



RAPPORT DE STAGE

Développement de méthodes de haute résolution avec contrôle d'erreur
pour la simulation des EDP évolutive multi-échelles en temps et en espace

Étudiant : Alexandre EDELINE
École : ENSTA Paris - Institut Polytechnique de Paris
Période : du 14/04/2025 au 17/10/2025

Laboratoire : CMAP - École Polytechnique
Maîtres de stages : Marc MASSOT et Christian TENAUD
Tuteur académique : Patrick CIARLET

3 octobre 2025

Remerciements

Je remercie chaleureusement Marc Massot pour sa bienveillance, sa disponibilité et plus généralement pour m'avoir accompagné au cours de ce stage. Je tiens à faire particulièrement l'éloge de ses capacités de superviseur : guider la compréhension, encourager l'initiative, orienter la recherche et les expériences, aiguiller la rédaction ; un authentique mentor.

J'exprime également toute ma gratitude à Christian Tenaud avec qui les échanges ont beaucoup oeuvré à la rigueur et la clarification de mes travaux.

Merci à toute l'équipe du HPC@Math qui m'a beaucoup soutenu et dont le travail a beaucoup facilité mes recherches. Je suis sincèrement reconnaissant à Pierre, Laurent et Loïc pour la "*hotline samurai*" et plus généralement pour leur aide efficace, patiente et bienveillante qui m'a permis de surpasser des problèmes informatiques parfois idiots, parfois insidieux et souvent obscurs... Je souligne au passage l'effort de l'équipe Samurai pour intégrer les fonctionnalités demandées en un temps record!

Je remercie Richard et Ward pour leur aimable relecture et leurs conseils. Je remercie Antoine pour m'avoir aidé sur un *bug* qui me semblait insurmontable. Je salue tous les doctorant du CMAP, les remerciant pour les débats passionnants (parfois absurdes) et je remercie tous les chercheurs, toujours disponible pour expliquer et vulgariser leurs recherches.

Je remercie Théophane, pour son soutiens et nos échanges. Je remercie enfin mes parents dont le soutien m'a toujours permis de me concentrer exclusivement et sans inquiétude sur mes études.

Abstracts

Résumé en Français

Mots-clés : Schémas Numériques, Simulation des EDP d'Évolution, Multirésolution Adaptative, Méthodes ImEx, Advection-Diffusion-Réaction, Analyse d'erreur numérique, Analyse de stabilité.

Les systèmes couplant mécanique des fluides et chimie complexe se modélisent par les équations d'advection-diffusion-réaction (ADR), une classe d'équations aux dérivées partielles dont la résolution numérique requiert à la fois des stratégies d'adaptation de maillage (par exemple la multirésolution adaptative, MRA) et des méthodes d'intégration temporelle spécifiques (splitting, schémas ImEx). Ce travail étudie les interactions entre ces deux composantes (adaptation spatiale et intégration temporelle). Il apporte (i) une comparaison empirique de l'effet de la MRA sur les schémas de splitting et les schémas ImEx, ainsi qu'une analyse (ii) théorique et (iii) numérique des couplage émergeant entre une méthode de Runge-Kutta classique et la MRA sur un problème de diffusion traité par méthode des lignes.

English abstract

Keywords : Numerical Schemes, Evolution PDE Simulation, Adaptive Multiresolution, ImEx Methods, Advection-Diffusion-Reaction, Numerical Error Analysis, Stability Analysis.

Systems coupling fluid mechanics and complex chemistry are modeled by advection-diffusion-reaction (ADR) equations, a class of partial differential equations whose numerical resolution requires both spatial mesh adaptation strategies (such as adaptive multiresolution, MRA) and dedicated time integration methods (splitting, ImEx schemes). This work investigates the interactions between these two components (spatial adaptation and temporal integration). It provides (i) an empirical comparison of the effect of MRA on splitting and ImEx schemes, and (ii) a theoretical and (iii) numerical analysis of the coupling that emerges between a classical RungeKutta method and MRA on a diffusion problem solved with the method of lines.

Table des matières

Remerciements	2
Abstracts	3
Liste des figures	7
1 Introduction	8
1.1 Présentation du laboratoire	8
1.1.1 Le laboratoire.	8
1.1.2 L'équipe HPC@Math	8
1.2 Présentation du sujet	9
1.2.1 Les difficultés des équations d'advection-diffusion-réaction	9
1.2.2 Les stratégies de simulation des équations d'advection-diffusion-réaction	10
1.3 Problématique	13
1.4 Organisation du rapport	13
2 Préambule mathématique	14
2.1 Intégrations des EDOs	15
2.1.1 Schémas explicites et implcites.	15
2.1.2 Ordre de convergence d'un schéma	15
2.1.3 Stabilité et raideur	16
2.2 Simulation des EDPs d'évolution	18
2.2.1 Les méthodes des lignes.	18
2.2.2 Les volumes finis	19
2.2.3 Analyse de schéma numériques	20
2.3 Les équations d'advection-diffusion-réaction	23
2.3.1 Trois opérateurs au propriétés mathématiques très différentes	23
2.3.2 Difficultés mathématiques intrinsèques	24
2.3.3 Conclusion sur les équations d'ADR	25
2.4 La Multirésolution Adaptative	26
2.4.1 La transformée multi-échelle	26
2.4.2 L'adaptation	29
2.4.3 Algorithmes de simulation numérique	30
3 Contribution	31
3.1 Étude de méthodes ImEx sur une équation de diffusion-réaction	32
3.1.1 L'équation de Nagumo	33
3.1.2 Les méthodes ImEx	35

3.1.3	Analyse de stabilité	38
3.1.4	Étude de la convergence	43
3.1.5	Mise en lumière expérimental de couplages entre la méthode en temps et l'adaptation spatiale	44
3.1.6	Conclusion	45
3.2	Obtention de l'équation équivalente d'une méthode de lignes avec multirésolution adaptative sur un problème de diffusion.	46
3.2.1	Cadre de l'étude	47
3.2.2	Les équations équivalentes	50
3.2.3	Complément expérimental	53
3.2.4	Conclusion	55
3.3	Impact de la qualité de reconstruction des flux pour les problèmes diffusion avec AMR.	56
3.3.1	Présentation des trois algorithmes	56
3.3.2	Expérience numérique avec une méthode Runge et Kutta explicite	57
3.3.3	Analyse de stabilité	59
3.3.4	Expérience numérique avec une méthode explicite stabilisée	60
3.3.5	Lien avec les équations équivalentes	63
3.3.6	Extension sur une équation de diffusion-réaction	66
4	Conclusion	68
	Bibliographie	71

Table des figures

2.1	Illustration du comportement attendu de l'erreur d'un schéma d'ordre deux dont le seuil d'instabilité est $\Delta t > 10^{-1}$.	17
2.2	Exemple de maillage adapté par multirésolution adaptative grâce au logiciel Samurai.	26
2.3	Exemple de grille dyadique	27
3.1	Profils des ondes solutions de l'équation de Nagumo pour différents ratios k/D avec le produit $kD = 1$ fixé (c'est à dire à vitesse fixée). L'augmentation du ratio k/D accentue le gradient spatial.	33
3.2	Plage de valeurs du terme de réaction non-linéaire et de sa différentielle pour deux coefficients de réactions : $k = 1$ et $k = 10$.	34
3.3	Pour différents couples D et k , diagrammes de stabilité des méthodes ImEx et de référence sur l'équation de Nagumo.	40
3.4	Pour $k = 500$ et $D = 500$: diagrammes de stabilité des méthodes ImEx et de référence sur l'équation de Nagumo.	42
3.5	Comparaison de la convergence du schéma de splitting avec celle des méthodes ImEx222 et ImEx232 sur l'équation de Nagumo avec $D = 0.1$ et $k = 10$.	43
3.6	Convergence des schémas ImEx et de splitting, adaptés en espace par MRA, sur l'équation de Nagumo pour $k = 10$, $D = 0.1$. Les flux sont évalués au niveau courant (cf. 3.2), la prédiction/reconstruction est assurée par un prédicteur à trois points et l'erreur est comparé à une solution convergé en temps.	44
3.7	Illustration d'une simulation du cas test 3.44 avec conditions de Dirichlet homogène et affichage de l'erreur locale au temps final.	53
3.8	Saturation de la convergence temporelle avec une méthode RKE2 sur l'équation de diffusion. L'erreur L2 stagne malgré la diminution du pas de temps, illustrant la domination de l'erreur spatiale due à la contrainte CFL $\Delta t \propto \Delta x^2$.	54
3.9	Convergence temporelle d'ordre 2 avec une méthode SDIRK-RK2 sur l'équation de diffusion pour différentes profondeurs de maillages adaptés. L'ordre théorique est préservé comme attendu, puisque les flux sont évalués au niveau courants.	55
3.10	Illustration des trois algorithmes évalués. L'algorithme 1 en bleu calcul le flux à partir des cellule de la grille au même niveau que l'interface étudiée. L'algorithme 2 reconstruit les valeurs de la solution sur la grille de niveau inférieur. L'algorithme 3 reconstruit au niveau le plus fin possible. Plus l'algorithme reconstruit finement, plus les valeurs moyennes sont données sur des cellules petites et plus cela s'approche d'une valeur "ponctuelle".	57
3.11	Convergence pour les différentes méthodes numériques.	60

3.12 Profils d'erreur pour différentes valeurs de la constante CFL. C'est le pas d'espace est fixé, cela revient simplement à changer le pas de temps.	62
3.13 Régression entre l'erreur numérique expérimentale (AMR + reconstruction fine) et la dérivées 4 ^e de la solution.	64
3.14 Régression entre l'erreur numérique expérimentale (AMR + reconstruction fine) et une combinaison linéaire des dérivées 4 ^e et 6 ^e de la solution.	64
3.15 Courbes de convergence de chaque méthode d'AMR pour différents paramètres de l'équation. Plus k est élevé, plus le profil de l'onde est raide et plus la réaction domine. La célérité de l'onde est néanmoins identique pour chaque jeu de paramètres puisque le produit kD reste constant d'une expérience à l'autre.	67

Chapitre 1

Introduction

1.1 Présentation du laboratoire

1.1.1 Le laboratoire.

Le Centre de Mathématiques Appliquées de l'École Polytechnique¹ (CMAP) a été créé en 1974. Cette création répond au besoin émergent de mathématiques appliquées face au développement des méthodes de conception et de simulation par calcul numérique dans de nombreuses applications industrielles de l'époque (nucléaire, aéronautique, recherche pétrolière, spatial, automobile). Le laboratoire fut fondé grâce à l'impulsion de trois professeurs : Laurent SCHWARTZ, Jacques-Louis LIONS et Jacques NEVEU. Jean-Claude NÉDÉLEC en fut le premier directeur, et la première équipe de chercheurs associés comprenait P.A. RAVIART, P. CIARLET, R. GLOWINSKI, R. TEMAM, J.M. THOMAS et J.L. LIONS. Les premières recherches se concentraient principalement sur l'analyse numérique des équations aux dérivées partielles. Le CMAP s'est diversifié au fil des décennies, intégrant notamment les probabilités dès 1976, puis le traitement d'images dans les années 1990 et les mathématiques financières à partir de 1997.

Le laboratoire a formé plus de 230 docteurs depuis sa création et a donné naissance à plusieurs *startups* spécialisées dans les applications industrielles des mathématiques appliquées. Le CMAP comprend trois pôles de recherche : le pôle analyse, le pôle probabilités et le pôle décision et données. J'ai intégré l'équipe **HPC@Maths** du **pôle analyse**.

1.1.2 L'équipe HPC@Math

L'équipe HPC@Math² travaille à l'interface des mathématiques, de l'informatique et de la physique (mécanique des fluides, thermodynamique) pour développer des méthodes numériques efficaces pour la simulation des EDP. L'équipe s'intéresse entre-autre aux problèmes multi-échelles, au travers des équations d'advection-réaction-diffusion. Les travaux de l'équipe s'inscrivent toujours dans un contexte HPC (high performance computing). Ce terme désigne l'usage optimal des ressources informatiques disponibles et de leur architectures. Chaque méthode est développée en se demandant si elle pourra être facilement parallélisée, si elle pourra être déployée sur GPU ou sur un cluster de calcul de manière efficace.

1. <https://cmap.ip-paris.fr>

2. <https://initiative-hpc-maths.gitlab.labos.polytechnique.fr/site/index.html>

1.2 Présentation du sujet

Des moteurs aux batteries, des enjeux de défenses au nucléaire civil en passant par les problématiques de sécurité, de nombreux problèmes industriels font intervenir des systèmes physico-chimiques modélisés par un couplage entre la mécanique des fluides et des dynamiques chimiques complexes. Ces modèles décrivent par exemple les problèmes de combustion [24, 14], d'électrochimie [27], de corrosion [4] de propagation d'effluents [16]. Les équations émergeant de ces modèles sont les équations d'advection-diffusion-réaction (ADR), une gamme d'équations aux dérivées partielles (EDP) dont les trois opérateurs traduisent le couplage sus-mentionné. Elles comportent donc un opérateur d'advection (transport par un flux), un opérateur de diffusion (éparpillement de la matière par l'agitation thermique) et un opérateur de réaction (modélisant une ou plusieurs réactions chimiques).

Comme détaillé plus tard, ces équations sont difficile à simuler avec précision. L'objet du stage est donc de contribuer à l'élaboration de méthodes de résolution haute-résolution de ces EDPs.

Cette introduction présente d'abord les difficultés liées aux équations d'ADR, puis réalise un état de l'art des stratégies pour surmonter ces difficultés, par la suite vient la problématique à laquelle les contributions du stage répondent.

1.2.1 Les difficultés des équations d'advection-diffusion-réaction

Les équations d'ADR posent deux difficultés majeures aux numériciens :

- A. des opérateurs aux propriétés antagonistes
- B. des solutions multi-échelles

A Des opérateurs aux propriétés antagonistes

Le terme antagoniste signifie que les méthodes numériques efficaces pour un opérateur sont inadaptées aux autres. Plus de détails sont donnés en 2.3 mais rapidement :

- ◇ les *méthodes explicites*³ sont adaptés à l'opérateur d'advection, mais présentent des problèmes de *stabilité*⁴ pour les opérateurs de réaction et de diffusion.
- ◇ le caractère local et très *raide*⁵ de l'opérateur de réaction⁶ pousse à utiliser des *méthodes implicites*⁷, cependant le caractère non-local de la diffusion rend les coûts calculatoires prohibitifs.
- ◇ les méthodes explicites stabilisées comme [3] sont très adaptées à la résolution de l'opérateur de diffusion, mais ne sont assez stables pour l'opérateur de réaction.

Ainsi, une approche monolithique (qui traiterait tous les opérateurs d'un bloc) échouera systématiquement à simuler le système de manière satisfaisante.

3. Définition en 2.1

4. Définition de stabilité en 2.1

5. Définition d'opérateur raide en 2.1

6. Cette raideur vient des temps de relaxations très courts (quelques nanosecondes) des modèles de chimie complexe [34]

7. Définition en 2.1

B Des solutions multi-échelles

Les solutions des équations d'ADR sont souvent multi-échelles en temps et en espace [13] : pour obtenir une précision donnée, chaque zone spatio-temporelle requiert des niveaux de résolution différents. Par exemple pour un problème de combustion, tant que la combustion n'est pas initiée, un pas de temps et une résolution spatiale grossiers représentent fidèlement le système. En revanche après l'allumage, le pas de temps doit être réduit drastiquement pour rendre compte de la complexité des réactions de combustion déclenchées et, proche de la flamme, une résolution spatiale élevée est nécessaire pour rendre compte des structures complexes de la flamme, alors que loin de la flamme une résolution grossière reste suffisante. Ainsi soit la résolution reste faible mais la précision médiocre ; soit elle est élevée partout mais une grande partie du coût en calcul et en mémoire est injustifié, car une part significative de la simulation ne requiert pas une telle résolution.

1.2.2 Les stratégies de simulation des équations d'advection-diffusion-réaction

Pour parer ces difficultés trois familles de stratégies existent :

- A. Le découplage d'opérateurs, permet de développer des schémas non-monolithiques où chaque opérateur est traité indépendamment des autres selon ses propriétés.
- B. L'adaptation en espace, permet d'adapter localement la finesse du maillage, une résolution spatiale où la structure de la solution le demande (forts gradients, discontinuités...) et une résolution plus faible où la solution est lisse, simple, régulière.

En voici un bref état de l'art⁸.

A Les stratégies de découplage d'opérateurs

Deux approches de découplage d'opérateurs existent : la séparation (ou *splitting*) d'opérateurs et les méthodes implicites-explicites (ImEx). La frontière est poreuse entre ces deux concepts puisque certains schémas de *splitting* peuvent être vus comme des ImEx. Des détails mathématiques complémentaires sur les méthodes ImEx sont présentés en 3.1.2.

Le *splitting* repose sur un développement de Taylor de l'exponentielle de matrice. Il s'agit en pratique de simuler les opérateurs les uns après les autres de sorte que le résultat soit équivalent d'un schéma monolithique qu'à une erreur de *splitting* maîtrisée. Concrètement, la solution de l'EDP :

$$\begin{aligned} \text{Trouver } u \in \mathcal{F}(\mathbb{R}^+, U) \text{ tel que :} \\ \partial_t u = Au + Bu, \\ u(t = 0, \cdot) = u_0(\cdot). \end{aligned} \tag{1.1}$$

s'écrit comme :

$$u(t, \cdot) = e^{t(A+B)} u_0(\cdot). \tag{1.2}$$

8. Pour être exhaustif, il faudrait également évoquer les stratégies d'adaptations temporelles, où le pas de temps est choisi dynamiquement à partir de critères *a priori* ou *a posteriori*.

où $e^{(A+B)} : \mathbb{R}^+ \rightarrow U$ est le semi-groupe correspondant à l'EDP (si A et B sont des matrices, c'est simplement l'exponentielle de matrice). Le schéma de *splitting* de Lie, précis à l'ordre un repose sur le développement suivant :

$$e^{\Delta t(A+B)} u_0 = e^{\Delta t B} \circ e^{\Delta t A} u_0 + \underbrace{\mathcal{O}(\Delta t^2)}_{\text{Erreur de splitting}}. \quad (1.3)$$

En acceptant cette erreur de *splitting* local $\mathcal{O}(\Delta t^2)$, passer d'un l'état u_0 au temps t à un état u_1 au $t + \Delta t$, il suffit de simuler l'opérateur A à partir de u_0 pendant Δt pour obtenir \tilde{u}_0 , puis de simuler l'opérateur B à partir de \tilde{u}_0 pendant Δt pour obtenir u_1 . À aucun moment il n'est précisé les schémas utilisés pour simuler A et B , ils peuvent chacun être adapté à l'opérateur simulé (par une méthode implicite et l'autre explicite). Le *splitting* vient au prix d'une erreur global entachée d'une erreur supplémentaire (erreur *splitting*). L'avantage est la liberté totale concernant le choix des schémas simulant chaque opérateur. Un des schémas de *splitting* les plus utilisés est le schéma de Strang [32], il permet de découpler deux opérateurs avec une précision d'ordre deux en temps.

Les méthodes ImEx se distinguent du *splitting* classique en imposant des conditions entre les schémas utilisés pour chaque opérateur, tirant de cette contrainte une erreur de découplage plus faible. Les premières méthodes ImEx (hors *splitting*) virent le jour à la fin des années 1990 avec les méthodes Runge et Kutta additives (ARK) par Ascher *et al* [5], complétés dans les années 2000 par [22] puis par [28]. Ces méthodes reposent sur une combinaison de plusieurs méthodes RK (une par opérateur) préservant un ordre de convergence global. L'objectif est de dépasser le *splitting* posant de sérieux problèmes au delà l'ordre deux ou encore lorsque l'on cherche à découpler plus de deux opérateurs. Cela a pavé la voie à des méthodes ImEx plus complexes intégrant par exemples des méthodes stabilisées [1] ou encore à des méthodes ImEx couplées espace-temps, faites sur mesures pour une équation spécifique [30].

B Les stratégies d'adaptation en espace

L'objectif est ici d'utiliser le maillage avec le moins de points possibles tout en préservant une précision choisie.

B.i Les stratégies *features-based* et *goal-based* Plusieurs stratégies d'adaptation en espace existent et se déclinent en deux grandes catégories[33] :

1. Les approches *features-based* utilisent une quantité locale de la solution (par exemple la vitesse de l'écoulement) et ses gradients pour inférer localement le besoin de résolution spatial et adapter en conséquence.
2. Les approches *goal-based* utilisent une fonction de coût reflétant la qualité globale de la solution selon l'adaptation choisie. Généralement la fonction de coût est issue d'un problème adjoint sur une quantité d'intérêt.

B.ii La multi-résolution adaptative (MRA) Le stage se centre sur la *multi-résolution adaptative* [19] qui ne tombe dans aucune des deux catégories précédente. Puisqu'elle est fondée avant tout sur des arguments issus de la théorie de l'information. L'origine de la multirésolution adaptative remonte aux travaux d'Ami Harten [19] dans les années 1990 qui adapte une méthode de compression

de donnée par analyse multi-échelle aux algorithmes de simulation. Ces travaux furent par la suite prolongés entre autre dans [20, 9]. Cette technique permettant de réduire drastiquement les temps de simulation a été largement appliquée et est une approche très compétitive [12] pouvant tirer parti des infrastructures de calcul moderne (multi-coeur, GPU) [31].

Une présentation mathématique est proposée dans le préambule mathématique en 2.4. Succinctement, avec un schéma MRA, la solution est représentée sur plusieurs grilles de finesses différentes. L'adaptation (compression) consiste à ignorer l'information superflue contenue sur les grilles les plus fines là où la solution est assez régulière. À une erreur de compression et de reconstruction près, un niveau de résolution maximal (grille fine) peut être retrouvé grâce à une reconstruction par interpolation polynomiale (décompression).

1.3 Problématique

La multirésolution-adaptative comme stratégie d'adaptation spatiale Plusieurs schéma d'adaptation par MRA existent. *L'approche standard* est de faire évoluer la solution sur le maillage adapté (compressé) à partir des valeurs du maillage adapté. *Une approche non-standard* est de faire évoluer la solution sur le maillage adapté, mais à partir de valeurs reconstruites, décompressées, plus précises. **Révéler et comprendre les interactions entre l'adaptation spatiale par multirésolution-adaptatives et les schémas d'intégration temporelle utilisés pour simuler les équations d'advection-diffusion-réaction.** Pour aborder cette vaste problématique, mon travail fourni trois contributions ce focalisant chacune sur un point différent de la question :

1. Quelles sont les propriétés des approches ImEx par rapport au splitting et quelle stratégie est la plus impactée par l'adaptation spatiale?
2. D'où viennent les interactions entre l'adaptation spatiale et l'intégration en temps et comment limiter ce couplage néfaste?

1.4 Organisation du rapport

Le rapport est structuré comme suit.

- Ce chapitre pose le contexte, la problématique et les objectifs.
- Le Chapitre 2 sert de préambule mathématique et algorithmique : il rappelle les méthodes de discrétisation pour EDO et EDP, les équations d'advection-diffusion-réaction, quelques outils d'analyse de schémas numériques (stabilité, équations équivalentes), ainsi que les principes de la multi-résolution adaptative; le lecteur expérimenté peut le parcourir rapidement et s'y référer au besoin.
- Le Chapitre 3 rassemble trois contributions :
 - ◊ (3.1) : Une étude de la stabilité de deux méthodes ImEx ARK; suivie d'une comparaison avec un schéma de splitting de leur stabilité et convergence pour l'équation-test de Nagumo (réaction-diffusion).
 - ◊ (3.2) : Une étude théorique, via le calcul d'équations équivalentes, du comportement de l'erreur d'une schéma méthode des lignes pour un problème de diffusion. Cela dans trois contextes : (I) sans multi-résolution adaptative, (II) avec MRA standard, (III) avec MRA non-standard (reconstruction des flux).
 - ◊ (3.3) : Une étude expérimentale des différences entre les trois schéma de MRA mentionné précédemment permettant de mettre en relation les résultats théoriques de la contributions 3.2 avec les observations expérimentales.
- Le dernier chapitre (chapitre 4) conclut en trois volets : scientifique, technique et personnel.

Chapitre 2

Préambule mathématique

Ce préambule mathématique présente divers concepts innervant dans les travaux du stage. Le lecteur habitué peut ignorer ce chapitre et le consulter ponctuellement au besoin. Les sujets suivants y sont introduits :

- ◇ [2.1](#) Les méthodes de simulations des équations différentielles ordinaires (EDOs).
- ◇ [2.2](#) Les méthodes de simulations des équations aux dérivées partielles d'évolutions.
- ◇ [2.3](#) Les équations d'advection-réaction-diffusion et les difficultés numériques qu'elles posent.
- ◇ [2.4](#) La multi-résolution adaptative (MRA) comme méthode d'adaptation spatiale.

2.1 Intégrations des EDOs

Les techniques d'approximation d'EDPs d'évolution comportent souvent une étape nécessitant la résolution d'une équation différentielle ordinaire ¹ (EDO), c'est à dire une équation différentielle ne faisant intervenir qu'une seule variable différenciée, généralement le temps. Cette section rappelle quelques notions d'analyse et de simulation des EDOs du premier ordre.

Définition 2.1.1 (Équation différentielle ordinaire). Une équation différentielle ordinaire (du premier ordre) est une équation de la forme :

$$\begin{aligned} u' &= A(u, t) \quad u : t \in \mathbb{R}^+ \mapsto u(t) \in \mathbb{R}^d \\ u(0) &= u_0 \in \mathbb{R}^d. \end{aligned} \tag{2.1}$$

2.1.1 Schémas explicites et implicites.

L'approximation des EDO se fait grâce à des schémas numériques, c'est à dire une suite d'élément (u^n) de \mathbb{R}^d . Donnés une EDO et un pas de discrétisation temporel Δt , $u^n \in \mathbb{R}^d$ est approxime la solution de l'EDO au temps $t^n = n\Delta t$. C'est à dire que la suite $(u^n)_{n \in \mathbb{N}} \in (\mathbb{R}^d)^{\mathbb{N}}$ définie par un schéma numérique cherche à avoir à avoir $u^n \approx u(t = n\Delta t)$. Deux catégories de schémas numériques existent : les schémas explicites et les schémas implicites

Définition 2.1.2 (Schéma explicite). Un schéma numérique est dit explicite si la solution au pas de temps $n + 1$ est obtenu seulement grâce à la solution pas de temps n . Cela se formule usuellement sous la forme :

$$u^{n+1} = u^n + f(u^n, \Delta t). \tag{2.2}$$

Définition 2.1.3 (Schéma implicite). Un schéma numérique est dit implicite si la solution au pas de temps $n + 1$ est obtenue au moins en partie grâce à la solution pas de temps $n + 1$. Cela peut s'écrire écrit comme :

$$u^{n+1} = u^n + f(u^{n+1}, \Delta t). \tag{2.3}$$

Comparaison entre ces deux classes de schémas : Une itération d'un schéma implicite nécessite donc l'inversion d'un système linéaire ou non linéaire sur \mathbb{R}^d . De fait, une itération implicite est généralement plus coûteuse qu'une itération d'un schéma explicite ². Cependant pour des raisons de stabilités (voir 2.1.3) les méthodes explicites peuvent nécessiter des pas de temps bien plus fin, et donc bien plus d'itérations. Le choix entre méthode explicite et implicite dépend de bien des facteurs (du problème, du niveau de précision voulu, de la difficulté d'implémentation etc...) c'est un enjeu central de la simulation numérique.

2.1.2 Ordre de convergence d'un schéma

L'ordre de convergence permet de lier l'erreur des solutions numérique au pas de temps, c'est-à-dire de quantifier l'efficacité d'un schéma numérique. Pour définir la notion d'ordre de convergence

1. On utilisera aussi le terme *système dynamique*, même si en toute rigueur ce concept est un peu plus large.

2. En particulier si la dimension de la solution d est grande.

d'un schéma numérique, il faut d'abord définir son erreur.

Définition 2.1.4 (Erreur locale d'un schéma). L'erreur locale d'un schéma numérique de résolution d'une EDO est l'erreur que commet le schéma sur un pas de temps. Autrement dit si l'on note u la solution de l'EDO à partir d'un état initial $u_0 = u(t = 0)$ et u_1 l'approximation numérique proposée par la schéma pour un pas de temps Δt à partir de l'état u_0 , l'erreur locale est :

$$e(\Delta t) = \|u(\Delta t) - u_1\|_{\mathbb{R}^d}. \quad (2.4)$$

Définition 2.1.5 (Erreur globale d'un schéma). L'erreur globale d'un schéma est l'erreur que commet le schéma sur plusieurs pas de temps. Si l'on note u^n la solution numérique au temps $t^n = n\Delta t$, alors l'erreur globale du schéma jusqu'au temps final $T = N\Delta t$ peut être définie comme :

$$E(\Delta t) = \sum_{n=0}^N \|u^n - u(t^n)\|_{\mathbb{R}^d} \quad (2.5)$$

ou plus simplement encore :

$$E(\Delta t) = \|u^n - u(T)\|_{\mathbb{R}^d}. \quad (2.6)$$

Définition 2.1.6 (Ordre de convergence). Un schéma numérique de résolution d'une EDO est dit d'ordre p si de manière équivalente :

- ◊ L'erreur locale vérifie : $e(\Delta t) = O(\Delta t^{p+1})$
- ◊ L'erreur globale vérifie : $E(\Delta t) = O(\Delta t^p)$

2.1.3 Stabilité et raideur

Un schéma numérique d'ordre p converge asymptotiquement en $O(\Delta t^p)$ vers la solution exacte d'une EDO lorsque Δt est assez petit. Cependant, cette convergence n'est effective que si le schéma est **stable**. Un schéma instable conduit à une divergence de la solution numérique : en pratique, au-delà d'un pas de temps critique Δt_0 , la norme $\|u^n\|$ croît sans borne³, comme illustré figure 2.1. Ainsi, pour entrer dans le régime asymptotique de convergence, il faut respecter une contrainte de stabilité de type $\Delta t \leq \Delta t_0$. Lorsque ce seuil est très faible, la simulation devient coûteuse car elle nécessite un grand nombre d'itérations $T_{\text{final}}/\Delta t$. De manière générale, les schémas explicites sont plus sensibles à ces contraintes que les schémas implicites.

Définition 2.1.7 (Stabilité d'un schéma numérique). Un schéma numérique $(u^n)_{n \in \mathbb{N}} \in (\mathbb{R}^d)^{\mathbb{N}}$ est dit stable si la norme de la solution ne croît pas d'un pas de temps à l'autre :

$$\|u^{n+1}\| \leq \|u^n\|. \quad (2.7)$$

En pratique, cette condition est satisfaite lorsque Δt reste inférieur à un seuil Δt_0 dépendant du problème et du schéma.

La stabilité d'un schéma dépend directement du spectre de l'opérateur A dans l'équation différentielle (cf. (2.1)). Lorsque A possède des valeurs propres à grande partie réelle négative, les méthodes

3. Phénomène souvent appelé « explosion numérique ».

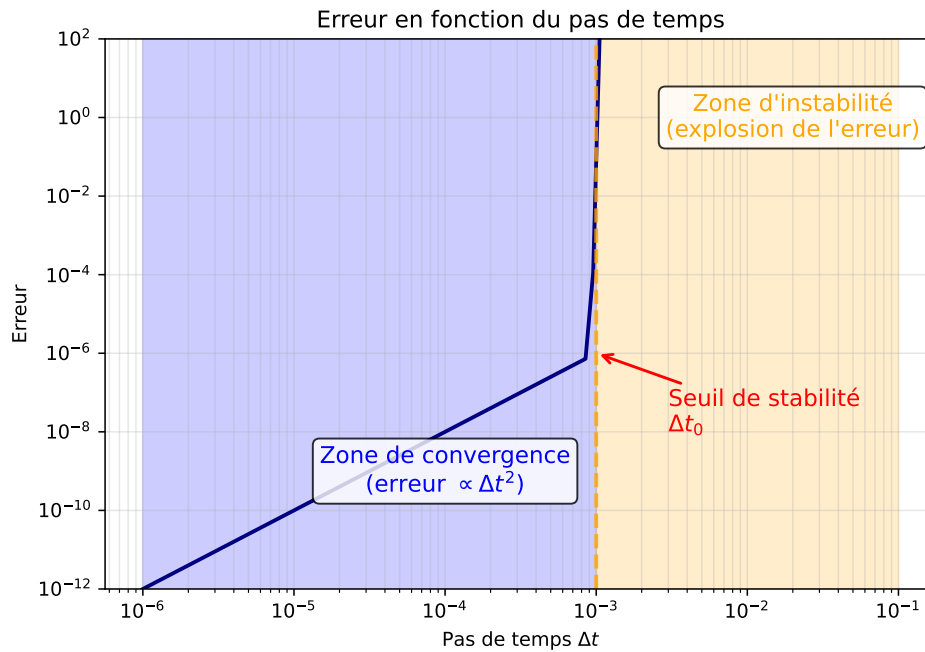


FIGURE 2.1 – Illustration du comportement attendu de l'erreur d'un schéma d'ordre deux dont le seuil d'instabilité est $\Delta t > 10^{-1}$.

explicites imposent des pas Δt extrêmement petits pour rester stables. Dans ce cas, le problème est qualifié de *raide*.

Définition 2.1.8 (Problème raide). Un système dynamique

$$\frac{du}{dt} = A(u, t), \quad u(t) \in \mathbb{R}^d \quad (2.8)$$

est dit *raide* si la jacobienne J_A admet des valeurs propres très négatives en valeur absolue, entraînant une condition de stabilité tellement restrictive que les méthodes explicites deviennent inutilisables en pratique.

Pour analyser la stabilité d'un schéma, on utilise la notion de *fonction de stabilité*, introduite à partir de l'équation test linéaire $u' = \lambda u$.

Définition 2.1.9 (Fonction de stabilité). Pour un schéma numérique appliqué à l'équation $u' = \lambda u$, on définit l'indice spectral $z = \lambda \Delta t$ et une fonction $S(z)$ telle que

$$U^{n+1} = S(z) U^n.$$

Le schéma est stable pour un pas Δt donné si et seulement si

$$|S(z)| \leq 1. \quad (2.9)$$

Exemple 2.1.10 (Équation de Dahlquist). Considérons

$$u'(t) = -\lambda u(t), \quad \lambda > 0, \quad (2.10)$$

$$u(0) = u_0, \quad (2.11)$$

dont la solution exacte est $u(t) = u_0 e^{-\lambda t}$.

Raideur. Lorsque λ est grand, la décroissance est très rapide. Un schéma explicite impose alors un pas Δt extrêmement petit pour rester stable : le problème est dit raide car les explicites échouent par instabilité.

Fonction de stabilité. On pose $z = \lambda \Delta t$ et on calcule $S(z)$:

- *Euler explicite* : $U^{n+1} = (1 - z)U^n \Rightarrow S(z) = 1 - z$. Stabilité $\iff 0 \leq z \leq 2$, soit $\Delta t \leq 2/\lambda$. Pour $\lambda \gg 1$, la contrainte est prohibitive.
- *Euler implicite* : $U^{n+1} = \frac{1}{1+z}U^n \Rightarrow S(z) = \frac{1}{1+z}$. Ici $|S(z)| \leq 1$ pour tout $z \geq 0$: aucune restriction sévère, même pour λ très grand.

Cet exemple illustre la notion de raideur : un problème est dit raide lorsque les méthodes explicites deviennent inutilisables à cause de la contrainte de stabilité, alors qu'un implicite reste stable.

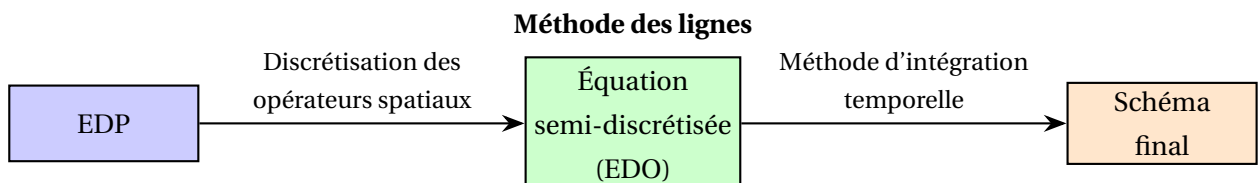
Pour aller plus loin, on peut introduire des notions plus fines (A-stabilité, L-stabilité), développées par exemple dans [18].

2.2 Simulation des EDPs d'évolution

Les équations aux dérivées partielles d'évolutions sont des EDPs dont une des variables différenciée est le temps. Cette section introduit divers éléments d'analyse et simulation pour ces équations. Elle introduit d'abord la notion de méthode des lignes, une approche classique d'élaboration de schéma numériques pour les EDPs d'évolutions. Puis elle présente la notion de volume fini qui est le paradigme de discrétisation spatiale utilisé dans ce stage. Enfin elle détaille quelques outils d'analyse des schémas numériques pour les EDPs d'évolution (convergence, stabilité, analyse d'erreur).

2.2.1 Les méthodes des lignes.

Définition 2.2.1 (Méthode des lignes). Une méthode des lignes est une famille de méthodes numériques pour approximer les EDP d'évolutions. Elle consiste à discrétiser les opérateurs spatiaux de l'équation afin d'obtenir une équation semi-discrétisée en espace, puis à utiliser une technique d'intégration en temps, pour obtenir la discrétisation complète de l'équation.



Il existe également des approches plus sophistiquées, comme les méthodes couplées espace-temps [11], mais ce stage se concentre sur les méthodes des lignes.

2.2.2 Les volumes finis

La technique de discrétisation spatiale utilisée dans ce travail est celle des *volumes finis* [26]. Cette approche discrétise la valeur moyenne sur les cellules du maillage, là où les approches par *différences finies* [25] discrétisent la valeur au noeuds du maillage et celles par *éléments finis* [8] discrétisent l'espace fonctionnel lui même.

Définition 2.2.2 (Volumes finis). Donné une discrétisation d'un domaine Ω par un maillage en cellules $(C_j)_{j \in J}$ dont le volume de l'ordre Δx^d (ou d est la dimension), la discrétisation par volume fini approxime les quantités :

$$U_j = \frac{1}{|C_j|} \int_{C_j} u(x) d\Omega. \quad (2.12)$$

Les volumes finis brillent lors de la simulation les lois de conservations, c'est à dire les EDPs de la forme :

$$\partial_t u = \operatorname{div}(f(u)). \quad (2.13)$$

En effet lorsque cette relation est moyennée sur une cellule C_j du maillage, cela donne :

$$\int_{C_j} \partial_t u = \int_{C_j} \operatorname{div}(f(u)), \quad (2.14)$$

$$\partial_t U_j = \int_{\partial C_j} f(u). \quad (2.15)$$

La notions de flux est centrale dans l'approche par volume finie et très importante pour comprendre le travail réalisée en 3.2.

Définition 2.2.3 (Flux physique). Dans l'équation 2.15, le terme $\int_{\partial C_j} f(u)$ est appelé Φ_j le terme de *flux* de la cellule j ; physiquement il quantifie "l'entrée de $f(u)$ " au sein de la cellule C_j . En une dimension $\Phi_j = f(u_j^+) - f(u_j^-)$; le flux dépend simplement des valeurs à l'interface de la cellule.

La définition précédente est une intégrale de bord faisant intervenir les valeurs exactes de u le long de l'interface. Malheureusement, paradigme des volumes n'offre qu'un accès aux valeurs moyennes de u sur chaque cellule. Ainsi, il faut approximer Φ_j à partir des valeurs moyennes.

Définition 2.2.4 (Flux numérique). Un *flux numérique* Ψ est fonction permettant d'approximer Φ_j à partir des valeurs moyennes sur la cellule j et les cellules voisines : $\Psi(U_{j-s}, \dots, U_j, \dots, U_{j+s})$. Le nombre de cellules intervenant dans le calcul s est appelé le *stencil*.

Définition 2.2.5 (Flux numérique d'ordre p). Pour être exploitable, un flux numérique doit être *consistant* à un ordre $p \geq 1$ avec la loi de conservation étudiée. Donné une solution u de régularité C^{p+1} un flux numérique est dit consistant à l'ordre p si, $\forall j$:

$$|\Psi(\tilde{U}_{j-s}(t), \dots, \tilde{U}_j(t), \dots, \tilde{U}_{j+s}(t)) - \Phi_j| = O(\Delta x^p). \quad (2.16)$$

Où \tilde{U}_j désigne la moyenne exacte (et non une approximation) de u sur la cellule C_j .

D'un point de vue méthode des lignes, le paradigme des volumes finis donne l'équation semi-discrétisée en espace suivante :

$$\partial_t U_j(t) = \Psi(U_{j-s}(t), \dots, U_j(t), \dots, U_{j+s}(t)), \quad (2.17)$$

il ne reste qu'à l'intégrer grâce à une méthode d'intégration des EDOs.

2.2.3 Analyse de schéma numériques

Pour qualifier une schéma numérique, sur différents sujets comme la faisabilité de sa mise en oeuvre, la qualité de la solution qu'il fournit ou encore la dynamique de l'erreur qu'il introduit, divers concepts ont été développés. Ce qui suit introduit les notions pertinentes pour comprendre les travaux du stage :

- ◇ A L'analyse de stabilité.
- ◇ B.i La quantification de l'erreur grâce à l'étude de la convergence.
- ◇ B.ii La qualification de la dynamique de l'erreur grâce à la notion d'équation équivalente.

A Stabilité d'un schéma numérique

Les schémas simulant les équations aux dérivées partielles ont les mêmes nécessités de stabilité, que ceux simulant les EDOs. D'ailleurs les méthodes des lignes transforment précisément une EDP en EDO. Pour éviter les redondances, le lecteur se référera à la partie 2.1. La seule différence entre la stabilité des schémas pour EDP et pour EDO est que, lorsque qu'une EDP est résolue numériquement, la raideur de l'opérateur spatial discret dépend généralement explicitement du pas d'espace Δx . De fait condition de stabilité ne pas réellement EDP mais plutôt de l'EDO résultant de la discrétisation en espace. Cela mène à des conditions du type : $\Delta t \leq \lambda(\Delta x)$. La constante λ est appelée la constante de stabilité, ou de manière parfois incorrecte la constante de Courant-Friedrich-Levy (CFL).

B Analyse de l'erreur

L'analyse de l'ordre de convergence d'un schéma permet de quantifier l'erreur asymptotique (c'est à dire lorsque les pas de temps et d'espace sont "assez petits") que commet le schéma.

B.i Ordre de convergence d'un schéma Les schémas numériques sont désignés par $(U_j^n)_{n,j}$ où n représente le pas de temps et j la cellule. Ainsi, si la grille contient c cellules, pour tout n , U^n est un élément de l'espace vectoriel \mathbb{R}^c . Le vecteur $(u(x_j, t^n))_{j \in \{1, \dots, c\}}$ est noté $u(\cdot, t^n)$.

Définition 2.2.6 (Erreur globale). L'erreur globale E_G d'un schéma $(U_j^n)_{n,j}$ sur un temps $T_f = N_f \Delta t$ peut être définie comme l'erreur au temps final :

$$E_G = \Delta x^d \|U^{N_f} - u(\cdot, T_f)\| \quad (2.18)$$

où $\|\cdot\|$ désigne une norme sur \mathbb{R}^c . Ou bien comme l'intégrale de l'erreur sur tous les pas de temps :

$$E_G = \sum_{k=0}^{N_f} \Delta x^d \|U^k - u(\cdot, t^k)\| \quad (2.19)$$

Définition 2.2.7 (Ordre de convergence d'un schéma). Un schéma $(U_j^n)_{n,j}$ est dit convergent à l'ordre p en temps et q en espace si son erreur de globale s'écrit : $E_G = O(\Delta t^p + \Delta x^q)$.

B.ii Equation équivalente

Définition 2.2.8 (Équation équivalente). L'équation équivalente d'un schéma est l'EDP dont la solution satisfait le schéma. Elle est calculée par des développements de Taylor en temps et en espace. Comparer l'équation équivalente et l'équation cible fait alors naturellement apparaître les termes d'erreur et leur dynamique.

Ce qui suit décrit une méthode générique pour obtenir l'équation équivalente d'un schéma, la notion d'équation équivalente est clé dans la contribution en 3.2.

Première étape : développement de Taylor L'existence d'une fonction assez régulière vérifiant le schéma est supposée. Dans le schéma numérique les termes $u_{k+\delta_x}^{n+\delta_t}$ sont remplacés par leur pendant continu : $u(x + \delta_x \Delta x, t + \delta_t \Delta t)$. Le développement de Taylor suivant est alors réalisé :

$$\begin{aligned} u(x + \delta_x \Delta x, t + \delta_t \Delta t) &= u(x, t) + \sum_{i=1}^{\infty} \frac{(\delta_x \Delta x)^i}{i!} \frac{\partial^i u}{\partial x^i}(x, t) \\ &+ \sum_{j=1}^{\infty} \frac{(\delta_t \Delta t)^j}{j!} \frac{\partial^j u}{\partial t^j}(x, t) \\ &+ \sum_{i,j \geq 1} \frac{(\delta_x \Delta x)^i (\delta_t \Delta t)^j}{i! \cdot j!} \frac{\partial^{i+j} u}{\partial x^i \partial t^j}(x, t) \end{aligned} \quad (2.20)$$

L'équation aux dérivées partielles qui apparaît est l'équation équivalente. Elle peut être tronquée à l'ordre voulu.

Deuxième étape : procédure de Cauchy-Kovaleskaya (optionnelle) Afin d'enrichir l'analyse, et permettre le développement de schémas couplés espace-temps, l'étape précédente peut être suivie d'une procédure de Cauchy-Kovaleskaya. La procédure de Cauchy-Kovaleskaya consiste à utiliser la relation entre les dérivées en espace et en temps données par l'équation cible et remplacer les dérivées en temps par des dérivées en espace dans l'équation équivalente. Par exemple, pour un schéma ayant pour équation cible la diffusion $\frac{\partial u}{\partial t} = D \frac{\partial^2 u}{\partial x^2}$, cela consiste à écrire de manière itérée :

$$\begin{aligned} \frac{\partial^2 u}{\partial t^2} &= D^2 \frac{\partial^4 u}{\partial x^4} \\ \frac{\partial^3 u}{\partial t^3} &= D^3 \frac{\partial^6 u}{\partial x^6} \\ &\vdots \\ \frac{\partial^n u}{\partial t^n} &= D^n \frac{\partial^{2n} u}{\partial x^{2n}} \end{aligned} \quad (2.21)$$

Dernière étape : relation entre le pas de temps et d'espace (optionnelle) Lorsque le schéma est utilisé en pratique, il est courant d'imposer une relation entre les pas d'espace et de temps, par exemple une condition de stabilité du type $\Delta t \propto \Delta x$ ou $\Delta t \propto \Delta x^2$. Il est donc utile d'injecter cette

relation dans l'équation équivalente pour comprendre le comportement du schéma en conditions réelles.

Ces outils d'analyse numérique constituent les fondements nécessaires pour aborder la simulation numérique des EDPs. La section suivante présente la multi-résolution adaptative qui permet de réduire le coût en calcul et en mémoire de ces méthodes.

2.3 Les équations d'advection-diffusion-réaction

Le contexte physique naturel des équations d'advection, diffusion, réaction est le suivant : des particules sont placées dans un milieu fluide où elles **diffusent**. Ce milieu fluide est en mouvement, il déplace les particules, les **advecte**. Enfin les particules **réagissent** entre elles et ces réactions modifient les grandeurs thermodynamiques (température, pression) et *in fine* les propriétés du milieu fluide. Les équations d'advection, diffusion, réaction modélisent le couplage trois phénomènes.

Cette partie présente les propriétés trois opérateurs des équations d'ADR (2.3.1) et les difficultés liées à leur résolution numérique (2.3.2).

2.3.1 Trois opérateurs aux propriétés mathématiques très différentes

A Advection

L'advection désigne le transport d'une quantité par un flot. L'opérateur d'advection le plus simple est l'opérateur de transport $c \frac{\partial}{\partial x}$:

$$\frac{\partial u}{\partial t} = c \frac{\partial u}{\partial x} \quad (2.22)$$

De manière générale l'opérateur d'advection d'une quantité u par un flot \underline{a} s'écrit $\underline{a} \cdot \nabla u$. Par exemple dans les équations de Navier-Stokes, le terme $\underline{v} \cdot \nabla \underline{v}$ modélise la vitesse \underline{v} transportée par elle-même. Une version simplifiée de ce phénomène est l'équation bien connue de Burgers.

Le spectre des opérateurs d'advections sont généralement à valeurs propres imaginaires [25, Chap. 10]. Ainsi ils sont peu raides mais résonnants. Les méthodes explicites sont souvent les plus adaptées pour les traiter [25, Chap. 10].

B Diffusion

La diffusion désigne l'*éparpillement* de particules au sein d'un milieu fluide⁴. Ce phénomène est la limite macroscopique du déplacement microscopique des particules dû à l'agitation thermique. L'opérateur de diffusion le plus classique est celui de l'équation de la chaleur :

$$\frac{\partial u}{\partial t} = D \Delta u. \quad (2.23)$$

Le spectre de cet opérateur est \mathbb{R}^- [25, Chap. 9], il est donc infiniment raide. L'opérateur discret correspondant ne capte qu'une partie de ce spectre. En pratique la raideur de l'opérateur discret augmente de manière quadratique avec la finesse de la discrétisation spatiale [25, Chap. 9].

Cet opérateur moyennement raide (comparé aux opérateurs de réaction). Ainsi on pourrait penser qu'une méthode implicite est adéquate. Cependant ce n'est généralement pas le cas. En effet le coefficient de diffusion n'est souvent pas uniforme ni isotrope. La structure de la matrice de l'opérateur discrétisé évolue donc et à chaque itération il faut inverser un nouvel opérateur implicite. Cette tâche est rendue plus ardue encore car l'opérateur de diffusion couple tout l'espace, c'est-à-dire qu'il

4. En théorie de l'information cela décrit la tendance de l'entropie augmenter et l'information à se moyenniser, se flouter.

est non local.⁵, il faut inverser une matrice de taille $d \gg 1$ dont la structure peut être très hétérogène (car le coefficient de diffusion varie dans l'espace). Aujourd'hui il semble plus pertinent d'utiliser des méthodes explicites stabilisées qui peuvent gérer sa raideur relative comme les méthodes ROK2 et ROK4[2]. Sans entrer dans les détails, la clé de ces méthodes consiste à prendre une méthode numérique explicite standard et lui ajouter des étages de sorte à ce que la fonction d'amplification résultante soit la fonction d'amplification de la méthode standard multipliée par un polynôme minimal sur \mathbb{R}^- comme un polynôme de Tchebichev.

C Réaction

Les phénomènes de réaction sont en général bien adaptés aux méthodes implicites car ils sont locaux et extrêmement raides. En effet, les temps typiques d'une réaction chimique⁶ sont de l'ordre de la nano-seconde [34]. De fait, les réactions chimiques sont très difficiles à simuler par des méthodes explicites. En revanche les méthodes implicites sont très efficaces dans ce contexte. En effet l'inversion de l'opérateur implicite peut être décomposé en plusieurs résolutions de petits systèmes ce qui est moins coûteux et parallélisable. Cela est permis grâce à la localité des réactions chimiques (à chaque pas de temps les particules au sein d'une cellule ne réagissent qu'avec les autres particules de la même cellule). En pratique cela signifie qu'il est possible de mettre en oeuvre une petite méthode implicite par cellule plutôt qu'une gargantuesque méthode globale; ce qui revient à inverser un opérateur de petite dimension en chaque cellule et non inverser un énorme système.

2.3.2 Difficultés mathématiques intrinsèques

La simulations des équations d'advection-réaction-diffusion se heurte à deux difficultés majeures, **le couplage des trois opérateurs** mentionnés précédemment et **le caractère multi-échelles des solutions**.

A Première difficulté : le couplage des opérateurs

Le développement précédent montre que résoudre chaque phénomène individuellement, n'est pas insurmontable. Toutefois, les résoudre tous en même temps, c'est-à-dire les coupler, est en pratique très difficile. En effet, lorsque ces trois opérateurs sont couplés, il en résulte un unique opérateur qui doit être traité par une méthode numérique. C'est là que surgissent les difficultés : si la méthode est explicite (éventuellement stabilisée), la raideur de la réaction impose des pas de temps extrêmement restrictifs, à l'inverse si la méthode est implicite, la non-localité de la diffusion demande l'inversion d'un système de taille déraisonnable. Cette approche naïve, monolithique, n'est donc pas adaptée. Il faut par conséquent, trouver d'autres stratégies.

5. Si l'opérateur de diffusion eût été local il aurait pu être inversé en résolvant plusieurs petits systèmes et ce, potentiellement en parallèle. Cela serait bien moins coûteux qu'inverser un grand système. À titre d'exemple, inverser une matrice pleine de taille 10^6 demande environ 10^{18} opérations, alors qu'inverser 100 systèmes de taille 10^4 n'en demande $100 \times 10^{12} = 10^{14}$ soit dix mille fois moins. Si ces résolutions sont parfaitement parallélisés, alors l'accélération théorique serait d'un million. Malheureusement comme l'opérateur de diffusion couple tout l'espace ce n'est pas possible en dimension deux et trois.

6. En réalité une réaction chimique aussi simple en apparence qu'une combustion H_2/O_2 fait intervenir une dizaine de composés et réactions intermédiaires [10], dont les temps typiques sont très faibles.

B Seconde difficulté : le caractère multi-échelles des solutions

Les solutions étudiées sont souvent multi-échelles, en temps et en espace. Cela signifie que certaines zones spatio-temporelles requièrent une finesse d'approximation élevée pour pouvoir reproduire fidèlement le comportement physique, tandis qu'en d'autres zones une approximation grossière est suffisante. Prenons l'exemple d'un incendie. Au début le foyer est très restreint et seule cette zone doit être maillée finement, car partout ailleurs *il ne se passe rien*. Petit à petit l'incendie se propage et la zone à mailler finement augmente. Un autre exemple de phénomène multi-échelle est la détonation, il faut mailler finement, au foyer de l'explosion et le front de l'onde de choc. Mais la zone non atteinte par l'explosion, qui n'a pas encore reçu le choc, pourrait être maillée très grossièrement. Dans ces conditions, il est imaginable qu'un maillage naïf mène à ce qu'en certains instants, 90% du domaine soit maillé avec un pas d'espace 100 fois plus fin que nécessaire, or en trois dimension mailler 100 fois trop finement multiplie par un million le nombre de cellules. Il y a alors une grande inefficacité computationnelle.

2.3.3 Conclusion sur les équations d'ADR

Les solutions multi-échelles ainsi que le couplage des trois opérateurs rendent les équations d'advection-diffusion-réaction difficiles à résoudre numériquement. D'une part les trois ne peuvent être traitées efficacement par une approche monolithique classique. D'autre part, le caractère multi-échelle tend au gaspillage de ressources de calculs. Cette dernière difficulté pousse les numériciens vers des stratégies d'adaptation en espace, celle étudiée dans ce stage est la multi-résolution adaptative (MRA) dont les détails techniques sont donnés à la section suivante.

2.4 La Multirésolution Adaptative

La multi-résolution adaptative (MRA) est une méthode d'adaptation en espace, très efficace pour les problèmes multi-échelles. L'objectif est de concentrer les efforts computationnels là où ils sont nécessaires. Cette méthode est très étudiée par l'équipe du CMAP qui a développé le code Samurai⁷ centré sur cette méthode. Concrètement cela consiste à augmenter la résolution de la grille de cal-



FIGURE 2.2 – Exemple de maillage adapté par multirésolution adaptative grâce au logiciel [Samurai](#).

cul où la solution est complexe et la diminuer où la solution est simple à décrire. La MRA est donc une méthode de HPC (*high performance computing*) puisqu'elle vise à optimiser l'allocation des ressources de calcul.

Cette partie introduit le lecteur à cette méthode en présentant d'abord le concept mathématique de transformée multi-échelle (ou transformée en ondelette) qui est à la base de la MRA. Puis il est expliqué comment la transformée multi-échelle permet d'adapter le maillage pour optimiser la charge computationnelle. Une fois ces prérequis établis, l'algorithme typique de mise en oeuvre de la multi-résolution adaptative est décrit. Vient enfin une présentation des différentes implémentations de la MRA.

2.4.1 La transformée multi-échelle

Cette partie présente la transformée multi-échelle discrète. La transformée multi-échelle continue en simulation numérique, c'est bien sûr la version discrète qui est utile. Elle se veut avant tout introductive et omet ou simplifie certaines notions ; plus de détails sont donnés en [29].

A Définition mathématique

Les explications sont développées en dimensions un à des fins pédagogiques, la plupart des concepts s'entendent naturellement aux dimensions supérieures. De plus, la discrétisation de l'espace se fait

7. <https://github.com/hpc-maths/samurai>

selon une grille dyadique, d'autre choix pourraient être fait mais c'est un choix simple, naturel et standard. Il faut détailler cette notion.

Définition 2.4.1 (Grille dyadique). Une grille dyadique ou discrétisation dyadique d'un intervalle $I \subset \mathbb{R}$ est une série de partitions de I indexées par des entiers $j \in J \subset \mathbb{N}^*$. La discrétisation de niveau j correspond à une partitions de I en 2^j intervalles (voir fig. 2.3). Ainsi à chaque changement de niveau, du niveau j vers le niveau $j+1$, la résolution de la discrétisation est doublée. Les cases de cette partition dyadique sont indexées par deux entiers : j le niveau de résolution de la grille et k l'index de la case au sein de ce niveau. En particulier les cases $2k$ et $2k+1$ du niveau $j+1$ correspondent à la case k du niveau j .

Niveau $j-1$	$k=0$				$k=1$			
Niveau j	$k=0$		$k=1$		$k=2$		$k=3$	
Niveau $j+1$	$k=0$	$k=1$	$k=2$	$k=3$	$k=4$	$k=5$	$k=6$	$k=7$

FIGURE 2.3 – Exemple de grille dyadique

Dans ce qui suit il est supposé sans perte de généralité que la discrétisation se fait sur l'intervalle $[0, 1]$, ainsi le niveau j correspond à des cellules de tailles $1/2^j$ et la cellule k du niveau j est centrée en $x_k^j = \frac{k+(k+1)}{2} \frac{1}{2^j} = \frac{2k+1}{2^j}$. La notion d'ondelette se définit de la manière suivante :

Définition 2.4.2 (Ondelette). Une ondelette est une fonction $\Phi \in L^2(\mathbb{R})$ à support compact de moyenne nulle. Pour qu'une ondelette soit pertinente dans le cas de la transformée en multi-échelle il est requis que la famille $(x \mapsto \Phi(2^j k - x))_{(j,k) \in \mathbb{Z} \times \mathbb{Z}}$ forme une base de $L^2(\mathbb{R})$. En effet la transformée en ondelette sera une projection sur cette base, un peu comme la transformée de Fourier est une projection sur les fonctions trigonométriques.

Alors la transformée en ondelette discrète peut être définie :

Définition 2.4.3 (Transformée en ondelette discrète - *Discrete Wavelet Transform, DWT*). Donnée une fonction f , le coefficient γ_k^j de sa DWT sur la cellule k au niveau de résolution j est :

$$\gamma_k^j = \frac{1}{N_j} \int_{\mathbb{R}} \Phi(2^j \cdot k - t) f(t) dt, \quad (2.24)$$

Où N_j est un coefficient normalisation dépendant du niveau j .

Contrairement à une transformée de Fourier, les coefficients ne dépendent pas d'une mais de deux variables. En effet, la transformée en ondelette est plus riche d'informations. Là où la transformée de Fourier ne donne qu'une information sur le contenu fréquentiel d'un signal, la transformée en ondelette donne une information sur le contenu en fréquence et sur la localisation de ce contenu fréquentiel.

B La notion de détails

La multi-résolution adaptative se sert de la transformée multi-échelle pour adapter le maillage, c'est à dire compresser l'information (voir 2.4.2). Cela requiert l'introduction de la notion de détail. Ce concept permet de ne pas utiliser la transformée en ondelette pour quantifier le contenu absolu

porté par une échelle particulière, mais plutôt à comprendre en quoi ce contenu s'éloigne de ce que les échelles supérieures pourraient laisser supposer, en quoi il est *inattendu*. Pour résumer à partir d'un niveau de résolution j , on définit un **prédicteur** polynomial, qui tente d'inférer l'allure de la fonction au niveau $j + 1$. Puis le **détail** ne cherche pas naïvement à quantifier et localiser l'information contenue aux échelles du niveau $j + 1$ mais plutôt, à quantifier l'écart à la prédiction polynomiale.

B.i Le prédicteur Donné un point central (x_0, y_0) , et $2s$ voisins $(x_{-s}, y_{-s}), (x_{-s+1}, y_{-s+1}) \dots (x_{s-1}, y_{s-1}), (x_s, y_s)$, un prédicteur polynomial ponctuel cherche le polynôme P de degré $2s$ passant par ces $2s + 1$ points. Cela permet d'inférer des valeurs pour y en tout point x . Pour trouver $P(X) = \sum_{k=0}^{2s} a_k X^k$ revient à résoudre le système linéaire :

$$\forall j \in \{-s, \dots, 0, \dots, s\} : y_j = \sum_{k=0}^{2s} a_k x_j^k \quad (2.25)$$

Ce stage se focalise sur les volumes finis, donc ce n'est pas un prédicteur ponctuel, adapté au différences finies (voir ??), qui est utilisé mais un prédicteur sur la valeur moyenne. Il ne cherche à imposer les valeurs en chaque point mais à fixer la valeur moyenne sur chaque cellule, cela ajoute peu de complexité puisqu'il suffit d'ajouter une intégration lors de l'établissement du système linéaire pour travailler sur les valeurs moyennes. Pour l'usage que souhaité ici, il s'agit en d'évaluer la solution sur la cellule k du niveau de résolution j (ce qui correspond à la cellule de centre $x_k^j = (k + 1/2)2^{-j}$) au niveau de résolution supérieure $j + 1$, il faut donc appliquer le correcteur linéaire centré sur x_k^j en $x_{\pm} = \pm 2^{-(j+1)}$. En pratique cela revient à faire une combinaison linéaire des $2s$ voisins qui vient corriger la valeur en x_k^j . Le prédicteur dépend donc du nombre de voisins pris de part et d'autre, ce nombre noté s et appelé le *stencil* du prédicteur. Plus s est grand plus l'opération de prédiction est précise (mais peut éventuellement devenir bruitée⁸) et plus elle est coûteuse. Le coût exact n'est pas évident à estimer puisque qu'une combinaison linéaire quelques termes se fait en $O(1)$ sur les machines modernes, toutefois quelques subtilités détaillé en 2.4.2 interviennent. Ainsi les valeurs usuelles du stencil sont généralement $s = 1$ ou $s = 2$

B.ii Les détails À présent que le prédicteur à été décrit le concept de détail peut enfin être abordé. On suppose que l'on dispose d'une fonction \tilde{u}^j qui soit une approximation de la fonction u , au niveau de résolution j . Comme vu précédemment, le prédicteur permet d'obtenir une approximation de u au niveau $j + 1$ grâce à \tilde{u}^j . Cette prédiction est noté \hat{u}^{j+1} . Les détails à la résolution $j + 1$ sont alors définis comme les coefficients de la transformée multi-échelle de la différence entre cette prédiction et la vraie fonction : $u - \hat{u}^{j+1}$. Alors les coefficients de détails n'encode que ce qui n'était pas prédictible par l'interpolateur polynomial. Ce concept de détail est essentiel. En pratique on ne réalise pas l'opération comme expliqué plus haut puisque à prédicteur et ondelette fixés Φ , il existe une *ondelette duale* Ψ qui permet directement d'obtenir les détails pour la DWT sur Φ en réalisant une DWT sur Ψ ce qui accélère considérablement les calculs.

Une autre façon de voir la notion de détail est la suivante : pour un niveau de résolution j on note $V_j = \text{Vect}((\Phi_k^j)_k)$, c'est à dire l'ensemble de fonction représentables par les ondelettes de niveau j . Pour les ondelettes classiques⁹ la relation suivante est vérifiée $V_0 \subset V_1 \subset V_2 \subset \dots \subset V_N$. Et bien alors

8. Ce n'est pas détaillé ici mais le lecteur se référera à la théorie de l'interpolation...

9. C'est en tout cas vrai pour les ondelettes de Haar[29].

l'espace des détails, celui accessible par l'ondelette duale est Q_{j+1} le supplémentaire de V_j dans V_{j+1} , en d'autre terme, il représente toutes les informations, les *détails* contenues dans le niveau d'approximation V_{j+1} qui n'étaient pas prise en compte par V_j (à l'échelle j ce n'était que des détails). Les coefficients de la décomposition par l'ondelette duale sont Ainsi grâce à l'ondelette duale, il est possible de calculer les coefficients de détails d_k^j qui à chaque montée en résolution n'encode que l'information qui n'était pas contenue dans la décomposition en ondelette au niveau précédent.

Les deux visions ne sont pas rigoureusement les mêmes, la première représente ce qui est réalisé en pratique lors de la MRA, la seconde est la vision standard de la théorie des ondelettes. Toutefois les deux approches ont la même motivation, ne calculer et ne mettre en valeur à chaque niveau que ce qui est nouveau, ce qui n'était pas contenu dans les niveaux précédents.

B.iii Intuition sur les détails Lorsque l'on s'intéresse aux coefficients de détails d_k^j , l'indice j de *dilatation* fixe l'échelle analysée, c'est à dire la longueur d'onde analysée. Par exemple si $j = 5$, les coefficients d_k^5 donne une information sur l'information portée par les longueurs d'onde de l'ordre de $2^{-5} = 1/32$. La variable k précise l'indice de la cellule analysée. Par exemple $\|d_1^5\| > \|d_5^5\|$ signifie que l'information portée par l'échelle $1/32$ est plus importante au voisinage de la case 10 qu'au voisinage de la case 5. De même si $\|d_7^j\| > \|d_{14}^{j+1}\|$ cela signifie qu'au voisinage de $x = \frac{7}{2^j}$ les longueurs d'ondes $\frac{1}{2^j}$ sont plus présentes que les longueurs d'ondes $\frac{1}{2^{j+1}}$. Pour se fixer les idées, c'est comme si la transformée de Fourier n'avait qu'une vision globale du contenu en fréquence, quelle ne voyait que la moyenne sur le domaine de la transformée en ondelette pour chaque longueur d'onde.

$$\|TF[f](\omega = 2^j)\|^2 \sim \sum_k \|d_k^j\|^2. \quad (2.26)$$

Grâce à cette notion de détaille la décomposition multi-échelle permet une description de la solution physique où l'apport à la solution de chaque échelle, de chaque distance typique est quantifié, les coefficients d_k^j décrivant l'information contenue dans les échelles de l'ordre de 2^{-j} .

2.4.2 L'adaptation

A La compression par décomposition multi-échelle

Pour compresser une fonction (ou une image comme dans le processus jpg), le processus est très simple, il suffit de fixer un seuil de compression $\varepsilon > 0$, de calculer les coefficients de détails de la fonction, puis d'omettre (en pratique de retirer de la mémoire) les coefficient dont la norme est inférieure à ε . En pratique le coefficient de compression dépend du niveau étudié puisque le volume des cellules mise en jeu chute avec le niveau j (un détail de 10^{-2} à moins d'importance s'il porte sur des cellules de taille 10 que sur des cellules de taille 10^{-5}). En pratique l'algorithme serait le suivant :

1. Calculer les coefficients d_k^j .
2. Pour chaque niveau j et chaque coefficient d_k^j : Si $|d_k^j| < \varepsilon_j = 2^{-j}\varepsilon$, alors $d_k^j \leftarrow 0$, les coefficients sont seuillés.

Pour reconstruire au niveau de résolution souhaité : utiliser les détails jusqu'au niveau j le plus fin conservé puis utiliser le prédicteur pour interpoler jusqu'au niveau désiré. Le vocabulaire est heureusement choisit, pour compresser on omet les détails négligeables l'on conserve les détails importants.

B L'adaptation de maillage et l'heuristique d'Harten

L'adaptation de maillage, est une variation de la compression par décomposition multi-échelles précédemment décrite. C'est est une opération de compression de solutions physiques *prudente*. Les coefficients y sont seuillés de manière moins impitoyable : certains coefficients qui devraient être écartés par la compression sont malgré tout préservés. Ce choix se fait sur la base d'intuitions physiques, la plus connue étant *l'heuristique d'Harten*, introduite par Ami Harten [19], le père de la MRA. Elle stipule que même si un coefficient de détail d_k^j devrait être supprimé, si le niveau de détail du niveau supérieur (c'est à dire $d_{\lfloor k/2 \rfloor}^{j-1}$) est particulièrement élevé, par exemple qu'il est par deux fois supérieur au seuil ε_{j-1} , alors le coefficient doit être conservé. En d'autre terme, même si la compression considère l'information à l'échelle j négligeable, l'intuition physique pose son veto puisque les échelles supérieures sont de grandes magnitudes et que cela présage que dans les pas de temps à venir les échelles seront nécessaires à la fidèle capture des phénomènes physiques simulés. Par exemple cela peut signifier qu'un front d'onde est en train d'arriver dans les zone étudiée, il faut donc que la simulation puisse en capter toute la richesse.

2.4.3 Algorithmes de simulation numérique

A Algorithme général

L'intégration de la MRA un algorithme de simulation physique se fait de la manière suivante : tous les n_{MRA} pas de temps (éventuellement être à chaque pas de temps), la solution physique est adaptée selon le procédé explicité ci-dessus. Les détails précédemment omis devenant nécessaires sont obtenus grâce au prédicteur. Puis la simulation se poursuit à partir de cette grille de calcul adapté. L'économie de mémoire se fait puisque tous les détails ne sont pas stockés et l'économie de calculs puisque moins de cellules sont mises en jeu lors du déroulement de l'algorithme de simulation.

Par exemple, en dimension une, si une zone est adapté avec un niveau de détail de niveau 5, la densité de cellule sur lequel il faut réaliser des est de 2^5 et la densité mémoire est de $\sum_{j=0}^5 2^j$. Si le niveau le plus fin de la grille est par exemple 9, il y a un gain computationnel théorique de l'ordre de $2^{9-5} = 16$ et en terme de mémoire une économie de l'ordre de $\frac{\sum_{j=0}^5 2^j}{2^8} > 10$. Ce gain est exponentiel avec la dimension du problème.

B Le cas volumes finis

Il faut détailler ce qui signifie dans le cadre des volumes finis, *poursuivre la simulation sur la grille de calcul adaptée*. Le maillage adapté est constitué de cellules de tailles différents. Il s'agit alors pour chacune de ces cellules de prédire la valeur moyenne en son sein au pas de temps suivant, c'est à dire faire évoluer le maillage. Cela se fait en évaluant un flux à partir des valeurs de la solution aux interfaces de la cellule. Tout l'objet des volumes finis est : à partir de valeurs moyennes sur les cellules voisines, estimer les valeurs ponctuelles aux interfaces. La MRA amène alors une subtilité, les schémas volume finis usuels s'appuient sur des cellules de volumes similaires de l'ordre de Δx^d . Cependant,

Chapitre 3

Contribution

Ce chapitre présente les travaux du stage. La première contribution analyse deux méthodes ImEx appliquées à l'équation de Nagumo (une équation de diffusion-réaction) et les compare à une méthode de *splitting* sur les questions de stabilité et de convergence. La seconde contribution étudie l'impact de la multirésolution adaptative sur un problème de diffusion. Cette étude débute avec un volet théorique (obtention de l'équation équivalente du schéma avec multirésolution adaptative) et poursuit par un volet expérimental (étude de convergence grâce au logiciel Samurai).

3.1 Étude de méthodes ImEx sur une équation de diffusion-réaction

L'objectif est de comparer la pertinence des méthodes RK implicites-explicites au *splitting* d'opérateurs traditionnel. Pour introduire cette première étude l'équation de Nagumo est d'abord présentée comme un excellent cas test pour éprouver les méthodes de résolution des équations d'advection-diffusion-réaction. Dans un second temps les méthodes ImEx utilisées sont détaillées. Par la suite leur stabilité est évaluée dans un contexte général; puis en se focalisant sur l'équation de Nagumo, où elles sont comparée à une méthode de séparation d'opérateur classique (splitting de Strang). Ceci permet de valider la pertinence *a priori* de ces méthodes sur les équations de réaction-diffusion, et mène naturellement à une étude de convergence expérimentale.

3.1.1 L'équation de Nagumo

L'équation de Nagumo (ou FitzHugh-Nagumo) est issue de modèles de transmission de l'information nerveuse [15]. L'étude travaille sur la forme spatiale de l'équation [21] avec un terme de réaction cubique introduisant de la non-linéarité :

$$\partial_t u = \underbrace{D \partial_{xx} u}_{\text{diffusion}} - \underbrace{ku(1-u^2)}_{\text{réaction}}. \quad (3.1)$$

A Solutions Analytiques

L'équation admet des solutions propagatives sous la forme¹ :

$$u(x - ct) = \frac{e^{\sqrt{\frac{k}{2D}}((x-x_0)-ct)}}{1 + e^{\sqrt{\frac{k}{2D}}((x-x_0)-ct)}} \quad (3.2)$$

Avec : $c = \sqrt{\frac{kD}{2}}$ et x_0 le point de départ de l'onde.

Ainsi, le produit kD fixe la vitesse et le ratio $\frac{k}{D}$ la magnitude du gradient d'espace.

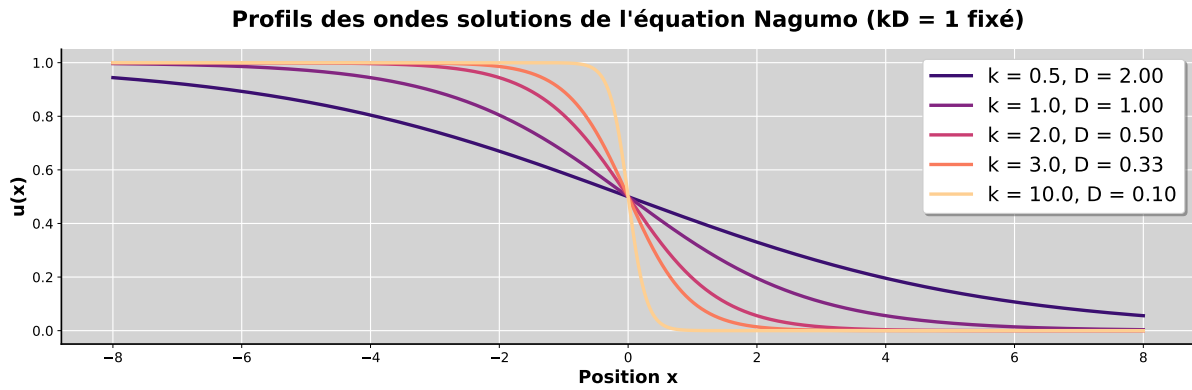


FIGURE 3.1 – Profils des ondes solutions de l'équation de Nagumo pour différents ratios k/D avec le produit $kD = 1$ fixé (c'est à dire à vitesse fixée). L'augmentation du ratio k/D accentue le gradient spatial.

B Analyse des opérateurs

Un analyse des deux opérateurs de l'EDP est nécessaire pur en saisir les enjeux. *L'opérateur de diffusion* est non-local et, discrétisé à l'ordre deux par n points et un pas Δx , les valeurs propres associées sont $\{\frac{2D}{\Delta x^2}(\cos \frac{p\pi}{n+1} - 1) \mid p \in \{1, \dots, n\}\}$ [7], ainsi la raideur du terme de diffusion croît linéairement avec le coefficient de diffusion D et de manière quadratique avec la finesse du maillage $1/\Delta x$. En effet les valeurs propres sont négatives et :

$$\max_p \left| \cos \frac{p\pi}{n+1} - 1 \right| \sim 2, \quad (3.3)$$

$$\min_p \left| \cos \frac{p\pi}{n+1} - 1 \right| \sim \frac{1}{2} \left(\frac{\pi}{n+1} \right)^2. \quad (3.4)$$

1. C'est une sigmoïde, qui se propage à vitesse $\sqrt{\frac{kD}{2}}$ [13].

Et donc :

$$\frac{\max_p |1 - \cos \frac{p\pi}{n+1}|}{\min_p |1 - \cos \frac{p\pi}{n+1}|} \approx n^2 \propto \left(\frac{1}{\Delta x} \right)^2. \quad (3.5)$$

Concernant *le terme de réaction*, en choisissant un état initial correspondant à 3.2, la solution reste entre 0 et 1. Ainsi le terme de réaction est local en espace, et ses valeurs propres sont comprises entre $-k$ et $2k$. En fonction de la valeur de u , la réaction se comporte localement (dans le temps et l'espace) comme une relaxation de temps caractéristique $\tau \sim \frac{1}{k}$ ou comme une explosion de temps caractéristique $\tau \sim \frac{1}{2k}$, en effet :

$$\text{Terme de réaction : } R(u) = ku(1 - u^2), \quad (3.6)$$

$$\text{Valeurs propres de la réaction : } R'(u) = k(1 - 3u^2). \quad (3.7)$$

Pour les valeurs étudiés, $k \leq 20$, la réaction reste ainsi peu raide par rapport au "vraies" réactions chimiques (il faut avoir conscience de cette différence pour considérer ce cas test de la bonne manière, ici la réaction est le terme le moins raide alors que sur des "vrais" applications, ce n'est pas le cas).

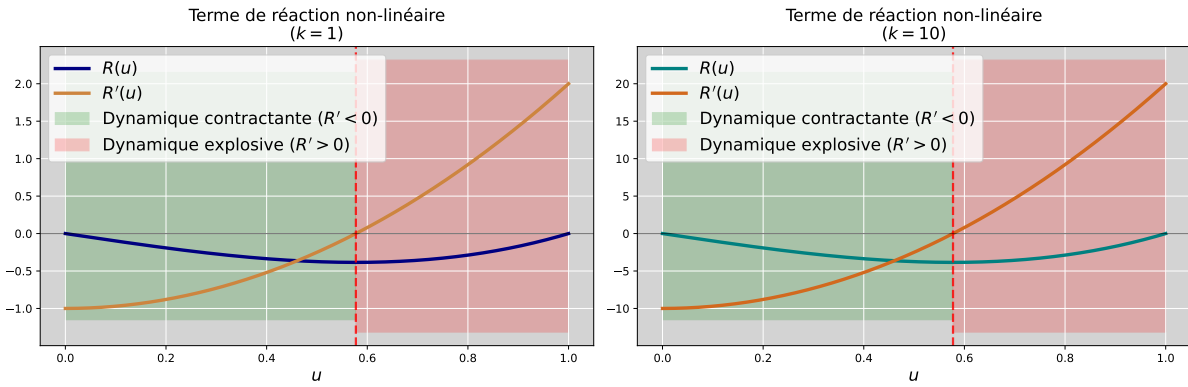


FIGURE 3.2 – Plage de valeurs du terme de réaction non-linéaire et de sa différentielle pour deux coefficients de réactions : $k = 1$ et $k = 10$.

C Conclusion sur l'équation de Nagumo

Ainsi l'équation de Nagumo, présente un terme de réaction², et un terme de diffusion. Cette équation fait émerger un front d'onde³ et dispose de deux paramètres k et D pour piloter aisément les propriétés des solutions. Cela en fait donc une équation-test de choix pour étudier le comportement de diverses méthodes dédiées aux équations d'advections-réaction-diffusion.

2. à noter qu'il n'est pas raide, comparé aux termes de réaction rencontrés en combustion.

3. Cela permet de tester le comportement de la multi-résolution adaptative.

3.1.2 Les méthodes ImEx

Les méthodes ImEx étudiées sont les méthodes de Runge et Kutta additives (RK-ImEx ou RK-additive). Ces méthodes consistent à sommer plusieurs méthodes de Runge et Kutta appliquées chacune à un opérateur différent. *L'objectif est d'intégrer chaque opérateurs avec des méthodes RK différentes (implicites ou explicites), en accord les spécificité de chaque opérateur et cela, indépendamment des autres opérateurs.*

A Un exemple

Pour introduire la méthodes de Runge et Kutta additives, on commence par un exemple simple usant d'une méthode RK-ImEx d'ordre un. Cette ImEx naît de la conjugaison de deux méthodes Runge et Kutta à un étages (RK1). Cette méthode est notée ImEx111 [5]. Les méthodes RK1 servant de briques élémentaires à la RK111 sont : un schéma d'Euler explicite et un schéma d'Euler implicite. Soit une équation d'évolution faisant intervenir deux opérateurs :

- ◊ L'opérateur A^E se prêtant aux méthodes explicites (par exemple, un opérateur peu raide mais non local).
- ◊ L'opérateur A^I se prêtant aux méthodes implicites (par exemple un opérateur raide mais local).

L'équation cible serait alors de la forme :

$$\partial_t u = A^E u + A^I u. \quad (3.8)$$

A.i Résolution par approche monolithique En n'utilisant qu'une seule RK1 pour tout le problème (approche monolithique), la dynamique serait approchée d'une des façon suivante. En simulant avec un schéma d'Euler explicite monolithique, la méthode s'écrit alors :

$$u^{n+1} = u^n + \Delta t (A^E + A^I) u^n. \quad (3.9)$$

Mais si l'opérateur A^I est très raide, la stabilité risque d'imposer un pas de temps très restrictif risquant de rendre la méthode non viable. En résolvant avec un schéma d'Euler implicite monolithique, la méthode s'écrit :

$$u^{n+1} = (Id - \Delta t (A^E + A^I))^{-1} u^n. \quad (3.10)$$

Mais si l'opérateur A^E rend l'inversion coûteuse; par exemple s'il est non-local (impliquant la résolution d'un gros système au lieu de plusieurs petits systèmes), et/ou s'il est non linéaire (nécessite d'être réinverser à chaque pas de temps); alors cette méthode ne sera pas viable non plus.

A.ii Résolution par une méthode ImEx : une Runge et Kutta Additive Lorsque la méthode ImEx111 est choisie, l'approximation au pas de temps $n+1$ s'écrit en sommant une contribution issue de la méthode Euler explicite (RKE1) et une contribution issue de la méthode Euler implicite (RKI1) :

$$u^{n+1} = u^n + \Delta t \left(\underbrace{k_1}_{\text{RKE1}} + \underbrace{k'_1}_{\text{RKI1}} \right) \quad (3.11)$$

La contribution issue de la RKE1 appliquée à A^E s'écrit (Euler explicite) :

$$k_1 = A^E u^n. \quad (3.12)$$

La contribution issue de la RKI1 appliquée à A^I s'écrit (Euler implicite) :

$$k'_1 = A^I u^{n+1} \quad (3.13)$$

Ainsi :

$$\begin{aligned} u^{n+1} &= u^n + \Delta t (A^E u^n + A^I u^{n+1}), \\ \text{donc : } u^{n+1} - \Delta t A^I u^{n+1} &= u^n + \Delta t A^E u^n, \\ \text{et donc : } u^{n+1} &= (Id - \Delta t A^I)^{-1} \circ (Id + \Delta t A^E) u^n. \end{aligned} \quad (3.14)$$

Ainsi dans cette méthode seul l'opérateur $Id - \Delta t A^I$ est inversé et les problèmes de raideurs sont résolus ; ce qui était l'objectif. Les traitements sur les opérateurs sont bien découplés lors de la résolution.

B Cadre mathématique général

Pour construire des méthodes plus complexes et d'ordres supérieurs introduisons le formalisme de [5] pour traiter les méthodes RK-additives. Ici, nous travaillons uniquement sur méthodes ImEx pour deux opérateurs mais théoriquement, il est possible de construire des méthodes ImEx pour traiter autant d'opérateurs que l'on le souhaite [22].

B.i Notations Une méthode ImEx additive est construite à partir d'une méthode implicite à s étages (une méthode DIRK et si possible SDIRK [23]) et d'une méthode explicite à $s + 1$ étages⁴. Pour uniformiser, le tableau de Butcher de la méthode implicite est complété par une ligne et une colonne de zéros afin que les deux méthodes s'écrivent comme si elles avaient le même nombre d'étages. Les tableaux de Butcher des deux méthodes s'écrivent alors :

Méthode RKE, $s + 1$ étages :

$$\text{RKE : } \begin{array}{c|c} \tilde{c} & \tilde{A} \\ \hline & \tilde{b}^T \end{array} = \begin{array}{c|cccccc} 0 & 0 & 0 & 0 & \cdots & 0 \\ \tilde{c}_1 & \tilde{a}_{10} & 0 & 0 & \cdots & 0 \\ \tilde{c}_2 & \tilde{a}_{20} & \tilde{a}_{21} & 0 & \cdots & 0 \\ \vdots & \vdots & \vdots & \ddots & \ddots & \vdots \\ \tilde{c}_s & \tilde{a}_{s0} & \tilde{a}_{s1} & \tilde{a}_{s2} & \cdots & 0 \\ \hline & \tilde{b}_0 & \tilde{b}_1 & \tilde{b}_2 & \cdots & \tilde{b}_s \end{array} \quad (3.15)$$

4. Au besoin, la méthode explicite peut être à s étages, qui est un cas particulier d'une méthode à $s + 1$ étages.

Méthode RKI (DIRK) s étages :

$$\text{RKI: } \begin{array}{c|c} c & A \\ \hline & b^T \end{array} = \begin{array}{c|cccccc} 0 & 0 & 0 & 0 & \cdots & 0 \\ c_1 & 0 & a_{11} & 0 & \cdots & 0 \\ c_2 & 0 & a_{21} & a_{22} & \cdots & 0 \\ \vdots & \vdots & \vdots & \ddots & \ddots & \vdots \\ c_s & 0 & a_{s1} & a_{s2} & \cdots & a_{ss} \\ \hline & 0 & b_1 & b_2 & \cdots & b_s \end{array} \quad (3.16)$$

où les coefficients \tilde{a}_{ij} , \tilde{b}_i , \tilde{c}_i définissent la méthode explicite et les coefficients a_{ij} , b_i , c_i définissent la méthode implicite DIRK.

B.ii Schéma général d'une méthode RK-additive Une étape de la méthode RK-additive appliquée entre les pas de temps n et $n+1$ au système $\frac{du}{dt} = A^E u + A^I u$ s'écrit :

Calcul des approximations intermédiaires : Les calcul des approximations aux pas de temps intermédiaires $(u_i)_{i \in \{0, \dots, s\}}$ se fait grâce à la relation :

$$u_i = u^n + \Delta t \sum_{j=0}^{i-1} \tilde{a}_{ij} A^E u_j + \Delta t \sum_{j=0}^i a_{ij} A^I u_j, \quad i = 0, 1, \dots, s, \quad (3.17)$$

En initialisant $u_0 = u^n$.

Soit en mettant en lumière le caractère implicite de la méthode sur A^I :

$$(Id - \Delta t a_{ii} A^I) u_i = u^n + \Delta t \sum_{j=0}^{i-1} (\tilde{a}_{ij} A^E u_j + a_{ij} A^I u_j), \quad i = 0, 1, \dots, s \quad (3.18)$$

Calcul de l'approximation définitive :

$$u^{n+1} = u^n + \Delta t \sum_{i=0}^s \tilde{b}_i A^E u_i + \Delta t \sum_{i=0}^s b_i A^I u_i \quad (3.19)$$

Cette formulation générale permet de construire des méthodes d'ordre élevé.

B.iii Ordre de convergence L'ordre d'une méthode RK-additive est évidemment borné par l'ordre des méthodes RK individuelles convoquées. Naturellement, cette borne n'est pas nécessairement atteintes; des conditions d'ordre liant les coefficients des méthodes individuelles entre eux doivent être respectées. Le nombre de ses conditions augmente (très) rapidement avec l'ordre de la méthode et le nombre d'opérateurs à résoudre [22], le lecteur motivé se référera par exemple à [17].

3.1.3 Analyse de stabilité

L'objectif est d'appréhender la viabilité des RK-ImEx sur l'équation de Nagumo. Dans ce but, leur stabilité est étudiée. Dans un premier temps, une étude générale de la stabilité de des RK-ImEx est menée. Puis, l'étude de stabilité se centre sur l'application à l'équation de Nagumo. L'ensemble des codes utilisés pour évaluer numériquement et afficher les domaines de stabilités sont disponibles à l'adresse : https://github.com/Ocelot-Pale/ImEx_stability_Nagumo.

A Étude de stabilité générale des RK-ImEx

Avec une méthode ImEx, les deux opérateurs de l'EDP sont découplés, c'est là l'intérêt. Cependant cela complexifie l'analyse usuelle de stabilisé. En effet la fonction de stabilité attend alors deux variables, le coefficient spectral Z_E associé à l'opérateur traité explicitement et le coefficient spectral Z_I associé à l'opérateur traité implicitement. Ainsi, pour chaque couple (Z_E, Z_I) d'indices spectraux, la fonction de stabilité prend une valeur différente, et comme les coefficients spectraux sont des nombres complexes, on ne peut plus visualiser d'un simple coup d'oeil le domaine de stabilité (comme en ... AFAIRE), puisque celui-ci se trouve dans un espace de dimension quatre⁵.

A.i Calcul des fonction d'amplification Afin d'étudier la stabilité linéaire des méthodes, les fonctions d'amplifications ont été numériquement évaluées grâce à une fonction informatique. La démarche est la suivante :

1. Entrer les valeurs de (Z_E, Z_I) pour lesquelles la fonction de stabilité doit être évaluée
2. Simuler un pas schéma en partant de $u_0 = 1$ appliqué à une équation du type Dahlquist :

$$\partial_t u = \lambda_E u + \lambda_I u$$
 - (a) Construire toutes les approximations intermédiaires avec les valeurs
 - (b) Construire l'approximation finale u_1
3. Évaluer la norme de U_1

Cette étude n'est pas détaillée ici par nécessité de concision, le lecteur intéresser pourra trouver les graphiques représentants les domaines de stabilité sur le [Notebook en ligne](#).

B Étude de stabilité appliquée à l'équation de Nagumo

Nous allons particulariser la démarche suivante en la centrant sur l'équation de Nagumo. Cela va nous permettre de comprendre comme se comportent les méthodes ImEx sur ce problème particulier.

B.i Valeurs propres mises en jeu Comme expliqué en 3.1.1 l'équation présente deux opérateurs :

- ◇ La diffusion dont le spectre s'étend de $\frac{-1}{L^2}$ à $\frac{-1}{\Delta x^2}$ (où L est la taille du domaine discrétisé).
- ◇ La réaction dont le spectre balaie continûment $-k$ jusqu'à $2k$

Pour restreindre l'analyse de stabilité il faut donc tracer le diagramme de stabilité des méthodes étudiées en prenant $Z_I \in \mathbb{R}^-$ et $Z_E \in [-k; 2k] \subset \mathbb{R}$ ce qui nous donne un espace à deux dimensions. Il faut ensuite placer des couples (Z_E, Z_I) correspondant. Lorsque l'on réalise ce travail nous trouvons les diagrammes suivant :

5. En effet la fonction de stabilité $\mathbb{R}\mathbb{C} \times \mathbb{C} \rightarrow \mathbb{R}$ et $\dim \mathbb{C} \times \mathbb{C} = 4$.

B.ii Résultats Le lecteur est invité à prendre un peu de temps pour comprendre la logique de ces graphiques car ils sont très éclairants. Ces diagrammes permettent d'analyser respectivement la stabilité de la méthode ImEx222, de la méthode ImEx232, ainsi qu'à titre de comparaison, la stabilité d'une méthode RKE d'ordre 2⁶ et d'une méthode de splitting⁷. Chaque colonne représente l'analyse d'une méthode différente. La première ligne présente le domaine de stabilité en fonction des indices spectraux $Z_E \in \mathbb{R}$ et $Z_I \in \mathbb{R}^-$. Les points bleus représentent les couples d'indices spectraux intervenant dans la résolution de l'équation de Nagumo pour les paramètres d'équation choisis (D et k) et les paramètres de discrétisation retenus (Δt et Δx). La seconde ligne n'est qu'un zoom de la première autour de ces indices spectraux. La dernière colonne (splitting) présente une disposition différente, puisque les opérateurs sont totalement découplés. La première ligne correspond à la fonction de stabilité de la méthode explicite (avec un zoom autour des indices spectraux de la réaction). la seconde ligne représente la fonction de stabilité de la méthode implicite. Dans les deux cas, l'intervalle tracé en bleu représente la plage de valeurs d'indices spectraux balayés par chaque opérateur.

B.iii Analyse

Analyse générale Analysons les domaines de stabilité des figures en fig. 3.3, pour l'instant nous ignorons les marqueurs bleus sur les figures.

- ◊ **Méthode RKE2 :** En troisième colonne, le diagramme de stabilité d'une méthode explicite naïve RKE2, sert de référence. Le domaine de stabilité accepte des valeurs propres négatives de magnitude deux, ce qui est résultat classique des méthodes Runge et Kutta explicites d'ordre deux. Ainsi domaine de stabilité s'étend jusqu'à -2 selon l'axe portant Z_E tant que la valeur propre Z_I est négligeable. De même le domaine de stabilité s'étend jusqu'à -2 selon Z_I tant que la valeur propre Z_E est négligeable. Enfin il y a une zone intermédiaire quand Z_E et Z_I sont tous les deux de l'ordre de l'unité⁸, où la raideur résultante est $Z_E + Z_I$.
- ◊ **Méthode ImEx232 :** En observant la seconde colonne, nous constatons que la méthode ImEx232 maintient un domaine de stabilité restreint (jusqu'à -2) selon l'axe Z_E , mais selon l'axe Z_I , le domaine de stabilité s'est étendu considérablement. C'est logique puisque la valeur propre Z_E est explicitée, son domaine pris seul n'a évolué, et la valeur propre Z_I peut être très raide (très négative) puisque la méthode explicite l'opérateur lié à Z_I .
- ◊ **Méthode ImEx222 :** Passant à la première colonne, le domaine de stabilité ImEx222 ressemble beaucoup à celui de l'ImEx232. Seulement, le domaine de stabilité s'élargit considérablement selon Z_E , pourvu que Z_I soit assez grand. Cette propriété est remarquable, cela signifie que la méthode traite couple les raideurs dans son traitement. Plus précisément, plus l'opérateur implicite est raide, plus l'opérateur explicite peut être raide⁹.

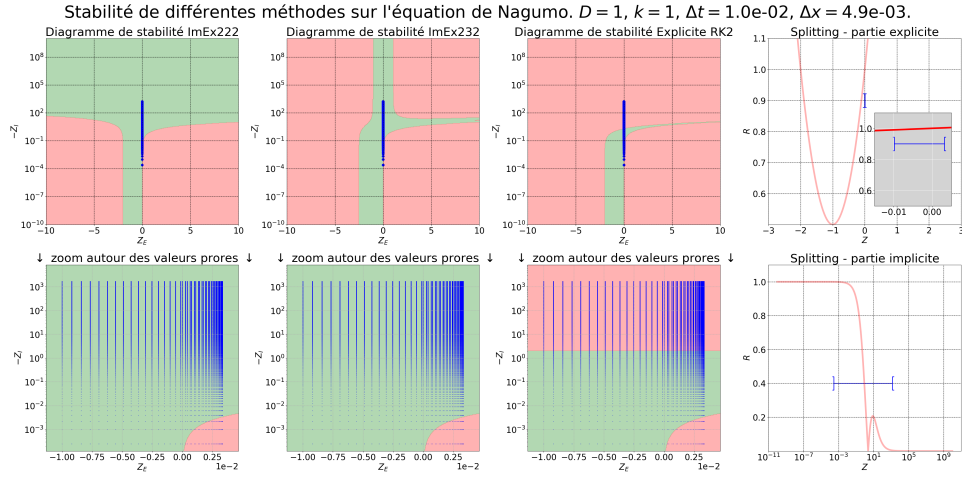
Analyse selon les paramètres de l'équation k et D Analysons grâce aux graphiques fig. 3.3 la disposition des valeurs couples de valeurs propres mis en jeu par l'équation de Nagumo selon les

6. Celle apparaissant dans ImEx222.

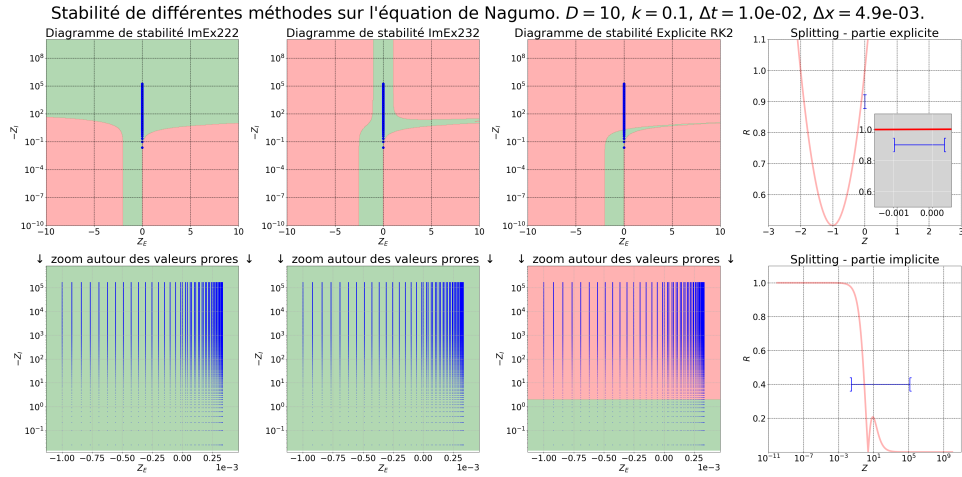
7. Où l'on utilise les méthodes implicites et explicites de la méthode ImEx222 mais dans un contexte de splitting de Strang.

8. Attention à l'échelle logarithmique.

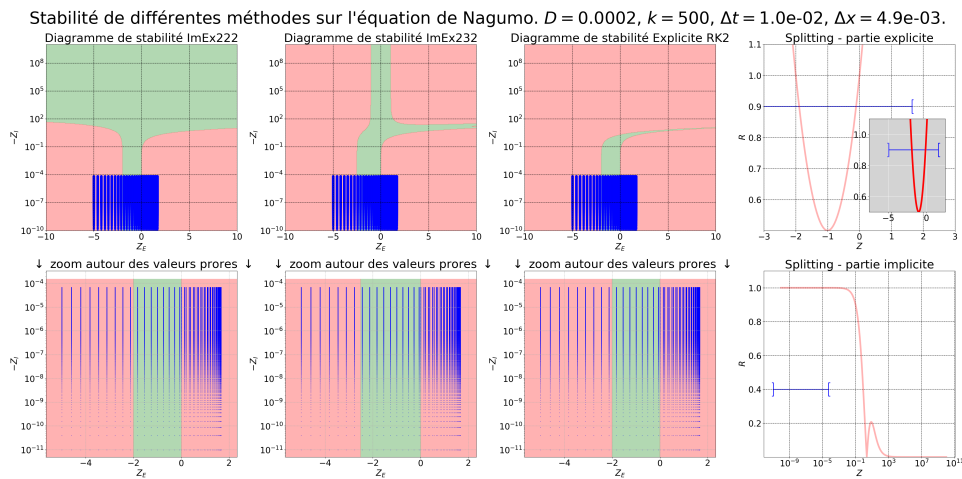
9. Cette analyse est partiellement erronée, nous verrons pourquoi au prochain paragraphe.



(a) Cas standard
 $D=1, k=1, dt=1.0e-03, dx=2.4e-03$



(b) Cas diffusion plus raide, réaction moins raide
 $D=10, k=0.1, dt=1.0e-02, dx=4.9e-03$



(c) Cas diffusion moins raide, réaction plus raide
 $D=2e-4, k=500, dt=1.0e-02, dx=4.9e-03$

FIGURE 3.3 – Pour différents couples D et k , diagrammes de stabilité des méthodes ImEx et de référence sur l'équation de Nagumo.

paramètre k et D . Les paramètres de simulation : Δt et Δx sont fixés. Les jeux de valeurs choisis sont $(k, D) = (1, 1)$, $(k, D) = (0.1, 10)$, $(k, D) = (500, 2 \cdot 10^{-4})$. Le produit kD est maintenu égal à un, ainsi la vitesse de propagation est toujours la même. Ces couples de valeurs propres Z_E, Z_I mis en jeu par les opérateur de l'équation sont tracés en bleus¹⁰.

♦ **Cas standard**, $(k, D) = (1, 1)$ - fig. 3.3a :

Dans ce cas, la raideur de la diffusion (Z_I) déstabilise la méthode RKE2 (on voit que de nombreux couples de v.p. entrent dans les zones rouges quand Z_I augmente). Pour ces valeurs de $(\Delta x, \Delta t)$ cette méthode n'est donc pas viable. C'est tout à fait normal, les méthodes imposent des pas de temps très restrictifs sur les problèmes de diffusion. En revanche, les méthodes ImEx sont tout à fait stable puisque, comme constaté précédemment, le domaine de stabilité s'étend infiniment quand $Z_I \rightarrow -\infty$. Le point notable est que certains couples de valeurs propres tombent malgré tout dans une zone instable (en bas à gauche). Mais cela n'est pas un problème car il s'agit de couples de valeurs propres où la valeur propres¹¹ de l'opérateur de réaction (Z_E) est positive. Donc la méthode n'est pas instable au sens où elle reflète simplement la dynamique explosive de la réaction. D'ailleurs si on se penchant sur le graphique de la partie explicite du splitting, on constate qu'il y a une zone (correspondant à Z_E positive) où la fonction d'amplification est d'amplitude supérieure à un, le splitting reproduit donc fidèlement la dynamique de la réaction. Ce qui peut être un problème est l'inverse, pour les méthodes ImEx, il y a des couples de valeurs propres où Z_E est positif et où la fonction d'amplification est d'amplitude inférieure à un. Cela pourrait être un frein pour reproduire fidèlement la dynamique explosive de la réaction dans les zones concernées¹².

♦ **Cas diffusion raide, réaction peu raide**, $(k, D) = (0.1, 10)$ - fig. 3.3b :

Ici, $D = 10$ donc toutes les valeurs propres liées à la diffusion sont multipliées par 10 par rapport au cas précédent. De fait la méthode RK2E de référence présente des instabilités pour encore plus de couples de valeurs propres n'est pas viable. Concernant les méthodes ImEx222 et ImEx232 elles sont stables, et cette fois-ci toutes les valeurs propres liées à la dynamique explosive de la réaction sont amorties.

♦ **Cas diffusion peu raide, réaction très raide**, $(k, D) = (500, 2 \cdot 10^{-4})$ - fig. 3.3c

Dans ce cas de figure, $k = 500$. La grande valeur du coefficient de réaction rend cette dernière très raide. Cela a pour effet de dilater selon l'axe des abscisse les indices spectraux puisque $Z_E \in [-500\Delta t, +1000\Delta t]$ alors que dans le cas $k = 1$: $Z_E \in [-\Delta t, 2\Delta t]$. Ici la méthode explicite au sein des ImEx n'est plus stable pour la réaction, ainsi toutes les méthodes deviennent instables. Le splitting également devient instable car il utilise aussi la méthode RK2E pour la réaction. Le fait que la méthode explicite de l'ImEx soit instable pour l'opérateur explicite peu sembler un obstacle infranchissable, cependant ce n'est pas si simple. Pour illustrer ce point, étendons l'analyse avec le cas spécial en fig. 3.4, dans ce cas la réaction est toujours raide $k = 500$ mais la diffusion est également très raide car $D = 500$ ¹³ Alors la méthode ImEx222 de-

10. Pour les Z_I le spectre est discret, pour Z_E , le spectre est continu, il a donc fallu échantillonner le long de l'axe Z_E

11. Dans cette section, nous identifions valeurs propres λ et indices spectraux $z = \lambda\Delta t$ puisque le pas de temps Δt est maintenu constant. Cette identification permet de discuter directement en termes de raideur des opérateurs.

12. Il n'est pas évident d'avoir *a priori* la bonne intuition car peut être que la diffusion calme en quelque sorte le caractère explosif de la réaction et qu'alors une fonction d'amplification d'amplitude < 1 est normal... Restons prudent sur cette analyse.

13. Jusqu'ici, la vitesse de propagation était la même dans tous les scénarios puisque kD était maintenu constant. Dans

vient stable, comme vu en B.iii, plus l'opérateur traité implicitement est raide, plus la méthode permet à l'opérateur traité explicitement d'être raide. C'est un cas remarquable où le couplage intervenant au sein de la méthode ImEx la rend plus stable que le splitting!

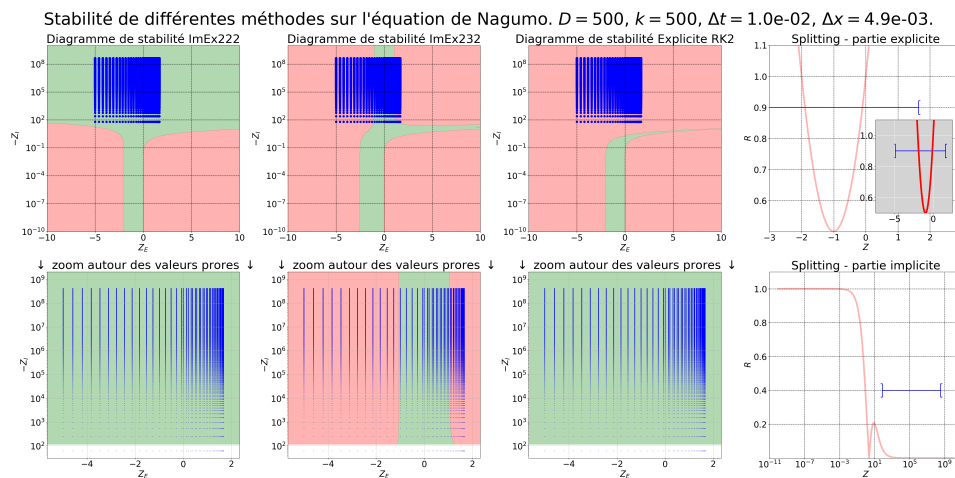


FIGURE 3.4 – Pour $k = 500$ et $D = 500$: diagrammes de stabilité des méthodes ImEx et de référence sur l'équation de Nagumo.

le scénario présenté ici, ce n'est plus le cas

3.1.4 Étude de la convergence

À présent que la stabilité des deux méthodes ImEx ont été comparées au *splitting* d'opérateur, il est naturel de poursuivre par une expérience numérique pour qualifier la convergence de chaque méthode et d'évaluer la pertinence des méthodes ImEx face au *splitting*.

Présentation de l'expérience : L'expérience est réalisée sur l'équation de Nagumo 1D à partir d'une solution initiale correspondant au profil de l'onde propagative de l'équation (voir 3.1.1). Succinctement, la simulation a lieu sur le domaine spatial $[-20, +20]$ entre $t = 0$ et $t = 3$. La grille spatiale est divisée en 2^{13} cellules ce qui équivaut à un pas d'espace $\Delta x \approx 4.8 \cdot 10^{-3}$. Des conditions de Neumann homogènes et une vitesse de propagation adaptées permettent de maintenir le front d'onde au centre du domaine et de négliger les effets de bords afin de comparer à la solution analytique exacte d'onde propagative. Les erreurs sont calculées sur le domaine $[-5, +5]$ pour ce centrer sur l'étude du front d'onde.

Résultats : Les résultats de l'expérience sont présentés en 3.5. Il apparaît que si le schéma de *splitting* et ImEx222 ont des performances similaires, la méthode ImEx232 a clairement une constante de convergence plus faible. De fait on voit que sur un maillage classique (non-adapté), les méthodes ImEx peuvent apporter une cohérence globale qui mène à une meilleure précision. Dans la section suivante, nous allons réitérer l'expérience sur un maillage adapté par multi-résolution adaptative pour étudier si l'adaptation en espace interagit avec les méthodes ImEx et la méthode de *splitting*.

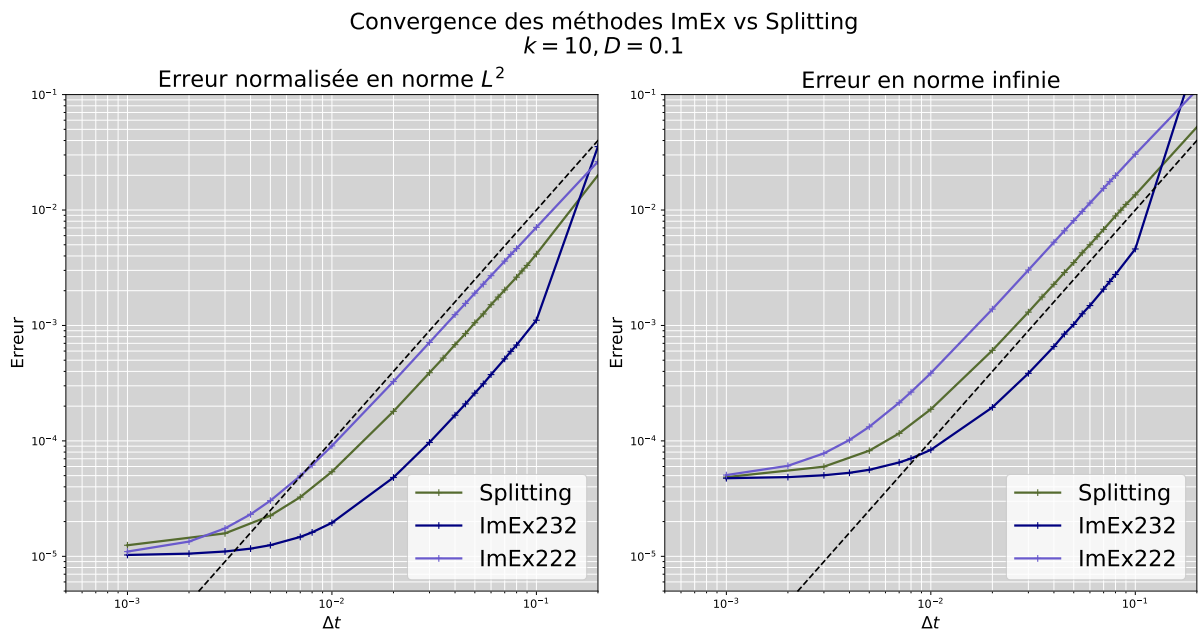


FIGURE 3.5 – Comparaison de la convergence du schéma de *splitting* avec celle des méthodes ImEx222 et ImEx232 sur l'équation de Nagumo avec $D = 0.1$ et $k = 10$

3.1.5 Mise en lumière expérimental de couplages entre la méthode en temps et l'adaptation spatiale

Objectifs et contexte de l'étude : L'objectif est d'observer d'éventuelles interactions entre la méthode de découplage des opérateurs (ImEx/splitting) et l'adaptation en espace par multi-résolution adaptative. Pour ce faire la comparaison entre ImEx222, ImEx2332 et splitting a été refaite (fig 3.6) en adaptant spatialement chaque schéma par MRA. Si des différences par rapport à l'étude précédente (fig. 3.5) apparaissent, ils résultent nécessairement de couplages entre la méthode d'intégration en temps et l'adaptation spatiale.

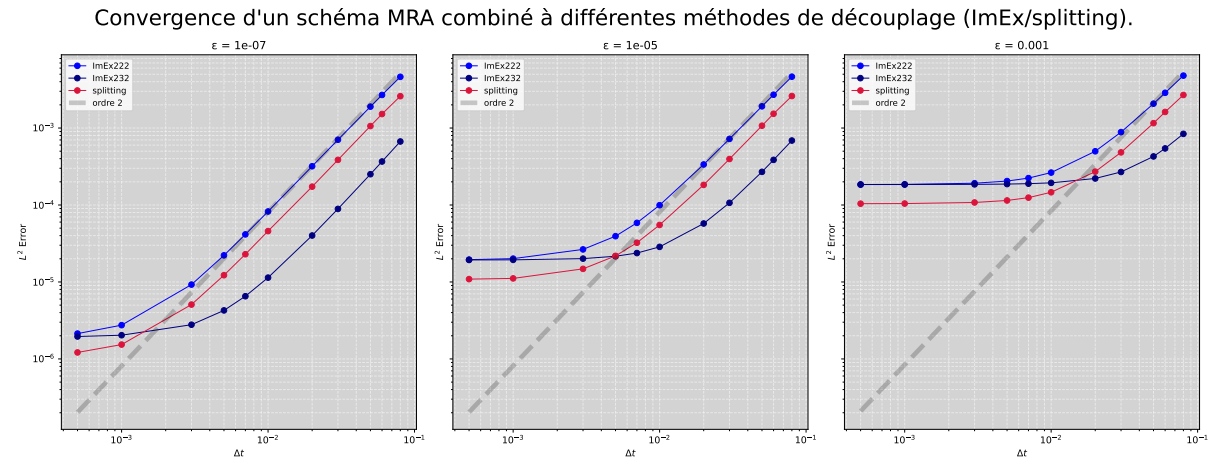


FIGURE 3.6 – Convergence des schémas ImEx et de splitting, adaptés en espace par MRA, sur l'équation de Nagumo pour $k = 10$, $D = 0.1$. Les flux sont évalués au niveau courant (cf. 3.2), la prédiction/reconstruction est assurée par un prédicteur à trois points et l'erreur est comparé à une solution convergé en temps.

Analyse des résultats : pour les grands pas de temps, la convergence est similaire au cas non-adapté (fig. 3.5). En particulier la méthode ImEx232 exhibe une constante de convergence plus faible que le splitting. En revanche, lorsque l'erreur sature les méthodes ImEx offrent systématiquement des performances moins bonnes que le splitting. La solution est comparée à une méthode quasi-exacte en temps, la convergence s'infléchit donc lorsque les erreurs de liée à la MRA sont de l'ordre des erreurs en temps. Il semble donc que l'erreur plateau soit composée d'un terme lié à la compression, au ϵ choisit, et d'une erreur de couplage entre l'adaptation spatiale et la méthode en temps. Plus précisément il apparaît que les méthodes ImEx interagissent avec l'adaptation spatiale d'une manière plus néfaste que le splitting.

3.1.6 Conclusion

Cette première contribution a comparé un schéma de splitting ERK2+IRK2 à deux schémas ImEx ARK sur des questions de stabilités de convergence. L'étude de convergence a été réalisées dans deux contextes différents : sans adaptation spatiale et avec adaptation spatiale par MRA. Les principaux résultats sont :

- ◇ **Stabilité :** Tandis que par nature le splitting découple les problématiques de stabilités, celle des méthodes ARK résulte d'un couplage entre le spectre des deux opérateurs. Cela pourrait être exploité astucieusement puisque dans certains cas (cf. B.iii) un opérateur implicite très raide peut stabiliser la méthode explicite. Ce serait particulièrement intéressant pour des problèmes de diffusion-réaction où une réaction implicite très raide pourrait étendre la stabilité d'une diffusion explicite.
- ◇ **Convergence :**
- ◇ **Couplage avec l'adaptation spatiale :** Il a été monté empiriquement que les méthodes ImEx peuvent interagir de manière néfaste avec l'adaptation spatiale par MRA. Il est probable que les méthodes de splitting interagissent également, mais il semble que cette interaction soit plus limitée. Comme montré au chapitre suivant (chapitre 3.2) ce type d'interaction (méthode en temps - adaptation spatiale) sont complexes et dépendent énormément des schémas temporels et des modalités de mises en oeuvre de la MRA. Donc cette étude ne suffit pas à tirer des conclusions générales, en revanche elle confirme l'existence de tels couplages et montre qu'ils peuvent être importants car (cf 3.1.5) une méthode ImEx meilleure que le splitting sur un schéma non-adapté devient moins bonne que le splitting pour un schéma adapté (dès lors que les erreurs de MRA ne sont plus négligeables).

3.2 Obtention de l'équation équivalente d'une méthode de lignes avec multirésolution adaptative sur un problème de diffusion.

La multirésolution adaptative (MRA) a démontré une grande efficacité expérimentalement. Cependant, son impact sur la qualité des solutions obtenues n'est pas encore totalement compris mathématiquement. En [6], une étude de l'erreur introduite par la multirésolution adaptative a été menée. Cette étude se concentre sur les équations d'advection résolues par des schémas de type *one-step* [11] et compare l'équation équivalente des schémas avec MRA et celle des schémas sans MRA pour mettre en lumière les différences introduites par la multirésolution. La présente étude se place dans la continuité de cette démarche¹⁴ et suit un cheminement similaire pour déterminer, grâce aux équations équivalentes, l'impact de la MRA sur une équation de diffusion résolue par une méthode des lignes d'ordre deux. La différence est donc double : d'une part la nature de l'opérateur étudié est différent (diffusion et non advection), d'autre part le type de schéma utilisés est également différent (ici une méthode des lignes, ne couplant pas les erreurs en espace et en temps est utilisé). D'abord, le problème cible est présenté ainsi que le schéma de référence et le schéma avec AMR. Par la suite, les équations équivalentes du schéma de référence et du schéma avec multirésolution adaptative sont évaluées puis analysées. Enfin, une tentative de confirmation expérimentale est présentée.

14. À un niveau plus modeste.

3.2.1 Cadre de l'étude

A Problème cible

Nous cherchons à résoudre le problème de diffusion suivant :

$$\partial_t u = D \partial_{xx} u. \quad (3.20)$$

Nous ignorons les problématiques de conditions de bords.

B Méthode des lignes utilisée

Pour résoudre cette équation aux dérivées partielles, nous utilisons une méthode des lignes. D'abord un schéma volume fini pour la discrétisation spatiale menant à l'équation semi-discrétisée suivante :

$$dtU(t) = \underbrace{\frac{D}{\Delta x}}_{\text{cellule}} \left(\underbrace{U_{k+1} - 2U_k + U_{k-1}}_{\text{approx. gradients}} \right) \quad (3.21)$$

Puis une méthode de Runge und Kutta explicite d'ordre deux sur l'opérateur linéaire donne :

$$\begin{aligned} U_k^{n+1} = & U_k^n \\ & + D \underbrace{\frac{\Delta t}{\Delta x}}_{\text{cellule}} \left(\underbrace{U_{k+1} - 2U_k + U_{k-1}}_{\text{approx. gradients}} \right) \\ & + \frac{1}{2} D^2 \underbrace{\frac{\Delta t^2}{\Delta x^2}}_{\text{cellule}} \left(\underbrace{U_{k+2} - 4U_{k+1} + 6U_k - 4U_{k-1} + U_{k-2}}_{\text{approx. gradients}} \right). \end{aligned} \quad (3.22)$$

Cela s'écrit sous la forme conservative suivante :

$$u_k^{n+1} = u_k^n + D \frac{\Delta t}{\Delta x} \left(\Phi_{k+1/2}^n - \Phi_{k-1/2}^n \right) + \left(D \frac{\Delta t}{\Delta x} \right)^2 \left(\Psi_{k+1/2}^n - \Psi_{k-1/2}^n \right) \quad (3.23)$$

Avec :

$$\Phi_{k+1/2}^n = \frac{1}{\Delta x} (u_{k+1}^n - u_k^n), \quad (3.24)$$

$$\Phi_{k-1/2}^n = \frac{1}{\Delta x} (u_k^n - u_{k-1}^n), \quad (3.25)$$

$$\Psi_{k+1/2}^n = \frac{1}{\Delta x^2} \left(\frac{1}{2} u_{k+2}^n - \frac{3}{2} u_{k+1}^n + \frac{3}{2} u_k^n - \frac{1}{2} u_{k-1}^n \right),$$

$$\Psi_{k-1/2}^n = \frac{1}{\Delta x^2} \left(\frac{1}{2} u_{k+1}^n - \frac{3}{2} u_k^n + \frac{3}{2} u_{k-1}^n - \frac{1}{2} u_{k-2}^n \right).$$

C La multirésolution adaptative, différents différents paradigmes?

La multirésolution adaptative est présentée avec plus de détails dans le préambule mathématique en 2.4. Pour rappel, MRA consiste à compresser le maillage, puis à effectuer les calculs sur le maillage compressé. Le schéma classique est le suivant :

1. Partir d'un état compressé au pas de temps n .
2. Calculer la solution au pas de temps $n + 1$
3. Compresser de nouveau selon un seuil de compression ε grâce à une transformée multiéchelle et à l'heuristique d'Harten.

Plusieurs stratégies existent pour réaliser le calcul d'un pas de temps à l'autre. La valeur d'une cellule à un niveau de détail donné est approchée au temps pas de temps suivant grâce à un flux numérique évalué soit

- ◊ à partir des cellules voisines au même niveau de détails. C'est la méthode dite *sans reconstruction des flux*. C'est la paradigme classique en MRA.
- ◊ à partir des cellules voisines au niveau le plus fin. C'est la méthode dite *avec reconstruction des flux au niveau le plus fin*. Cette approche n'est pas standard en MRA, elle nécessite une reconstruction grâce à l'opérateur prédiction de la solution au travers de plusieurs niveau. Toutefois il est attendu que cela réduise l'erreur puisqu'elle permet d'évaluer le flux numérique à partir de valeurs plus précises. Pour des problèmes d'advection, ces gains ont été établis théoriquement et validé expérimentalement en [6]. Ce travail réalise une étude similaire pour les problèmes de diffusion.

Cette contribution fournit les équations équivalentes des schémas sans MRA, avec MRA et sans reconstruction des flux et avec MRA et avec reconstruction des flux; à partir de quoi elle analyse les erreurs théoriques portées par chacune de ces approches. Dans la suite, seul est décrit le procédé de MRA avec reconstruction des flux car c'est le plus complexe et par soucis de concision, les calculs sans reconstruction des flux ne sont pas détaillés. **Calcul du flux au travers de Δl niveaux :**

Lorsque l'on applique le procédé de multirésolution, étant donné une cellule à un niveau de détail donné l , on cherche à faire évoluer la valeur à l'étape n vers la valeur à l'étape $n + 1$. Pour ce faire, il faut évaluer les flux à partir des cellules voisines. Dès lors plusieurs choix s'offrent à nous. Où bien on utilise les cellules voisines à leurs niveaux courants, où bien on use de l'opérateur de reconstruction afin d'estimer les cellules voisines à des niveaux plus fins.

Dans un premier temps le stencil est choisi égal à 1. L'opérateur de prédiction d'un niveau à l'autre s'écrit alors :

$$\hat{u}_{2k}^{l+1} = +\frac{1}{8}u_{k-1}^l + u_k^l - \frac{1}{8}u_{k+1}^l, \quad (3.26)$$

$$\hat{u}_{2k+1}^{l+1} = -\frac{1}{8}u_{k-1}^l + u_k^l + \frac{1}{8}u_{k+1}^l. \quad (3.27)$$

Puis en notant $\hat{u}_{(c)}^{l+\Delta l}$ cet opérateur de prédiction itéré au travers de Δl niveaux¹⁵ :

$$\begin{bmatrix} \hat{u}_{2^{\Delta l}k-2}^{(l+\Delta l)} \\ \hat{u}_{2^{\Delta l}k-1}^{(l+\Delta l)} \\ \hat{u}_{2^{\Delta l}k}^{(l+\Delta l)} \\ \hat{u}_{2^{\Delta l}k+1}^{(l+\Delta l)} \end{bmatrix} = \underbrace{\begin{bmatrix} +1/8 & 1 & -1/8 & 0 \\ -1/8 & 1 & +1/8 & 0 \\ 0 & +1/8 & 1 & -1/8 \\ 0 & -1/8 & 1 & +1/8 \end{bmatrix}}_{\text{Matrice de passage } P \text{ pour } s=1.}^{\Delta l} \begin{bmatrix} u_{k-2}^l \\ u_{k-1}^l \\ u_k^l \\ u_{k+1}^l \end{bmatrix} \quad (3.28)$$

C.i Flux calculés au niveau le plus fin. On travaille sur une cellule au niveau courant l (cellules de tailles $\tilde{\Delta x} = 2^{\Delta l} \Delta x$) et lon reconstruit les états au niveau $l + \Delta l$ grâce à des flux au niveau fin, dont les gradients sont approximé par un pas Δx . La mise à jour conservative utilisée est donc

$$u_k^{n+1} = u_k^n + \underbrace{\frac{D \Delta t}{\Delta x 2^{\Delta l}}}_{\text{cellule}} \left(\hat{\Phi}_{k+\frac{1}{2}}^n - \hat{\Phi}_{k-\frac{1}{2}}^n \right) + \left(\underbrace{\frac{D \Delta t}{\Delta x 2^{\Delta l}}}_{\text{cellule}} \right)^2 \left(\hat{\Psi}_{k+\frac{1}{2}}^n - \hat{\Psi}_{k-\frac{1}{2}}^n \right). \quad (3.29)$$

Les flux sont évalués au *niveau fin* (facteurs $1/\Delta x$ et $1/\Delta x^2$ portés par les flux) à partir d'états reconstruits $\hat{u}^{l+\Delta l}$:

$$\hat{\Phi}_{k-\frac{1}{2}}^n = \frac{1}{\Delta x} \left(\hat{u}_{2^{\Delta l}k}^{l+\Delta l} - \hat{u}_{2^{\Delta l}k-1}^{l+\Delta l} \right), \quad (3.30)$$

$$\hat{\Phi}_{k+\frac{1}{2}}^n = \frac{1}{\Delta x} \left(\hat{u}_{2^{\Delta l}(k+1)}^{l+\Delta l} - \hat{u}_{2^{\Delta l}(k+1)-1}^{l+\Delta l} \right), \quad (3.31)$$

$$\hat{\Psi}_{k-\frac{1}{2}}^n = \frac{1}{\Delta x^2} \left(\frac{1}{2} \hat{u}_{2^{\Delta l}k+1}^{l+\Delta l} - \frac{3}{2} \hat{u}_{2^{\Delta l}k}^{l+\Delta l} + \frac{3}{2} \hat{u}_{2^{\Delta l}k-1}^{l+\Delta l} - \frac{1}{2} \hat{u}_{2^{\Delta l}k-2}^{l+\Delta l} \right), \quad (3.32)$$

$$\hat{\Psi}_{k+\frac{1}{2}}^n = \frac{1}{\Delta x^2} \left(\frac{1}{2} \hat{u}_{2^{\Delta l}(k+1)+1}^{l+\Delta l} - \frac{3}{2} \hat{u}_{2^{\Delta l}(k+1)}^{l+\Delta l} + \frac{3}{2} \hat{u}_{2^{\Delta l}(k+1)-1}^{l+\Delta l} - \frac{1}{2} \hat{u}_{2^{\Delta l}(k+1)-2}^{l+\Delta l} \right). \quad (3.33)$$

C.ii Écriture matricielle. Pour simplifier l'implémentation des calculs dans les codes de calculs formel, il est pertinent d'écrire ce qui précède sous forme matricielle.

Pour les flux gauches :

$$\hat{\Phi}_{k-\frac{1}{2}}^n = \frac{1}{\Delta x} \begin{bmatrix} 0 \\ -1 \\ +1 \\ 0 \end{bmatrix}^\top \cdot \begin{bmatrix} \frac{1}{8} & 1 & -\frac{1}{8} & 0 \\ -\frac{1}{8} & 1 & +\frac{1}{8} & 0 \\ 0 & +\frac{1}{8} & 1 & -\frac{1}{8} \\ 0 & -\frac{1}{8} & 1 & +\frac{1}{8} \end{bmatrix}^{\Delta l} \cdot \begin{bmatrix} u_{k-2}^l \\ u_{k-1}^l \\ u_k^l \\ u_{k+1}^l \end{bmatrix}, \quad (3.34)$$

$$\hat{\Psi}_{k-\frac{1}{2}}^n = \frac{1}{\Delta x^2} \begin{bmatrix} -\frac{1}{2} \\ +\frac{3}{2} \\ -\frac{3}{2} \\ +\frac{1}{2} \end{bmatrix}^\top \cdot \begin{bmatrix} \frac{1}{8} & 1 & -\frac{1}{8} & 0 \\ -\frac{1}{8} & 1 & +\frac{1}{8} & 0 \\ 0 & +\frac{1}{8} & 1 & -\frac{1}{8} \\ 0 & -\frac{1}{8} & 1 & +\frac{1}{8} \end{bmatrix}^{\Delta l} \cdot \begin{bmatrix} u_{k-2}^l \\ u_{k-1}^l \\ u_k^l \\ u_{k+1}^l \end{bmatrix}. \quad (3.35)$$

15. Au sens où l'on applique le prédicteur à des données déjà issues d'une prédiction.

Pour les flux droits :

$$\hat{\Phi}_{k+\frac{1}{2}}^n = \frac{1}{\Delta x} \begin{bmatrix} 0 \\ -1 \\ +1 \\ 0 \end{bmatrix}^\top \cdot \begin{bmatrix} \frac{1}{8} & 1 & -\frac{1}{8} & 0 \\ -\frac{1}{8} & 1 & +\frac{1}{8} & 0 \\ 0 & +\frac{1}{8} & 1 & -\frac{1}{8} \\ 0 & -\frac{1}{8} & 1 & +\frac{1}{8} \end{bmatrix}^{\Delta l} \cdot \begin{bmatrix} u_{k-1}^l \\ u_k^l \\ u_{k+1}^l \\ u_{k+2}^l \end{bmatrix}, \quad (3.36)$$

$$\hat{\Psi}_{k+\frac{1}{2}}^n = \frac{1}{\Delta x^2} \begin{bmatrix} -\frac{1}{2} \\ +\frac{3}{2} \\ -\frac{3}{2} \\ +\frac{1}{2} \end{bmatrix}^\top \cdot \begin{bmatrix} \frac{1}{8} & 1 & -\frac{1}{8} & 0 \\ -\frac{1}{8} & 1 & +\frac{1}{8} & 0 \\ 0 & +\frac{1}{8} & 1 & -\frac{1}{8} \\ 0 & -\frac{1}{8} & 1 & +\frac{1}{8} \end{bmatrix}^{\Delta l} \cdot \begin{bmatrix} u_{k-1}^l \\ u_k^l \\ u_{k+1}^l \\ u_{k+2}^l \end{bmatrix}. \quad (3.37)$$

3.2.2 Les équations équivalentes

A Calcul des équations équivalentes

Tout les calculs ont été réalisés grâce à la bibliothèque de calcul formel Sympy et les codes sont disponibles à l'adresse : https://github.com/Ocelot-Pale/etude_MR_RK2.

A.i Équation équivalente du schéma de référence (sans multirésolution adaptative) Le calcul de l'équation équivalente sans MRA donne :

$$\frac{\partial u}{\partial t} = D \frac{\partial^2 u}{\partial x^2} + \Delta x^2 \frac{D}{12} \frac{\partial^4 u}{\partial x^4} - \Delta t^2 \frac{D^3}{6} \frac{\partial^6 u}{\partial x^6} - \Delta t^3 \frac{D^4}{24} \frac{\partial^8 u}{\partial x^8}. \quad (3.38)$$

Le schéma de référence est donc bien d'ordre deux en espace et en temps.

A.ii Équation équivalente avec multirésolution adaptative et sans reconstruction des flux Lorsque la multirésolution adaptative usuelle (sans reconstruction des flux) est introduite dans le schéma, en supposant une adaptation de Δl niveaux de détails, l'équation équivalente devient :

$$\frac{\partial}{\partial t} u = D \frac{\partial^2 u}{\partial x^2} + (2^{\Delta l} \Delta x)^2 \frac{D}{12} \frac{\partial^4 u}{\partial x^4} - \Delta t^2 \frac{D^3}{6} \frac{\partial^6 u}{\partial x^6} - \Delta t^3 \frac{D^4}{24} \frac{\partial^8 u}{\partial x^8} \quad (3.39)$$

En somme, sans reconstruction des flux, le schéma avec MRA se comporte sans surprises comme le schéma de référence mais sur un maillage plus grossier. La constante d'erreur en espace est alors de l'ordre de $2^{2\Delta l} \frac{D}{12} \frac{\partial^4 u}{\partial x^4}$ au lieu de $\frac{D}{12} \frac{\partial^4 u}{\partial x^4}$.

A.iii Équation équivalente avec multirésolution adaptative et avec reconstruction des flux En évaluant les flux à partir des cellules reconstruites au niveau le plus fin, l'équation équivalente est :

$$\begin{aligned} \frac{\partial u}{\partial t} = & D \frac{\partial^2 u}{\partial x^2} \\ & - \frac{\Delta t}{2} D^2 (2^{2\Delta l} - 1) \frac{\partial^4 u}{\partial x^4} - \Delta t^2 \frac{D^3}{6} \frac{\partial^6 u}{\partial x^6} - \Delta t^3 \frac{D^4}{24} \frac{\partial^8 u}{\partial x^8} \\ & + 2^{2\Delta l} \frac{D \Delta x^2}{12} \frac{\partial^4 u}{\partial x^4} - 2^{2\Delta l} \frac{D \Delta l \Delta x^2}{4} \frac{\partial^4 u}{\partial x^4} \end{aligned} \quad (3.40)$$

Ce schéma est d'ordre un en temps contrastant avec l'ordre deux des deux schémas précédents. Ainsi, théoriquement la reconstruction au plus fin des flux réduit l'ordre de convergence temporelle de la méthode des lignes. Cependant en pratique le caractère explicite de la méthode ERK2 impose la contrainte de stabilité $\lambda = \frac{D\Delta t}{\Delta x^2} < 1/2$ masquant la perte d'ordre puisque cela mène à l'élution équivalente :

$$\begin{aligned} \frac{\partial u}{\partial t} = & D \frac{\partial^2 u}{\partial x^2} \\ & + \Delta x^2 \left(\frac{2^{2\Delta l} D \lambda \frac{\partial^4 u}{\partial x^4}}{2} - \frac{2^{2\Delta l} D \Delta l \frac{\partial^4 u}{\partial x^4}}{4} - \frac{D \lambda \frac{\partial^4 u}{\partial x^4}}{2} + \frac{2^{2\Delta l} D \frac{\partial^4 u}{\partial x^4}}{12} \right) \\ & - \Delta x^6 \frac{D \lambda^3 \frac{\partial^8 u}{\partial x^8}}{24} - \Delta x^4 \frac{D \lambda^2 \frac{\partial^6 u}{\partial x^6}}{6} \end{aligned} \quad (3.41)$$

B Comparaison

Pour comprendre le mécanisme menant à cette perte d'ordre, l'équation équivalente du schéma avec AMR et reconstruction des flux est comparée à celle du schéma de référence, **avant** d'appliquer la procédure de Cauchy-Kovaleskaya ; c'est à dire sans exploiter $\partial_t u = D \partial_{xx} u$.

B.i Sans multirésolution (schéma de référence) L'équation modifiée sans multirésolution, avant procédure de Cauchy-Kovaleskaya est :

$$\begin{aligned} \frac{\partial u}{\partial t} = & D \frac{\partial^2 u}{\partial x^2} \\ & + \frac{1}{2} \left(D^2 \frac{\partial^4 u}{\partial x^4} - \frac{\partial^2 u}{\partial t^2} \right) \Delta t + \frac{D}{12} \frac{\partial^4 u}{\partial x^4} \Delta x^2 - \frac{1}{24} \frac{\partial^4 u}{\partial t^4} \Delta t^3 - \frac{1}{6} \frac{\partial^3 u}{\partial t^3} \Delta t^2. \end{aligned} \quad (3.42)$$

Se compense par
la procédure de
Cauchy-Kovaleskaya

La méthode est bien d'ordre un, car à l'ordre un : $\frac{\partial u}{\partial t} = D \frac{\partial^2 u}{\partial x^2}$ et donc le terme $D^2 \frac{\partial^4 u}{\partial x^4} - \frac{\partial^2 u}{\partial t^2}$ se compense au cours de la procédure de Cauchy-Kovaleskaya.

B.ii multirésolution avec reconstruction des flux L'équation modifiée avec multirésolution et reconstruction des flux, sans appliquer procédure de Cauchy-Kovaleskaya est :

$$\begin{aligned} \frac{\partial u}{\partial t} = & D \frac{\partial^2 u}{\partial x^2} \\ & + \frac{\Delta t}{2} \left(2^{2\Delta l} D^2 \frac{\partial^4 u}{\partial x^4} - \frac{\partial^2 u}{\partial t^2} \right) - \frac{\Delta t^3}{24} \frac{\partial^4 u}{\partial t^4} - \frac{\Delta t^2}{6} \frac{\partial^3 u}{\partial t^3} + \frac{\Delta x^2}{12} (1 - 3\Delta l) 2^{2\Delta l} D \frac{\partial^4 u}{\partial x^4} \end{aligned} \quad (3.43)$$

Ne se compensent plus.

Dans ce cas le terme en facteur du Δt ne se s'annule plus. En effet le terme $D^2 \frac{\partial^4 u}{\partial x^4}$ est devenu au cours de la reconstruction $2^{2\Delta l} D^2 \frac{\partial^4 u}{\partial x^4}$. En conséquence, la méthode perd un ordre de convergence temporel.

Ce mécanisme s'explique de la manière suivante : dans l'équation équivalente, le terme $\frac{\partial^2 u}{\partial t^2}$ apparaît indépendamment de la discrétisation spatiale ¹⁶. La méthode des lignes initiale crée un terme *sur*

16. Il emerge de la différence $u_k^{n+1} - u_k^n$ à k fixé.

mesure pour le compenser en approximant le terme spatial $D^2 \frac{\partial^4 u}{\partial x^4}$. Cependant au cours du processus de compression-reconstruction, cette approximation est entachée d'un facteur $2^{2\Delta l}$. En d'autres termes le terme spatial construit pour compenser un terme temporel a été modifié par la multi-résolution, alors que le terme temporel lui n'est pas affecté par la multirésolution. Ainsi, les deux termes ne se compensent plus et l'ordre est perdu.

C Conclusion sur le résultat obtenu grâce aux équations équivalentes

Il a été ici mis en lumière que la reconstruction des flux au plus fin sur appliquée à méthode des lignes très simple peut théoriquement mener à un couplage des erreurs espace-temps polluant l'ordre initial de la méthode. Alors que cela n'arrive pas lorsque les flux sont évaluées plus grossièrement. En particulier l'étape de reconstruction-reconstruction altère des termes spatiaux qui ne compensent plus certaines erreurs temporelles et perturbent l'ordre de la méthode Runge et Kutta.

3.2.3 Complément expérimental

A Présentation du cas test

Pour tâcher d'observer la perte d'ordre des expériences numériques ont été menées grâce au logiciel Samurai. Les expériences ont portées sur la simulation de l'équation de diffusion en 1D avec une solution initiale en forme de courbe de Gauß avec conditions de Dirichlet homogènes au bords. Pour limiter les effets de bords (qui non pris en compte dans l'analyse précédente) le domaine à été pris *grand* devant la largeur de la gaussienne initiale et le temps final assez petit pour que la diffusion n'atteigne pas le bord (qualitativement). L'erreur à été calculée comme la norme L^2 de l'erreur au temps final centrée sur la gaussienne¹⁷. Ce cas test est pertinent car il offre une solution lisse mais avec des gradients (donc l'AMR doit compresser à divers endroits) tout en offrant une solution analytique connue pour comparer l'erreur. En effet le problème posé sur $\mathbb{R}^+ \times \mathbb{R}$ (sans conditions de bord) :

$$\begin{aligned} \partial_t u &= \partial_{xx} u, \\ u(t=0, x) &= \frac{1}{\sqrt{4\pi a}} \exp\left(-\frac{x^2}{4a}\right). \end{aligned} \quad (3.44)$$

Admet pour solution :

$$u(t=0, x) = \frac{1}{\sqrt{4\pi a(1+t)}} \exp\left(-\frac{x^2}{4a(1+t)}\right). \quad (3.45)$$

La solution numérique avec conditions de bord a été comparée avec la solution analytique sans conditions de bord, au vu de la taille du domaine et des temps de simulation cela ne devrait pas interférer de manière mesurable.

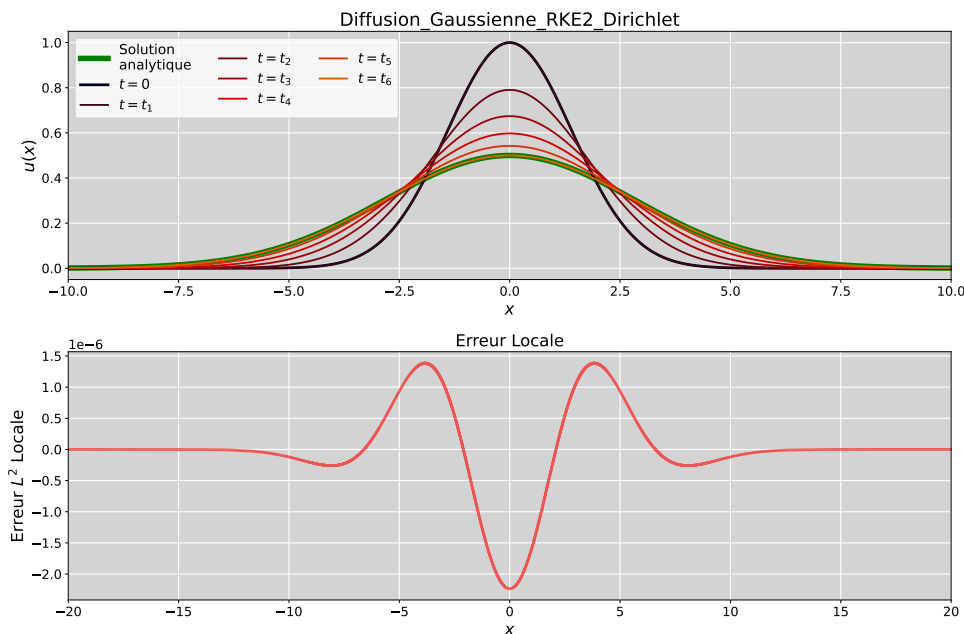


FIGURE 3.7 – Illustration d'une simulation du cas test 3.44 avec conditions de Dirichlet homogène et affichage de l'erreur locale au temps final.

17. On calcul l'erreur autour de la gaussienne et pas sur tout le domaine car sinon elle serait artificiellement faible puisque la solution est quasi-nulle sur le reste du domaine qui a été pris grand pour éviter les effets de bord.

B Des défis expérimentaux

L'observation du mécanisme de perte d'ordre mis théoriquement à jour précédemment est une tâche ardue.

B.i Une expérience exacte impossible à reproduire Il n'est pas possible d'utiliser la méthode numérique utilisée dans l'étude théorique pour essayer de la valider expérimentalement. En effet, la méthode est une RK2 explicite, elle impose sur ce problème de diffusion une condition de stabilité du type $\Delta t \propto \Delta x^2$, ce qui en pratique donne $\Delta t \ll \Delta x$. De fait, la majorité des erreurs sont liés au pas d'espace "grand" devant le pas de temps, et donc si l'on fixe le pas d'espace pour faire converger la méthode en temps, l'erreur est déjà saturée en temps et on n'observe rien. C'est un classique de l'analyse numérique.

Erreur L^2 en fonction de Δt

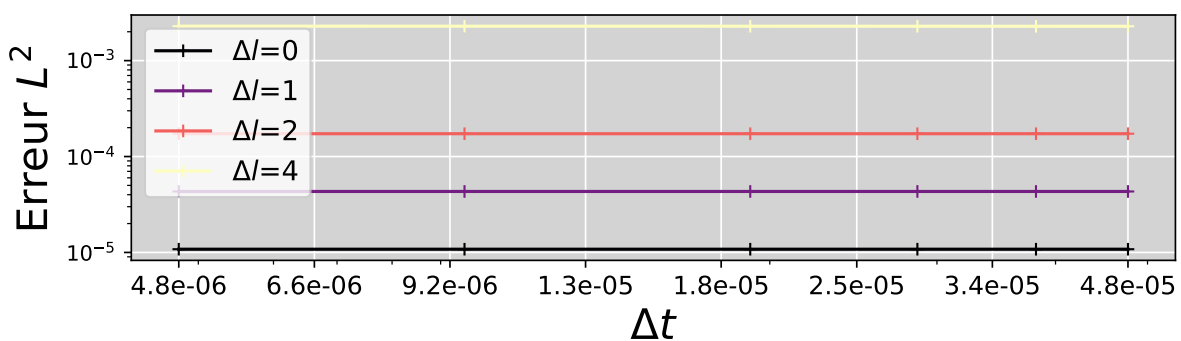


FIGURE 3.8 – Saturation de la convergence temporelle avec une méthode RKE2 sur l'équation de diffusion. L'erreur L^2 stagne malgré la diminution du pas de temps, illustrant la domination de l'erreur spatiale due à la contrainte CFL $\Delta t \propto \Delta x^2$.

B.ii Une tentative infructueuse Suite à cette limite, l'expérience a été retentée avec une méthode voisine mais sans contrainte de stabilité. Le code de calcul Samurai a donc été relancé avec une méthode RK implicite d'ordre deux, plus précisément une SDIRK. Cependant, la reconstruction des flux au niveau le plus fin est difficile à mettre en oeuvre pour les méthodes implicites et donc le schéma n'a été éprouvé qu'avec la MRA classique. Dans ce cas l'ordre deux en temps est bien maintenu (voir fig. 3.9), comme ce que prévoyais l'équation équivalente pour la ERK2. Toutefois cela ne dit rien du comportement avec reconstruction au niveau le plus fin.

B.iii Des biais multiples D'autres biais expérimentaux peuvent expliquer l'invisibilité du phénomène. Par exemple l'étude précédente ne prend pas en compte les conditions de bord. Une autre hypothèse est peut être que le maillage n'est compressé que localement ce qui n'altère que peut être pas la convergence globale. Enfin, dans les calculs théoriques précédents, il a été fait l'hypothèse que l'évaluation est faite au niveau le plus fin (que la solution est entièrement reconstruite pour l'évaluation des flux), ce qui n'est pas fait en pratique. Cette idée vient du fait qu'intuitivement, si l'on reconstruit jusqu'au niveau le plus fin les termes servant dans le calcul des flux, alors l'erreur devrait

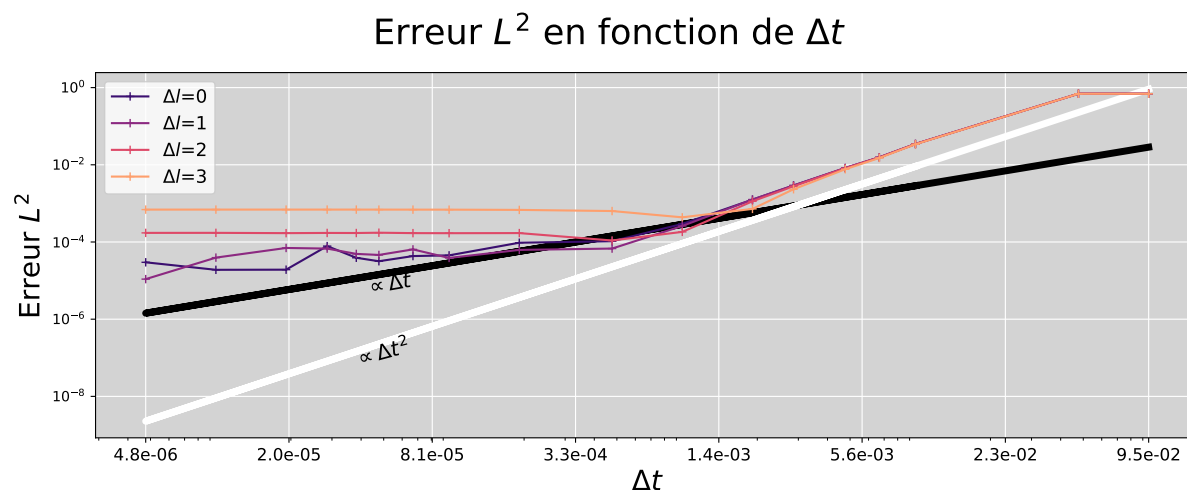


FIGURE 3.9 – Convergence temporelle d'ordre 2 avec une méthode SDIRK-RK2 sur l'équation de diffusion pour différentes profondeurs de maillages adaptés. L'ordre théorique est préservé comme attendu, puisque les flux sont évalués au niveau courants.

diminuer; c'est ce que suggère [6]. Cependant cette fonctionnalité n'étant pas encore disponible dans le logiciel de calcul, l'expérience a été réalisée sans reconstruire les flux au niveau le plus fin mais en prenant la valeur disponible au niveau courant de compression. Il semble peu probable que ce soit la cause de la non-observation du phénomène de perte d'ordre mais cela reste un biais potentiel. Enfin peut être qu'un bug s'est glissé dans mon implémentation mais cela semble peu probable puisque ce serait une erreur d'implémentation qu'il "améliore" l'ordre de convergence...

3.2.4 Conclusion

En conclusion, en résolvant le problème de diffusion grâce à la méthode des lignes proposée, la multi-résolution adaptative usuelle (sans reconstruction des flux) préserve l'ordre du schéma. En revanche et contre toute attente, lorsque les flux sont reconstruits au niveau le plus fin, l'ordre deux en temps du semble formellement réduit à un. Malheureusement ce phénomène n'a pas pu être mis en lumière expérimentalement, notamment à cause de la contrainte de stabilité. Face à ces difficultés des expériences numériques plus ambitieuses ont été entreprises (méthodes stabilisées, différents niveaux de reconstruction, extension à d'autres cas que la diffusion "pure" etc...). C'est l'objet de la contribution suivante qui étudie empiriquement l'impact de la reconstruction ou non du flux sur les problèmes de diffusion puis de diffusion-réaction.

3.3 Impact de la qualité de reconstruction des flux pour les problèmes diffusion avec AMR.

Cette troisième contribution s'inscrit directement dans la continuité de la précédente. L'objectif est ici d'étudier expérimentalement l'impact de la multirésolution et de ses modalités de mise en oeuvre sur les approximations numériques des problèmes diffusifs. Elle propose trois algorithmes d'AMR se distinguant par la qualité de l'évolution des flux numériques.

D'abord les différences entre les trois algorithmes d'AMR sont mises en lumière, puis une première expérience numérique reprenant le schéma numérique de la contribution précédente (VF + ERK2) est réalisée. Face à des résultats surprenants, l'hypothèse de l'émergence d'instabilités résiduelles pour certains algorithmes a été émise. Cependant cette hypothèse semble être invalidée par une étude de stabilité linéaire réalisée grâce à Sympy. La contrainte de stabilités imposées par la méthode ERK2 n'avait jusqu'ici permises d'observer que des solutions convergées en temps (erreur spatiale dominante). Restait inconnu l'impact de la méthode d'évaluation des flux numériques dans un contexte où les erreurs temporelles restent dominantes. Une seconde expérience a alors été réalisée avec la méthode explicite stabilité ROK4 au lieu de la méthode ERK2, permettant d'accéder à des pas de temps plus grands.

3.3.1 Présentation des trois algorithmes

La différence entre les trois algorithmes étudiés réside dans la manière de conjuguer les volumes finis [26] et la multirésolution adaptative. Le calcul des *flux*, central dans les méthodes volumes finis, requiert l'évaluation de termes dépendant de la solution aux *interfaces* amont et aval des cellules. Les volumes finis n'approximant que les valeurs moyennes sur les cellules et non les valeurs ponctuelles aux interfaces, les termes de flux sont alors évalués comme fonction des valeurs moyennes sur les cellules voisines de l'interface.

La MRA rend la définition des flux numériques non-univoque - c'est ce qui est ici étudié.

Comme la MRA définit plusieurs grilles de pas $\Delta x, 2\Delta x, 4\Delta x, \dots$ il est ambigu de choisir sur quelle grille évaluer les voisins de chaque interface. En effet, à niveau de détail l fixé, le flux numérique doit-il être déterminé à partir des cellules voisines du niveau $l, l+1, l+2, \dots$ (voir le schéma en fig. 3.10)? Le premier algorithme étudié (la référence en MRA), consiste à évaluer les flux à partir des voisins du même niveau que la cellule étudiée. C'est à dire que si flux concerne une cellule de la grille de niveau l , les voisins de l'interface sont choisis également au niveau l . Cela revient à résoudre localement l'EDP au niveau courant de la grille. Cet algorithme est la norme en MRA car ne requiert aucun calcul supplémentaire, les valeurs sur la grille au niveau l sont directement accessibles. Le second algorithme consiste à systématiquement choisir les cellules de la grille la plus fine. Intuitivement c'est le plus précis, mais plus coûteux car la grille la plus fine n'est pas directement accessible (voir la partie sur la reconstruction en 2.4). Enfin le troisième algorithme est un compromis entre les deux approches précédentes. Il consiste à calculer les flux à partir des valeurs d'un niveau en deçà du niveau courant, pour gagner un peu en précision sans pour autant s'exposer à des coûts computationnels prohibitifs. En [6], la différence théorique entre les deux premiers algorithmes a été étudiée sur des problèmes d'advection linéaires et une comparaison expérimentale entre les trois algorithmes a été réalisée sur des problèmes d'advections linéaires et non-linéaires.

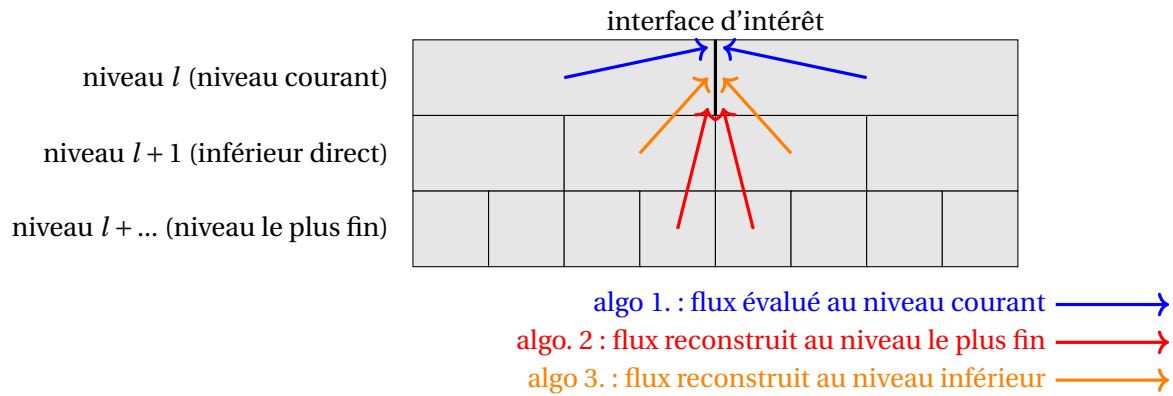


FIGURE 3.10 – Illustration des trois algorithmes évalués. L’algorithme 1 en bleu calcule le flux à partir des cellules de la grille au même niveau que l’interface étudiée. L’algorithme 2 reconstruit les valeurs de la solution sur la grille de niveau inférieur. L’algorithme 3 reconstruit au niveau le plus fin possible. Plus l’algorithme reconstruit finement, plus les valeurs moyennes sont données sur des cellules petites et plus cela s’approche d’une valeur "ponctuelle".

3.3.2 Expérience numérique avec une méthode Runge et Kutta explicite

La première expérience numérique porte sur l’équation de diffusion résolue par le même schéma numérique que dans l’étude théorique précédente (cf : 3.2). C’est à dire une discrétisation spatiale d’ordre deux du Laplacien intégré en temps par une méthode Runge et Kutta explicite d’ordre 2. Une méthode implicite pourrait paraître plus appropriée pour s’affranchir des problèmes de stabilité ; cependant l’inversion d’un système linéaire couplé à la reconstruction des flux à des niveaux plus fin (non-standard, algos 2 et 3) est difficile à implémenter informatiquement et aurait ralenti l’étude. À cause de la contrainte de stabilité $\Delta t \propto \Delta x^2$, seules des solutions convergées en temps (erreur spatiale dominante puisque $\Delta t \ll \Delta x$) sont observées. Cette contrainte est levée dans l’expérience suivante en 3.3.4.

A Résultats numériques

Les résultats sont étonnants, l’algorithme 1 (le plus grossier) offre la plus faible erreur et plus l’algorithme reconstruit finement les flux plus l’erreur augmente. À titre d’exemple, pour un état initial en forme de courbe de Gauss, une seuil de compression $\varepsilon = 10^{-4}$ et un maillage présentant jusqu’à 4 niveau de finesse, les erreurs L^2 au temps final $T_f = 1$ sont les suivantes :

Algorithme n^o	Niveau d’évaluation des flux	Erreur L^2
1	Courant	1×10^{-4}
2	Inférieur direct	2×10^{-4}
3	Plus fin	3×10^{-4}
référence	Sans AMR	2×10^{-5}

Bien sûr, l’algorithme sans AMR reste celui avec la plus petite erreur.

B Analyse et hypothèses

Ce résultat est assez surprenant puisqu'on s'attendrait à ce qu'une reconstruction plus précise du flux donne de meilleurs résultats. Face à ces résultats, plusieurs hypothèses sont émises :

1. La reconstruction introduit des *instabilités* dans le schéma.
2. L'usage de cellules plus grandes pour évaluer les flux augmente le caractère diffusif du schéma et réduit "accidentellement" l'erreur.
3. Peut-être que cette tendance ne vaut que sur des solutions où l'erreur spatiale domine et que les résultats seraient différents lorsque l'erreur temporelle reste dominante.

La première hypothèse est éprouvée dans la prochaine section [3.3.3](#) et une méthode stabilisée est utilisée en ?? pour explorer numériquement la troisième hypothèse.

3.3.3 Analyse de stabilité

Cette partie étudie la stabilité des différents algorithmes d'AMR à l'aide d'une analyse type Fourier - Von Neumann. Ces résultats ont été obtenus en adaptant le code de calcul formel ayant fourni les équations équivalentes en 3.2. Une petite interface graphique est disponible à cette adresse https://github.com/OcelotPala/etude_MR_RK2/blob/main/code_2/stabilite_AMR_plotly.html A FINIR ...

A Conclusion

Il semble que le résultat numérique surprenant précédemment présenté de vienne pas de problèmes de stabilités liées aux reconstructions, en effet si ces problèmes de stabilités apparaissaient, ils devraient être plus prononcés sur la méthode où la reconstruction se fait au niveau courant. Alors que c'est l'inverse qui est observée.

3.3.4 Expérience numérique avec une méthode explicite stabilisée

Pour observer ce qui se produit lorsque l'erreur n'est pas saturée en espace mais que l'erreur temporelle intervient également, le logiciel Ponio¹⁸ a été couplé à Samurai. Il permet d'utiliser facilement des méthodes d'intégration en temps complexe. Grâce Ponio, l'expérience précédente a été réitérée en remplaçant la méthode ERK2 par la méthode stabilisée ROCK4 [3]. Cette méthode reste explicite (ce qui permet grâce à Samurai d'étudier facilement les différentes façons d'évaluer les flux) tout en assouplissant significativement la contrainte de stabilité.

A Résultats numériques

Les résultats sont présentés en fig. 3.11. Chaque graphique correspond à un paramétrage différent de l'MRA. Les solutions sont comparées à une solution convergée en temps afin d'isoler les erreurs temporelles et celles liées à la MRA de l'erreur spatiale. La grille la plus fine du maillage de niveau 12, ce qui correspond à 2^{12} cellules et le maillage peut être représenté sur 6 niveaux, donc du niveau 12 au niveau $12 - 6 = 6$. Le graphique de droite correspond à un seuil de compression modéré ($\varepsilon = 10^{-5}$) et celui de gauche à un seuil de compression plus restrictif ($\varepsilon = 10^{-6}$). Sur chaque graphique l'erreur L^2 est tracée en fonction du pas de temps pour chaque méthode de calcul du flux numérique : sans MRA (grille fine) ; **MRA niveau courant** ; **reconstruction d'un niveau** ; **reconstruction au niveau le plus fin**. Les principales observations sont :

1. Pour un petit pas de temps : plus les flux sont reconstruit finement plus l'erreur augmente, ce qui est étonnant mais cohérent avec l'expérience numérique précédente.
2. Pour un grand pas de temps : la méthode d'évaluation du flux importe peu. Les erreurs en temps dominent les erreurs liées à la MRA.
3. Il existe une gamme de pas de temps intermédiaires où les méthodes reconstruisant les flux sur-performent la MRA classique **et** (étonnant!) la méthode sur grille fine.

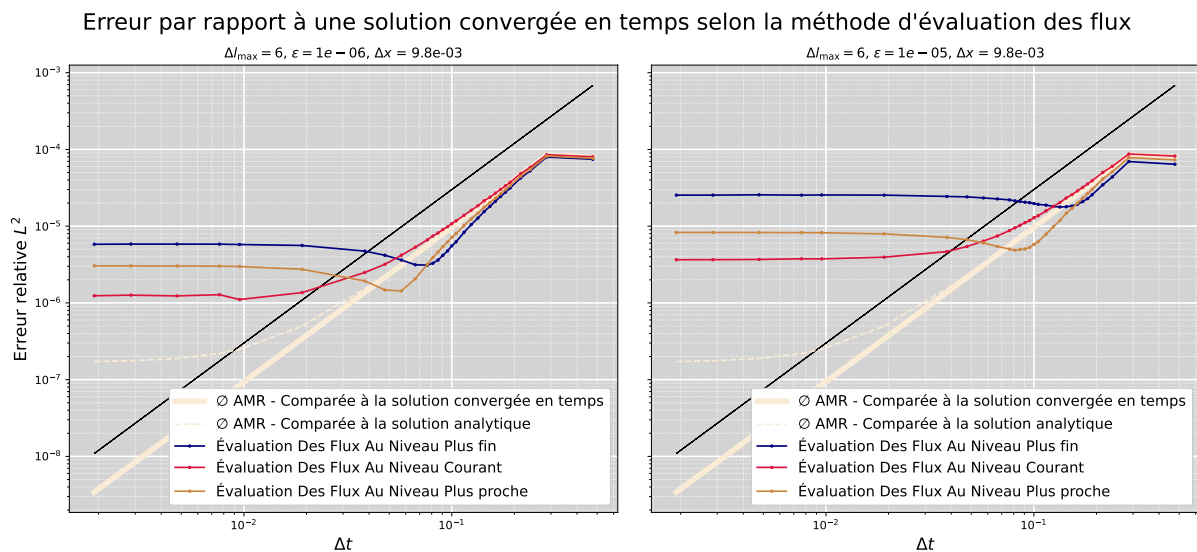


FIGURE 3.11 – Convergence pour les différentes méthodes numériques.

18. <https://github.com/hpc-maths/ponio>

Afin d'éclairer ces résultats surprenants, la figure 3.12 présente le profil des erreurs pour les différents régimes de pas de temps : petit, intermédiaire, grand.

Pour la méthode sur grille fine et la méthode MRA classique : le profil présente une *cloche centrale* qui "s'écrase" au fur et à mesure que le pas de temps se réduit, de fait l'erreur chute avec le pas de temps.

En revanche pour les méthodes impliquant une reconstruction : le profil présente également une *cloche centrale* mais au lieu de s'écraser, elle change progressivement de convexité et de signe. Elle est d'abord positive convexe pour les grands pas de temps puis concave négative pour les petits pas de temps. Pour un régime intermédiaire, la cloche est aplatie en zéro ce qui provoque pour ces pas de temps, la chute brutale de l'erreur.

Ce comportement très intéressant - le changement de signe de la cloche - résulte du couplage entre les erreurs en temps du schéma et les erreurs de la reconstruction de la MRA. Dans la suite, nous allons relier cette observations au équations équivalentes précédemment établies.

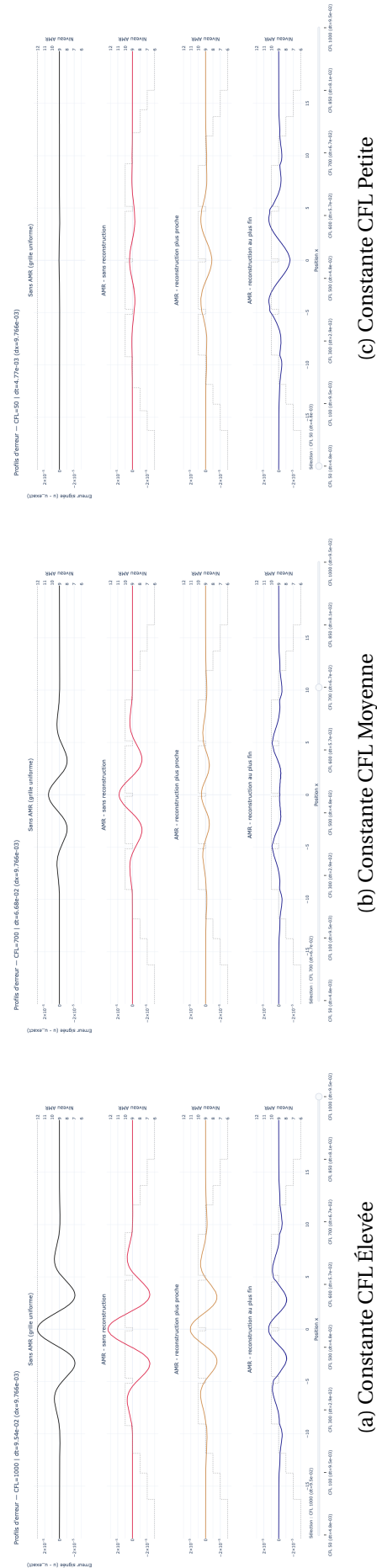


FIGURE 3.12 – Profils d'erreur pour différentes valeurs de la constante CFL. C'est le pas d'espace est fixé, cela revient simplement à changer le pas de temps.

3.3.5 Lien avec les équations équivalentes

Pour comprendre les résultats de convergence précédents (3.3.4), le profil des erreurs a été tracé (voir 3.12) pour chaque méthode numérique et pour différents régimes de CFL; puis ces observations ont été reliées aux équations équivalentes développées en 3.2. Cette mise en relation est cohérente car les équations équivalentes ont été calculées pour un schéma d'intégration en temps ERK2 et l'expérience a été réalisée avec ROCK2 qui est une ERK2 stabilisée.

Les résultats précédents ont d'abord été confirmés :

1. Pour les petites CFL, les méthodes avec reconstruction sont moins précisées que les méthodes sans reconstruction.
2. Il existe une gamme de CFL très précise pour laquelle les schémas avec reconstruction surperforment les autres méthodes.

Une première hypothèse a été formulée mais elle a été par la suite invalidée. Une seconde hypothèse a alors été proposée qui a été validée.

A Première Hypothèse

L'observation suivante a de plus été faite : *il semble que le profil de l'erreur avec reconstruction ressemble à la dérivée spatiale d'ordre quatre de la solution et que le signe de cette erreur change avec la CFL.*

Une première hypothèse a d'abord été proposée : *l'évolution du profil d'erreur selon la CFL s'explique par le terme ... dans l'équation équivalente ??.* Pour les grandes CFL le coefficient ... est ... et pour les petites CFL il est Enfin pour $\lambda = \dots$ ce coefficient serait null expliquant chute de l'erreur pour ces CFL, l'erreur s'allégeant du terme d'ordre deux.

Validation expérimentale : Pour valider cette hypothèse, une régression linéaire entre l'erreur numérique et la dérivée 4^e de la solution a été réalisée par une méthode des moindres carrées :

$$\min_{\alpha \in \mathbb{R}} \|\alpha \partial_x^4 u - \text{err}\|^2.$$

Ce modèle (fig. 3.13) explique bien l'erreur pour les petites CFL ($R^2 > 0.9$) mais mal pour les grandes CFL ($R^2 \sim 0.2$).

B Seconde Hypothèse

Une seconde observation a alors complété la première : *à grande CFL, l'erreur ne ressemble pas à l'opposé de $\partial_x^4 u$ mais à $\partial_x^6 u$.*

Une seconde hypothèse a alors été émise : *l'erreur s'explique aux grandes CFL car dans l'équation équivalente le terme $\lambda^2 \Delta x^4 \partial_x^6 u$ domine et aux petites CFL car le terme $\lambda \delta x^2 \partial_x^4 u$ domine.* Les piètres performances de la méthode avec reconstruction pour de petites CFL s'explique alors simplement, il y a plus de coefficients devant le terme d'erreur $\Delta x^2 \partial_x^4 u$, dominant pour les petites CFL :

$$\cdot \tag{3.46}$$

$$\cdot \tag{3.47}$$

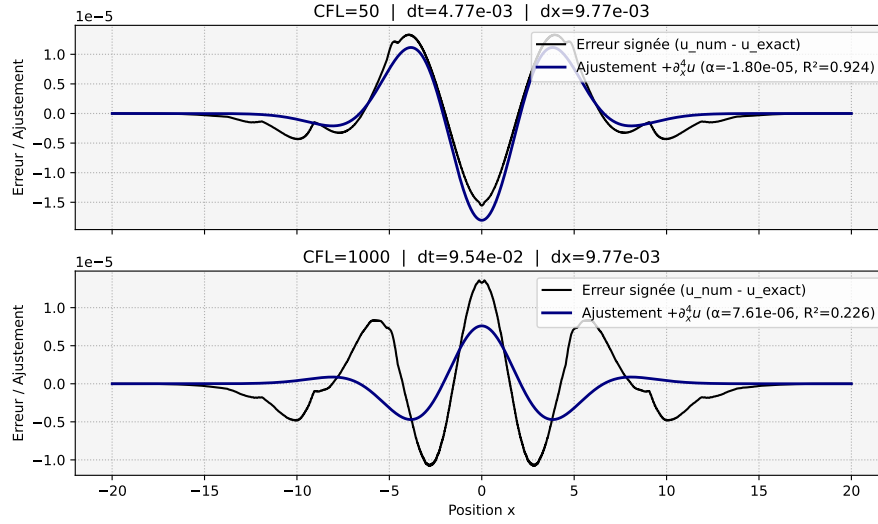


FIGURE 3.13 – Régression entre l'erreur numérique expérimentale (AMR + reconstruction fine) et la dérivées 4^e de la solution.

Pour valider cette nouvelle hypothèse, l'erreurs a été modélisée comme $\text{err} \approx \alpha \partial_x^4 u + \beta \partial_x^6 u$ par moindre carré. Ce modèle explique très bien l'erreur pour tous les régimes de CFL (voir 3.14).

Correspondance Erreur Num. $\propto \alpha \partial_x^4 u + \beta \partial_x^6 u$

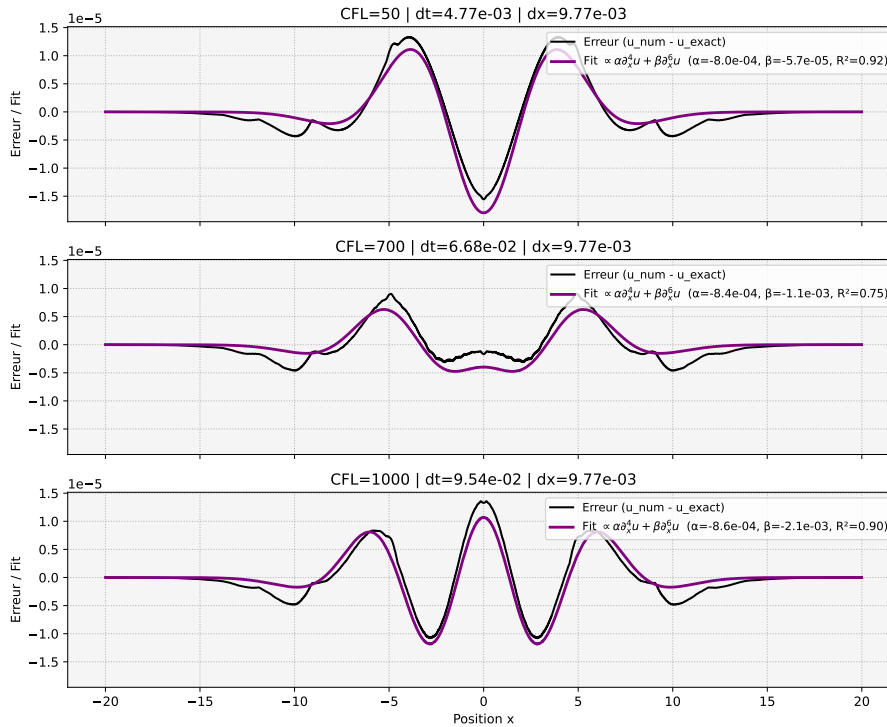


FIGURE 3.14 – Régression entre l'erreur numérique expérimentale (AMR + reconstruction fine) et une combinaison linéaire des dérivées 4^e et 6^e de la solution.

Remarque : la plage de CFL où la méthode avec reconstruction est plus la précise correspond en fait au cas où les profils de $\alpha \partial_x^4 u$ de $\beta \partial_x^6 u$ se compensent. C'est donc un comportement tout à fait acci-

dentel lié au fait que les dérivées de la courbe de Gauss sont en quelque sort "en opposition de phase".

Analyse : avec le recul, les résultats s'expliquent comme suit. Le schéma d'intégration en temps nécessite l'approximation de la dérivée spatiale d'ordre quatre de la solution. Ce qui ne peut être fait de manière satisfaisante à partir d'une prédiction polynomiale à trois point [**<empty citation>**]/ ??.

Complément : Ce n'est pas présenté ici, mais le travail a été refait avec un prédicteur à 5 points, permettant d'approximer correctement la dérivée d'ordre quatre. Alors la reconstruction apporte un gain notable, les solutions numériques avec et sans MRA sont dans ce cas très proches.

3.3.6 Extension sur une équation de diffusion-réaction

Cette expérience permet d'observer l'impact du niveau de reconstruction des flux quand l'opérateur de diffusion est couplé à un opérateur de diffusion et faisant émerger des dynamiques de couplage. L'équation de diffusion est donc remplacée par l'équation de Nagumo (voir 3.1.1). Des ondes progressives sont solution de cette équation, le schéma numérique donc doit être en mesure de suivre la dynamique du front d'onde.

A Résumé de l'expérience

L'expérience a été réalisée dans des conditions similaires à l'expérience numérique 3.1. C'est à dire :

- ◇ Profil de l'onde propagative comme état initial.
- ◇ Domaine étendu avec conditions de Neumann aux limites pour limiter les effets de bord

L'unique différence majeur est le remplacement de la méthode Runge et Kutta ImEx par un schéma de séparation d'opérateur avec une méthode stabilisée explicite pour la diffusion (ROCK 2 [3]). Ce choix découle, comme précédemment expliqué, de la nécessité d'éviter toute inversion de système linéaire lorsque les flux sont reconstruits à partir de reconstructions fines.

B Résultats de l'expérience

Les résultats de l'expérience sont donnés en figure 3.15, selon le pas de temps les observation sont :

- ◇ **Pas de temps "petits" - erreur spatiale dominante :** L'erreur se comporte pareillement au cas de la diffusion pure, plus les flux sont reconstruit finement, plus l'erreur augmente.
- ◇ **Pas de temps "grands" - erreur temporelle dominante :** L'erreur se comporte encore comme dans le cas de la diffusion pure, la méthode de reconstruction n'induite pas de variations sensibles de l'erreur.
- ◇ **Pas de temps 'intermédiaires" - erreur spatio-temporelle :** C'est l'observation la plus intéressante : plus les flux sont finement évalués, plus l'erreur chute brutalement. Comme si *la reconstruction fine des flux de diffusion prévenait un couplage des erreurs en temps et en espace*. Plus l'onde est raide, plus l'effondrement de l'erreur est marqué ce qui suggère que cette amélioration vient avant tout d'un meilleur suivi du front d'onde. Enfin, quand le pas de temps continue de diminuer, l'erreur remonte comme si une sorte d'instabilité pollueait la solution (voir figure AFAIRE).

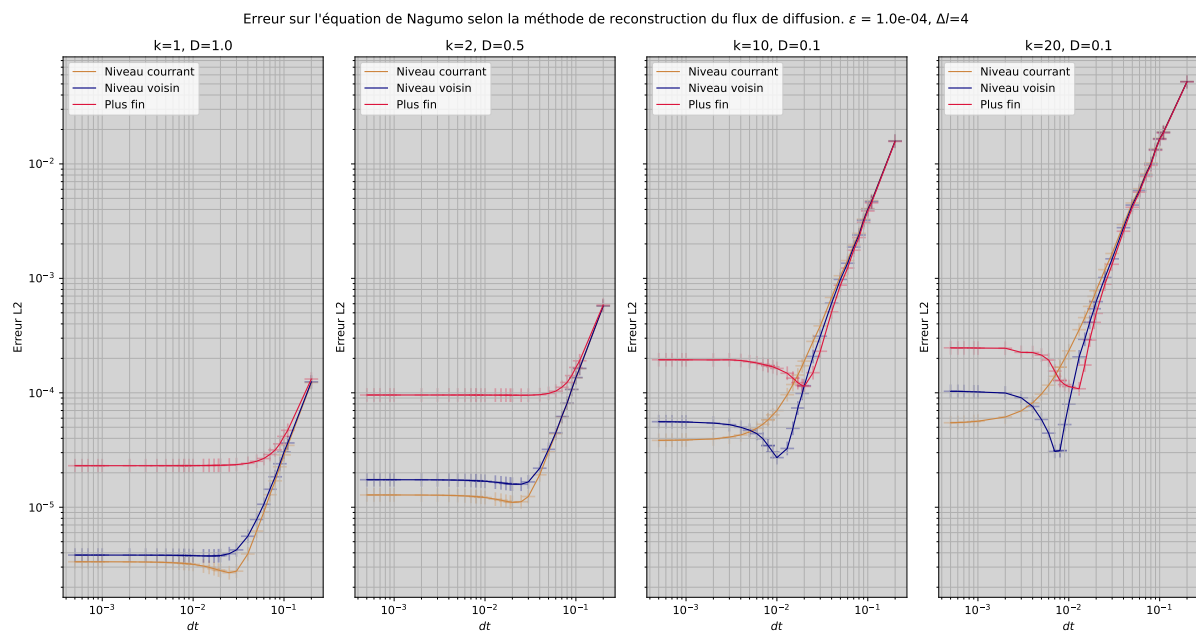


FIGURE 3.15 – Courbes de convergence de chaque méthode d'AMR pour différents paramètres de l'équation. Plus k est élevé, plus le profil de l'onde est raide et plus la réaction domine. La célérité de l'onde est néanmoins identique pour chaque jeu de paramètres puisque le produit kD reste constant d'une expérience à l'autre.

Chapitre 4

Conclusion

Conclusion scientifique Ce stage pave le chemin vers une compréhension plus fine du comportement des schémas de multi-résolution adaptative appliqués aux problèmes de diffusion. L'analyse théorique, fondée sur les équations équivalentes et les études de stabilité, a montré qu'un couplage peut apparaître entre les schémas d'intégration en temps et la MRA, réduisant l'ordre de convergence attendu. Les expériences numériques ont révélé un phénomène surprenant : une reconstruction de flux plus précise ne garantit pas nécessairement une meilleure exactitude globale. Si le lien entre théorie et expériences reste en partie à clarifier, la mise en évidence de ces phénomènes fournit une base solide et des pistes pour de futures investigations.

Conclusion sur ma progression technique Sur le plan théorique, j'ai approfondi des domaines variés : méthodes de volumes finis [26], systèmes dynamiques et méthodes de RungeKutta additives [18], ainsi que la théorie des ondelettes, en résonance avec mes cours antérieurs. Sur le plan pratique, je me suis familiarisé avec des outils puissants d'analyse (équations équivalentes, analyse de Von Neumann), avec des codes de recherche avancés tels que Samurai, et avec les exigences de la simulation numérique (estimation rigoureuse des erreurs, gestion de grilles de taille différente, instabilités). J'ai aussi développé mes compétences de programmation scientifique en Python et en C++, ainsi que ma maîtrise de l'environnement Unix (terminal, git, bash). Enfin, j'ai renforcé mes capacités de communication scientifique, en diffusant mes codes pour en favoriser la reproductibilité et en produisant des graphiques complexes mais lisibles, parfois interactifs.

Conclusion personnelle Ce stage a renforcé mon intérêt pour les mathématiques appliquées, en me confrontant à la fois aux exigences théoriques de l'analyse et aux réalités concrètes de la mise en œuvre computationnelle. Cela a révélé mon épanouissement au sein des environnements où les méthodes mathématiques contribuent directement à résoudre des problématiques complexes, qu'elles relèvent de la recherche scientifique ou d'applications technologiques. Les gains rendus possibles par la diversité de l'équipe, la réutilisation des outils développés et leur interopérabilité m'ont fait prendre conscience de l'importance du travail collectif et de la mise en commun des savoir-faire dans la réussite d'un projet scientifique.

Bibliographie

- [1] A. ABDULLE et G. VILMART. “PIROCK : A swiss-knife partitioned implicit–explicit orthogonal Runge–Kutta Chebyshev integrator for stiff diffusion–advection–reaction problems with or without noise”. In : *Journal of Computational Physics* 242 (2013), p. 869-888.
- [2] Assyr ABDULLE. “Fourth Order Chebyshev Methods with Recurrence Relation”. In : *SIAM Journal on Scientific Computing* 23.6 (2002), p. 2041-2054. DOI : [10.1137/S1064827500379549](https://doi.org/10.1137/S1064827500379549).
- [3] Assyr ABDULLE et Alexei A. MEDOVikov. “Second order Chebyshev methods based on orthogonal polynomials”. In : *Numerische Mathematik* 90.1 (2001), p. 1-18. DOI : [10.1007/s002110100292](https://doi.org/10.1007/s002110100292).
- [4] Mojtaba ALIASGHAR-MAMAGHANI et al. “Multiphysics Modeling of Chloride-Induced Corrosion Damage in Concrete Structures”. In : *Computers & Structures* 308 (2025), p. 107643. ISSN : 0045-7949. DOI : [10.1016/j.compstruc.2025.107643](https://doi.org/10.1016/j.compstruc.2025.107643).
- [5] Uri M. ASCHER, Steven J. RUUTH et Raymond J. SPITERI. “Implicit-explicit Runge-Kutta methods for time-dependent partial differential equations”. In : *Applied Numerical Mathematics* 25.2 (1997). Special Issue on Time Integration, p. 151-167. ISSN : 0168-9274. DOI : [https://doi.org/10.1016/S0168-9274\(97\)00056-1](https://doi.org/10.1016/S0168-9274(97)00056-1). URL : <https://www.sciencedirect.com/science/article/pii/S0168927497000561>.
- [6] BELLOTI et al. “Modified equation and error analyses on adaptative meshes for the resolution of evolutionary PDEs with Finite Volume schemes”. In : (2025).
- [7] M. BOUCHET. *Le laplacien discret 1D*. Notes de cours. Agrégation externe de mathématiques 2019-2020, Leçons 144, 155, 222, 226, 233. ENS Rennes, 2020. URL : <https://perso.eleves.ens-rennes.fr/~mbouc892/lapdisc1d.pdf>.
- [8] Philippe G. CIARLET. *Introduction à l'analyse numérique matricielle et à l'optimisation*. 1^{re} éd. Ouvrage de référence en analyse numérique et optimisation. Paris : Dunod, 1982. ISBN : 978-2-04-012365-3.
- [9] A. COHEN et al. “Fully adaptive multiresolution finite volume schemes for conservation laws”. In : *Mathematics of Computation* 72 (2003).
- [10] Marcus Ó CONAIRE et al. “A Comprehensive Modeling Study of Hydrogen Oxidation”. In : *International Journal of Chemical Kinetics* 36.11 (2004), p. 603-622. DOI : [10.1002/kin.20026](https://doi.org/10.1002/kin.20026).
- [11] V. DARU et C. TENAUD. “High order one-step monotonicity-preserving schemes for unsteady compressible flow calculations”. In : *Journal of Computational Physics* 193.2 (2004), p. 563-594. ISSN : 0021-9991. DOI : <https://doi.org/10.1016/j.jcp.2003.08.023>. URL : <https://www.sciencedirect.com/science/article/pii/S0021999103004327>.

- [12] Ralf DEITERDING et al. "Comparison of Adaptive Multiresolution and Adaptive Mesh Refinement Applied to Simulations of the Compressible Euler Equations". In : *SIAM Journal on Scientific Computing* 38.5 (2016), S173-S193. DOI : [10.1137/15M1026043](https://doi.org/10.1137/15M1026043).
- [13] Max Pedro DUARTE. "Méthodes numériques adaptives pour la simulation de la dynamique de fronts de réaction multi-échelle en temps et en espace". 2011ECAP0057. Thèse de doct. 2011. URL : <http://www.theses.fr/2011ECAP0057/document>.
- [14] Tarek ECHEKKI. "Multiscale methods in turbulent combustion : strategies and computational challenges". In : *Computational Science & Discovery* 2.1 (2009), p. 013001. DOI : [10.1088/1749-4699/2/1/013001](https://doi.org/10.1088/1749-4699/2/1/013001).
- [15] Richard FITZHUGH. "Impulses and Physiological States in Theoretical Models of Nerve Membrane". In : *Biophysical Journal* 1.6 (1961), p. 445-466. ISSN : 0006-3495. DOI : [https://doi.org/10.1016/S0006-3495\(61\)86902-6](https://doi.org/10.1016/S0006-3495(61)86902-6). URL : <https://www.sciencedirect.com/science/article/pii/S0006349561869026>.
- [16] V. GUVANASEN et R. E. VOLKER. "Numerical solutions for solute transport in unconfined aquifers". In : *International Journal for Numerical Methods in Fluids* 3.2 (1983), p. 103-123. DOI : <https://doi.org/10.1002/flid.1650030203>. eprint : <https://onlinelibrary.wiley.com/doi/pdf/10.1002/flid.1650030203>. URL : <https://onlinelibrary.wiley.com/doi/abs/10.1002/flid.1650030203>.
- [17] E. HAIRER. "Order conditions for numerical methods for partitioned ordinary differential equations". In : *Numerische Mathematik* 36.4 (1981), p. 431-445. ISSN : 0945-3245. DOI : [10.1007/BF01395956](https://doi.org/10.1007/BF01395956). URL : <https://doi.org/10.1007/BF01395956>.
- [18] Ernst HAIRER, Syvert P. NØRSETT et Gerhard WANNER. *Solving Ordinary Differential Equations I: Nonstiff Problems*. 2^e éd. T. 8. Springer Series in Computational Mathematics. Springer Berlin, Heidelberg, 1993, p. XV, 528. ISBN : 978-3-540-56670-0. DOI : [10.1007/978-3-540-78862-1](https://doi.org/10.1007/978-3-540-78862-1). URL : <https://doi.org/10.1007/978-3-540-78862-1>.
- [19] Ami HARTEN. "Adaptive Multiresolution Schemes for Shock Computations". In : *Journal of Computational Physics* 115.2 (1994), p. 319-338. ISSN : 0021-9991. DOI : <https://doi.org/10.1006/jcph.1994.1199>. URL : <https://www.sciencedirect.com/science/article/pii/S0021999184711995>.
- [20] M. KAIBARA et S. M. GOMES. "A fully adaptive multiresolution scheme for shock computations". In : *Godunov Methods : Theory and Applications*. Sous la dir. d'E. F. TORO. Kluwer Academic/-Plenum Publishers, 2001.
- [21] James KEENER et James SNEYD. *Mathematical Physiology*. 1^{re} éd. Interdisciplinary Applied Mathematics. New York, NY : Springer, 1998, p. 281. ISBN : 978-0-387-22706-1. DOI : [10.1007/b98841](https://doi.org/10.1007/b98841).
- [22] Christopher A. KENNEDY et Mark H. CARPENTER. "Additive RungeKutta schemes for convectiondiffusionreaction equations". In : *Applied Numerical Mathematics* 44.1 (2003), p. 139-181. ISSN : 0168-9274. DOI : [https://doi.org/10.1016/S0168-9274\(02\)00138-1](https://doi.org/10.1016/S0168-9274(02)00138-1). URL : <https://www.sciencedirect.com/science/article/pii/S0168927402001381>.

- [23] Christopher A. KENNEDY et Mark H. CARPENTER. “Diagonally Implicit Runge-Kutta Methods for Ordinary Differential Equations. A Review”. In : *NASA STI Program* (2025).
- [24] Chung K. LAW. *Combustion Physics*. Cambridge : Cambridge University Press, 2006. DOI : [10.1017/CB09780511754517](https://doi.org/10.1017/CB09780511754517).
- [25] Randall J. LEVEQUE. *Finite Difference Methods for Ordinary and Partial Differential Equations : Steady-State and Time-Dependent Problems*. Classics in Applied Mathematics. Philadelphia : Society for Industrial et Applied Mathematics, 2007. ISBN : 978-0-898716-29-0. DOI : [10.1137/1.9780898717839](https://doi.org/10.1137/1.9780898717839).
- [26] Randall J. LEVEQUE. *Numerical Methods for Conservation Laws*. Lectures in Mathematics ETH Zürich. Basel : Birkhäuser Verlag, 1990. ISBN : 978-3-7643-2464-3. DOI : [10.1007/978-3-0348-5116-9](https://doi.org/10.1007/978-3-0348-5116-9).
- [27] Iulian MUNTEANU et al. “Single-Particle Model of Li-ion Battery – Model Calibration and Validation Against Real Data in an Electric Vehicular Application”. In : *IFAC-PapersOnLine* 58.13 (2024). 12th IFAC Symposium on Control of Power & Energy Systems (CPES 2024), Rabat, Morocco, 10–12 July 2024, p. 23-30. ISSN : 2405-8963. DOI : [10.1016/j.ifacol.2024.07.454](https://doi.org/10.1016/j.ifacol.2024.07.454).
- [28] L. PARESCI et G. RUSSO. *Implicit-explicit Runge-Kutta schemes and applications to hyperbolic systems with relaxation*. 2010. arXiv : [1009.2757 \[math.NA\]](https://arxiv.org/abs/1009.2757). URL : <https://arxiv.org/abs/1009.2757>.
- [29] Marie POSTEL. “Approximations multiéchelles”. Polycopié, Laboratoire Jacques-Louis Lions, Université Pierre et Marie Curie.
- [30] Louis REBOUL. “Development and analysis of efficient multi-scale numerical methods, with applications to plasma discharge simulations relying on multi-fluid models”. 2024IPPAX134. Thèse de doct. 2024. URL : <http://www.theses.fr/2024IPPAX134/document>.
- [31] Diego ROSSINELLI et al. “Multicore/Multi-GPU Accelerated Simulations of Multiphase Compressible Flows Using Wavelet Adapted Grids”. In : *SIAM Journal on Scientific Computing* 33.2 (2011), p. 512-540. DOI : [10.1137/100795930](https://doi.org/10.1137/100795930). URL : <https://doi.org/10.1137/100795930>.
- [32] Gilbert STRANG. “On the construction and comparison of difference schemes”. In : *SIAM Journal on Numerical Analysis* 5.3 (1968), p. 506-517.
- [33] Guglielmo VIVARELLI, Ning QIN et Shahrokh SHAHPAR. “A Review of Mesh Adaptation Technology Applied to Computational Fluid Dynamics”. In : *Fluids* 10.5 (mai 2025). ISSN : 2311-5521. DOI : [10.3390/fluids10050129](https://doi.org/10.3390/fluids10050129). URL : <https://www.mdpi.com/2311-5521/10/5/129>.
- [34] Eva-Maria WARTHA, Markus BÖSENHOFER et Michael HARASEK. “Characteristic Chemical Time Scales for Reactive Flow Modeling”. In : *Chemical Engineering & Technology* 44.7 (2021), p. 1240-1249. DOI : [10.1002/ceat.202000573](https://doi.org/10.1002/ceat.202000573).