e.* $A \rightarrow B$ means that the class A is derived from the class B. While the dotted line arrows represent the link connecting the associated objects of the classes in cosideration. It defines an interaction between these objects. Concretely, here the relation $A \overset{x}{\dashrightarrow} B$ means that an object x of class B is passed by a pointer in the constructor of the class A.

We can see that the scheme (LPScheme or RelaxScheme) and the representation type of the solution (Uniform or Adaptive) can be implemented independently. That is to say we can apply the multiresolution method to any numerical scheme. The scheme does not see the properties of the mesh and reversely the construction of the adaptive mesh does not depend on the chosen scheme. We say that Adaptive is a « on-layer » of the Scheme which means that there is no dependence between these two classes. This makes the code AMRelax very flexible in the way that we can add to it any numerical scheme without modifying the adaptive's code. The new scheme can use either uniform or adaptive

Fig. C.6. Collaboration diagram for Uniform and Adaptive classes.

mesh with no difficulty.

Annexe D

Articles soumis

Nous joignons dans cette annexe deux articles soumis en 2007 et à apparaître. Le premier est soumis au journal SIAM et le deuxième est soumis au journal Mathematics of Computation.