



# Numerical study of multiscale non conservative transport equations modeling cell kinetics

Benjamin Aymard

► **To cite this version:**

Benjamin Aymard. Numerical study of multiscale non conservative transport equations modeling cell kinetics. General Mathematics [math.GM]. Université Pierre et Marie Curie - Paris VI, 2014. English. | NNT : 2014PA066254 |.

**HAL Id: tel-01087504**

**<https://tel.archives-ouvertes.fr/tel-01087504>**

Submitted on 26 Nov 2014

**HAL** is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.



---

**Simulation numérique d'un modèle multi-échelle  
de cinétique cellulaire formulé à partir  
d'équations de transport non conservatives**

**THÈSE DE DOCTORAT**

présentée par

**BENJAMIN AYMARD**

pour obtenir le grade de

**Docteur de l'Université Pierre et Marie Curie**

Spécialité

**Mathématiques Appliquées**

sous la direction de **FRÉDÉRIQUE CLEMENT** et **MARIE  
POSTEL**

Soutenue publiquement le 10/10/2014 devant le jury composé de

ADIMY MOSTAFA	Examineur
CHALONS CHRISTOPHE	Examineur
CLEMENT FRÉDÉRIQUE	Directeur de Thèse
COQUEL FRÉDÉRIC	Examineur
DONAT ROSA	Rapporteur
PERTHAME BENOÎT	Examineur
POSTEL MARIE	Directeur de Thèse



# Numerical study of multiscale non conservative transport equations modeling cell kinetics

---

## Abstract

---

The thesis focuses on the numerical simulation of a biomathematical, multiscale model explaining the phenomenon of selection within the population of ovarian follicles, and grounded on a cellular basis. The PDE model consists of a large dimension hyperbolic quasilinear system governing the evolution of density functions of cells (microscopic scale) for  $N_f$  follicles (in practice  $N_f$  is on the order of twenty).

These equations make use of two structural variables, the age and maturity, which play the role of the space variables. The equations are coupled in a nonlocal way by control terms involving moments of the solution with respect to maturity, defined on either the mesoscopic (follicle) or the macroscopic (ovaries) scale.

Three chapters of the thesis, presented in the form of articles [6, 7, 8], develop the method used to simulate the model numerically. The numerical code is designed to be implemented on a parallel architecture (MPI). PDEs are discretized with a Finite Volume scheme on an adaptive mesh driven by a multiresolution analysis. Flux discontinuities, at the interfaces between different cellular states, require a specific treatment to be compatible with, on the one hand, the high order numerical scheme and, on the other hand, the mesh refinement.

A chapter of the thesis is devoted to the calibration method, which translates the biological knowledge into constraints on the parameters and the model outputs. The multiscale character is crucial here again in the sense that parameters are used at the microscopic level in the equations governing the evolution of the density of cells within each follicle, whereas quantitative biological data are available at the mesoscopic and macroscopic levels.

The development of tests with a significant biological meaning requires intensive simulations, and the last chapter of the thesis focuses on the analysis of computational performances of the parallel code, based on statistical methods inspired from the field of uncertainty quantification.



# Simulation numérique d'un modèle multi-échelle de cinétique cellulaire formulé à partir d'équations de transport non conservatives

---

## Résumé

---

La thèse porte sur la calibration d'un modèle biomathématique multi-échelle expliquant le phénomène de sélection des follicules ovariens à partir du niveau cellulaire. Le modèle EDP consiste en un système hyperbolique quasi linéaire de grande taille gouvernant l'évolution des fonctions de densité cellulaire pour  $N_f$  follicules (en pratique  $N_f$  est de l'ordre d'une vingtaine). Ces équations d'évolution utilisent deux variables structurantes, l'âge et la maturité, qui jouent le rôle de variables d'espace. Les équations sont couplées de manière non locale par l'intermédiaire de termes de contrôle faisant intervenir les moments en maturité de la solution, intégrée à l'échelle mésoscopique (du follicule) et macroscopique (de l'ovaire).

Trois chapitres de la thèse présentent, sous forme d'articles publiés [6, 7, 8], la méthode développée pour simuler numériquement ce modèle. Elle est conçue pour être implémentée sur une architecture parallèle (MPI). Les EDP sont discrétisées avec un schéma Volumes Finis sur un maillage adaptatif piloté par une analyse multirésolution. Le modèle présente des discontinuités de flux aux interfaces entre les différents états cellulaires, qui nécessitent la mise en œuvre d'un couplage spécifique pour être compatible avec, d'une part, le schéma d'ordre élevé et, d'autre part, le raffinement de maillage.

Un chapitre de la thèse est dévolu à la méthode de calibration, qui consiste à traduire les connaissances biologiques en contraintes sur les paramètres et sur les sorties du modèle. Le caractère multi-échelle est là encore crucial au sens où les paramètres interviennent au niveau microscopique dans les équations gouvernant l'évolution des densités de cellules au sein de chaque follicule, alors que les données biologiques quantitatives sont disponibles au niveau mésoscopique et macroscopique sur les sorties agrégées en espace.

La mise au point de cas tests significatifs du point de vue biologique est très consommatrice de temps de calcul, et le dernier chapitre de la thèse porte sur l'analyse des performances numériques du code parallèle, avec une méthodologie empruntée aux méthodes statistiques de propagation d'incertitudes.



A Plume.





# Remerciements

Je remercie mes directrices de thèse, Marie Postel et Frédérique Clément, pour leur confiance en mon travail et mes idées, avec qui j'ai beaucoup appris, et qui m'ont toujours bien encadré.

Je remercie Marie d'avoir partagé avec moi sa grande expérience en numérique, de m'avoir poussé à être plus rigoureux, de m'avoir appris à mettre proprement en forme mes idées, avec notamment beaucoup d'astuces de rédaction. Merci aussi de m'avoir soutenu pour mes candidatures, en thèse puis en postdoc. Merci enfin pour votre patience, et toutes nos séances de travail au tableau, qui m'ont été très profitables.

Je remercie Frédérique, scientifique polyvalente, à la fois Mathématicienne et Biologiste, avec qui j'ai aussi beaucoup appris. J'apprécie (et j'admire) votre capacité à prendre du recul sur un projet et en avoir une vision à long terme, mais aussi à en gérer l'évolution pas à pas.

Je remercie Danielle Monniaux, de l'INRA Tours-Nouzilly, pour toutes nos réunions d'échanges et de confrontations de résultats, qui ont été très constructives, et même indispensables à l'évolution de notre projet. Merci aussi de nous avoir invités à plusieurs reprises à l'INRA, où nous avons eu l'immense privilège de visiter les installations du centre.

Merci à mes anciens professeurs qui ont eu un rôle décisif dans ma scolarité. Merci à Laurent Mertz, moniteur en L3, qui m'a fait découvrir l'analyse numérique des EDOs, et m'a poussé à continuer, au moment où j'allais arrêter mes études. Merci à Martin Vohralik, qui m'a initié en M1 à l'analyse numérique des EDPs. Un grand merci à Christophe Chalons, avec qui j'ai débuté la résolution numérique de systèmes de lois de conservation, et qui m'a fait l'honneur d'accepter de faire partie de mon jury de thèse.

Un immense merci à Frédéric Coquel, qui a été mon enseignant en M2, m'a initié aux problèmes hyperboliques, et m'en a transmis sa passion. Merci de m'avoir contacté pour me mettre en relation avec Marie, avec qui j'ai fait mon stage, qui a débouché sur cette thèse. Cela a été un honneur, et un plaisir, de travailler avec toi, car en plus d'être une encyclopédie vivante des problèmes hyperboliques, tous plus intéressants les uns que les autres, j'ai beaucoup apprécié tes qualités humaines.

Je remercie chaleureusement Rosa Donat et Emmanuel Grenier, qui ont accepté de rapporter ma thèse, ainsi que Benoît Perthame et Mostafa Adimy, qui ont accepté de faire partie de mon jury.

Merci à celles et ceux avec qui j'ai eu le plaisir de faire de l'enseignement en analyse numérique en tant que moniteur à l'UPMC : Leila Djaziri et Mohamed Oumri en

---

première année, Paul Cazeaux en deuxième année, Cindy Guichard, Alexander Lorz et Haïdar Mohamad en troisième année. L'ambiance était excellente, et les journées de soutenances orales resteront un bon souvenir (même si sur le coup cela paraissait très long!).

Merci aux nombreux membres du Laboratoire Jacques-Louis Lions. Ce Laboratoire est un cadre de travail exceptionnel, et c'est un honneur pour moi d'avoir fait partie de la liste de ses membres pendant mes trois années de thèse. Merci aux "sys-admin" Khashayar Dadras et ses complices Altaïr Pélissier et Hugues Moretto-Viry, pour leur efficacité, leur rapidité d'action, leur sympathie et leur sens de l'humour. Encore merci, au passage, pour votre aide dans l'installation de mes logiciels météo. Merci aussi à Antoine Le Hyaric, notamment pour ses conseils avisés en OpenGL. Merci à Pascal Joly et Philippe Parnaudeau pour leur sympathie, leur humour, et surtout pour leur aide en calcul parallèle, et pour le fonctionnement du cluster HPC1. Merci aux secrétaires Liliane Ruprecht, Salima Lounici, Nadine Foucart et Catherine Drouet, pour leur sympathie et leur aide précieuse. Merci à Pierre Lissy, avec qui nous avons géré les bureaux des doctorants, et merci à Mehdi Badsî et Dena Kazerani d'avoir repris le flambeau en troisième année. Merci enfin aux doctorants et post doctorants, en commençant par les anciens : Anne-Claire Egloff (et Christophe Denis), Marie Kray, Frédérique Billy, Annabelle Ballesta, Luna Dimitrio, Mamadou Gueye. Merci aux doctorants du 2ème étage (qui travaillent sur des sujets tellement sensibles qu'ils doivent travailler dans un bureau secret caché au LPMA) : Jean-Paul Daniel, Malik Drici, Giacomo Canevari (et bien sûr son colocataire Giulio), Casimir Emako (j'attends avec impatience ton article dans Nature), Tommaso Mingazzini, Sylvain Arguillère, Ludovick Gagnon et Guillaume Vergez. Et bien sûr un grand merci à Juliette Bouhours pour son immense gentillesse. Merci à celles et ceux que j'ai moins eu l'occasion de voir, avec qui j'ai passé de très bon moments, comme Ariane Trescasses, Justine Fouchet-Incaux, Sarra Maarouf, Rahma Agroum, Jiamin Zhu, Eugénie Poulon, Geneviève Dusson, Ange Toulougoussou, Pierre-Henri Tournier ("PhT") et Philippe Ung, Amine Loumi, Stefan Czimek, Carlo Marcati, Cristobal Quininao, Ryadh Aferssas. Merci, et bonne chance, aux doctorants de ma promotion : Olga Mula Hernandez, Abdellah Chkifa, Nicolas Carreno Godoy, Alicia Wang, Yannick Deleuze, Elisa Schenone et Pierre Jolivet. Merci au bureau "d'en face", le 329 : Sarah Eugène, Jan Elias, Thibault Liard, Pierre Jounieaux et Maxime Chupin. Un merci spécial pour le bureau 324 : Magali Tournus, co bureau pleine d'énergie, Wafaâ Haffaf, co bureau d'une immense gentillesse, Quang Van Nguyen, co bureau aussi discret que sympathique, Thibault Bourgeron, co bureau expert en Wikipédia et en nourriture gratuite, Ibrahim/Kenji Cheddadi et Anne-Céline Boulanger, nos invités permanents. Tout en travaillant, on a bien rigolé pendant ces trois années. L'ambiance était excellente, et c'était un plaisir de venir tous les jours dans ce bureau (qui, comme tout le monde le sait, est le meilleur bureau, et c'est bien pour cela qu'on veut nous l'enlever!). Je garderai un super souvenir de nos pauses thé, nos pauses déjeuners, nos repas de bureau.

Merci au centre de recherche de l'Inria Paris-Rocquencourt. Ce fut un cadre de travail exceptionnel, et fort agréable. Un grand merci à mon ancienne co-bureau Peipei

---

Shang, ainsi qu'à son amie Jixun Chu. Nous avons passé de très bon moments (je me rappellerai longtemps de votre leçon de cuisine de raviolis chinois!). Merci à Alain Marchand pour son aide précieuse en informatique. Merci à Martine Verneuille de m'avoir grandement facilité toutes les démarches administratives (c'est simple, je n'ai jamais eu de problème). Merci aux membres de l'équipe-projet Mycenae, et notamment ses sympathiques doctorants et post doctorants : Lucile Mégret, Elif Koksal, Soleidad Fernandez Garcia, Albert Granados. Merci aussi au passage à Mathieu Desroches. Merci aux membres d'autres équipes avec qui j'ai aussi passé de très bons moments : Xavier Dupuis, Gregory Arbia (le seul doctorant au monde à être propriétaire de sa maison et rouler en BMW), Chloé Audebert, Noémie Boissier et nos discussions voile/régate dans la navette Inria, Annabelle Collin, Aurora Armiento, François Bertaux, Emanuele Leoncini.

Merci aussi à celles et ceux que j'ai croisés et m'ont apporté leur aide, en plus de leur gentillesse, notamment Florent Renac (pour l'utilisation du beamer) et Julie Tryoen (pour LaTeX).

Merci au CNIF (Club de voile de l'UPMC). Tous ces instants de navigation m'ont permis de relativiser mes problèmes en Mathématiques, de prendre du recul, et ont, je pense, fait avancer cette thèse de manière indirecte. Merci donc à tout les vieux loups de mer du Club : Gilles Mawart, Jean-Yves Prieur, Christian d'Hier, Jean-Pierre Guillaume, Stéphane Rodes, Olivier Vacheron, ainsi qu'aux jeunes loups comme Fred Bordas ou Seb' Javaudin! Merci aussi à toutes les équipières et tous les équipiers que j'ai eu à bord. Un immense merci à Bernard Franc pour m'avoir formé, et pour nos navigations en Angleterre et en Irlande.

Enfin, merci à mes proches, pour votre soutien, votre patience, votre compréhension. Sans vous tous, cette thèse n'existerait pas. Merci à Stoby, Laure et les jumelles Erika et Emi. Merci à Arnaud, Cindy et Selena, et bien sûr David. Merci à Rémi et Mathilde. Merci à Matox et Justine. Merci à Greg Stock. Merci à Tony et Nath'. Merci à mes parents, Murielle et Serge, de m'avoir toujours soutenu, quoi qu'il arrive. Un immense merci à mon frère Nathan. Merci aussi à Sylvain, avec qui j'ai découvert ce qu'était une thèse, ainsi qu'à Gérard. Enfin, merci à Oana, que j'ai rencontrée pendant cette thèse, et qui partage à présent ma vie (merci pour ton soutien).

À vous toutes et à vous tous, qui avez contribué de près ou de loin à la réalisation de cette thèse, une fois de plus merci. À toutes celles et ceux que j'aurais pu oublier dans cette longue liste (et il y en a sûrement malheureusement), toutes mes excuses.



"Quel message délivrer après toutes ces expériences ? Il se résume en un mot : persévérer. C'est la leçon que j'ai tirée des 63 jours de marche en solitaire jusqu'au Pôle Nord. Sur ce parcours chaotique, j'ai souvent failli abandonner, et je dois à la persévérance l'immense satisfaction et la force intérieure gagnées au bout de cet impossible défi que je m'étais inventé. Il faut suivre la voie de ses rêves, même si le chemin est difficile".

Dr Jean-Louis Etienne.



# Table des matières

<b>1</b>	<b>Introduction</b>	<b>1</b>
1.1	Historique et Motivations . . . . .	1
1.2	Modèle multiéchelles . . . . .	4
1.2.1	Inconnues du problème . . . . .	5
1.2.2	Système de lois de conservations . . . . .	6
1.2.3	Contrôle non local . . . . .	10
1.3	Apports de la thèse . . . . .	12
1.3.1	Numerical simulation of the selection process of the ovarian follicles . . . . .	12
1.3.2	A numerical method for transport equations with discontinuous flux functions : application to mathematical modeling of cell dynamics . . . . .	13
1.3.3	Adaptive mesh refinement strategy for a non conservative transport problem . . . . .	17
1.3.4	Calibration . . . . .	18
1.3.5	Schéma numérique multiéchelles et analyse de performances par propagation d'incertitudes . . . . .	21
<b>2</b>	<b>Numerical simulation of the selection process of the ovarian follicles</b>	<b>25</b>
2.1	Biological and biomathematical background . . . . .	27
2.1.1	Biological background . . . . .	27
2.1.2	Biomathematical model . . . . .	29
2.2	Numerical Method . . . . .	32
2.2.1	Discretization . . . . .	32
2.2.2	Macroscopic scale : piecewise constant approximation of the hormonal control . . . . .	33
2.2.3	Mesosopic scale : finite volume scheme . . . . .	33
2.2.4	Macro/Meso scale Parallelization . . . . .	35
2.2.5	Parallel algorithm and improvement to order 2 in time . . . . .	36
2.3	Numerical simulations . . . . .	36
2.3.1	Initial condition . . . . .	36
2.3.2	Sets of default parameters . . . . .	37
2.3.3	Time evolution of the cell density for one follicle . . . . .	38
2.3.4	Competition between ten follicles . . . . .	39
2.3.5	Convergence rate test . . . . .	40
2.3.6	HPC test . . . . .	41



<b>3</b>	<b>A numerical method for transport equations with discontinuous flux functions : application to mathematical modeling of cell dynamics</b>	<b>49</b>
3.1	Introduction . . . . .	50
3.2	Toy models . . . . .	52
3.3	Numerical scheme . . . . .	54
3.3.1	Discretization . . . . .	54
3.3.2	Treatment of discontinuous coefficients . . . . .	56
3.4	Numerical validation . . . . .	58
3.4.1	1D test cases . . . . .	58
3.4.2	2D test cases . . . . .	64
3.5	Application to cell dynamics . . . . .	69
3.5.1	Numerical simulation . . . . .	74
3.6	Appendix : Exact solutions . . . . .	75
3.6.1	Exact solutions of the 1D problem with piecewise constant speeds . . . . .	75
3.6.2	Exact solution of the 2D problems with piecewise constant speeds . . . . .	77
<b>4</b>	<b>Adaptive mesh refinement strategy for a non conservative transport problem</b>	<b>81</b>
4.1	Introduction . . . . .	82
4.2	Investigation of numerical noise appearance . . . . .	84
4.2.1	Description of the simplified set-up . . . . .	84
4.2.2	Finite volume discretization on a two-level MR hierarchy . . . . .	85
4.2.3	Explication of the numerical noise appearance . . . . .	88
4.2.4	Adaptive doubling strategy . . . . .	91
4.2.5	Formalization and generalization of the doubling strategy . . . . .	92
4.3	Numerical validation of the adaptive thresholds strategies . . . . .	94
4.3.1	Effect of the adaptive doubling strategy on the solution . . . . .	94
4.3.2	Time adaptation of the MR threshold $\varepsilon_t$ . . . . .	99
4.3.3	Link between the MR and the doubling thresholds . . . . .	100
4.4	Application to a biological model . . . . .	102
4.4.1	Model and numerical method . . . . .	102
4.4.2	Numerical performances . . . . .	106
4.4.3	Comparison of the numerical performances for both models of mitosis . . . . .	110
4.5	Appendix : Description of the model for follicular development . . . . .	112
4.6	Appendix : Example of spurious numerical noise appearance . . . . .	114
4.7	Numerical noise appearance . . . . .	114
<b>5</b>	<b>Calibration</b>	<b>121</b>
5.1	Degrés de liberté . . . . .	122
5.1.1	Conditions initiales . . . . .	122

## Table des matières

---

5.1.2	Fonctions de vitesses . . . . .	123
5.1.3	Variables de contrôle . . . . .	127
5.2	Spécifications biologiques . . . . .	134
5.2.1	Mesures disponibles . . . . .	134
5.2.2	Trajectoires folliculaires . . . . .	136
5.3	Aides à la calibration . . . . .	136
5.3.1	Contraintes a priori . . . . .	136
5.3.2	Visualisation à une échelle microscopique . . . . .	137
5.3.3	Diagnostic a posteriori . . . . .	141
5.4	Méthode de calibration pour un follicule . . . . .	146
5.4.1	Détermination de la maturité asymptotique . . . . .	146
5.4.2	Détermination des paramètres affectant la dynamique . . . . .	149
5.4.3	Exemple de situation de référence à 1 follicule . . . . .	150
5.5	Méthode de calibration pour une cohorte de follicules . . . . .	153
5.5.1	Réglage de la pression sélective . . . . .	154
5.5.2	Abaques multi follicules . . . . .	155
5.5.3	Exemple de situation de référence à 10 follicules . . . . .	157
<b>6</b>	<b>Design and performance analysis of a multiscale scheme for controlled population dynamics</b> . . . . .	<b>163</b>
6.1	Multiscale model . . . . .	164
6.1.1	Exemple : selection process of ovarian follicles . . . . .	166
6.2	Multiscale numerical method . . . . .	169
6.2.1	Microscopic level : adaptive finite volume scheme . . . . .	170
6.2.2	Mesosopic and macroscopic scale : Numerical moments and control. . . . .	181
6.2.3	Parallel computing . . . . .	182
6.3	Performance analysis . . . . .	184
6.3.1	Method . . . . .	186
6.3.2	Mean load balance test . . . . .	187
6.3.3	Mean weak scalability test . . . . .	191
<b>7</b>	<b>Conclusion</b> . . . . .	<b>201</b>
<b>A</b>	<b>CodeFollicleMR : Notice utilisateur (getting started)</b> . . . . .	<b>203</b>
A.1	Simulation avec paramètres par défaut . . . . .	203
A.2	Visualisation . . . . .	203
A.3	Simulation avec plusieurs follicules, en choisissant un jeu de paramètres . . . . .	204
A.4	Simulation en créant son propre jeu de paramètres . . . . .	205
A.5	Interface Utilisateur . . . . .	206
<b>B</b>	<b>dualMR : Notice utilisateur (getting started)</b> . . . . .	<b>209</b>
B.1	Comment utiliser dualMR . . . . .	209
B.2	Visualiser les sorties macroscopiques et mésoscopiques . . . . .	209

B.3	Visualiser les sorties microscopiques . . . . .	210
<b>C</b>	<b>CodeFollicleMR : Notice programmeur</b>	<b>213</b>
C.1	Structures de données . . . . .	213
C.2	Organisation du code source . . . . .	217
C.3	Entrées/sorties . . . . .	219
	<b>Bibliographie</b>	<b>221</b>

# Introduction

Nous présentons dans ce manuscrit les résultats obtenus au cours d'une thèse de doctorat en Mathématiques Appliquées, effectuée sous la direction de Frédérique Clément (Mycenae, Inria Paris-Rocquencourt) et de Marie Postel (Laboratoire Jacques-Louis Lions, UPMC - Paris 06).

Nous avons développé une stratégie permettant de réaliser des simulations numériques fiables et performantes d'un modèle multi-échelle représentant la dynamique cellulaire au cours du processus de sélection des follicules ovariens. Ce modèle peut être considéré comme un système d'équations de transport non conservatives et faiblement couplées, et il présente plusieurs spécificités qui soulèvent des problèmes originaux et difficiles du point de vue numérique, qui ont nécessité la mise en œuvre de méthodes numériques nouvelles, dont la présentation est l'objet principal de ce manuscrit.

## Sommaire

<b>1.1</b>	<b>Historique et Motivations</b>	<b>1</b>
<b>1.2</b>	<b>Modèle multiéchelles</b>	<b>4</b>
1.2.1	Inconnues du problème	5
1.2.2	Système de lois de conservations	6
1.2.3	Contrôle non local	10
<b>1.3</b>	<b>Apports de la thèse</b>	<b>12</b>
1.3.1	Numerical simulation of the selection process of the ovarian follicles	12
1.3.2	A numerical method for transport equations with discontinuous flux functions : application to mathematical modeling of cell dynamics	13
1.3.3	Adaptive mesh refinement strategy for a non conservative transport problem	17
1.3.4	Calibration	18
1.3.5	Schéma numérique multiéchelles et analyse de performances par propagation d'incertitudes	21

## 1.1 Historique et Motivations

L'objectif principal de cette thèse est de simuler numériquement, de façon fiable et rapide, un modèle mathématique représentant un processus physiologique complexe, le développement terminal des follicules ovariens et la sélection des follicules

## Chapitre 1. Introduction

ovulatoires.

Les follicules ovariens sont des tissus de géométrie grossièrement sphérique, servant d’abri aux ovocytes (cellules germinales femelles), dont ils accompagnent la maturation, et assurant en partie la fonction endocrine de l’ovaire (sécrétion d’hormones, en particulier l’œstradiol). Le processus de sélection des follicules ovulatoires, qui a lieu au cours de la phase terminale du développement folliculaire, peut être considéré comme un phénomène de compétition indirecte entre follicules à l’issue duquel un ou plusieurs follicules dominants sont sélectionnés pour l’ovulation. Les autres follicules subissent un processus de dégénérescence et deviennent atrétiques. La sélection est opérée via un contrôle hormonal, la ressource principale nécessaire au développement terminal étant l’hormone hypophysaire FSH (follicle-stimulating-hormone) dont les niveaux sont en retour modulés par les hormones ovariennes produites par les follicules ovariens.

Le processus de sélection s’étend donc sur plusieurs échelles spatiales (voir Figure 1.1), allant de l’échelle microscopique (intracellulaire et cellulaire) à l’échelle macroscopique (organique : niveau ovarien et systémique : axe hypothalamo-hypophysogonadique), en passant par le niveau mésoscopique (tissulaire : follicule ovarien).

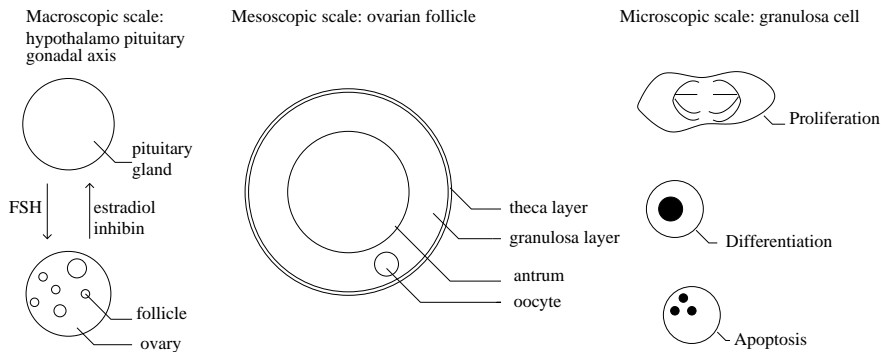


FIGURE 1.1 – *Présentation schématique des différentes échelles impliquées dans le développement folliculaire (Figure extraite de [6]). A gauche : boucle de rétrocontrôle entre les ovaires et l’hypophyse modulant les niveaux de FSH. Au milieu : coupe schématique d’un follicule ovarien. La granulosa, dont certaines cellules entourent l’ovocyte, est délimitée par une cavité liquidienne, l’antrum et par une autre couronne cellulaire, la thèque. A droite : différents états cellulaires rencontrés par les cellules de granulosa.*

Le modèle que nous considérons dans ce manuscrit étudie la dynamique de densités cellulaires évoluant dans un espace fonctionnel âge cytologique  $\times$  maturité ; il permet de distinguer différents états cellulaires (cellules en prolifération, cellules différenciées, cellules en apoptose), mais aussi différentes phases au sein du cycle cellulaire, et de considérer des termes de vitesse et de second membre spécifiques et représentant l’action différentielle du contrôle hormonal qui s’exerce sur les cellules de granulosa en fonction de leur position dans le domaine.

## Chapitre 1. Introduction

---

Pour aboutir à la formulation du modèle telle qu'elle a été proposée pour la première fois dans [41], plusieurs étapes ont été nécessaires. Ces étapes sont détaillées dans [22], nous n'en faisons qu'un résumé succinct ici. Le lecteur intéressé pourra aussi consulter les articles de synthèse [27], [23] et [61] pour plus de détails sur les hypothèses biologiques sous-jacentes.

Dans une première étape, un modèle compartimental à base d'ODEs a été développé pour représenter la répartition de l'effectif cellulaire de la granulosa entre des cellules en prolifération et des cellules différenciées [26]. Les taux de transfert entre compartiments sont dépendants de l'âge folliculaire et leurs paramètres peuvent être déterminés en ajustant l'effectif total aux données biologiques disponibles. Un problème de contrôle optimal défini à partir de ce modèle a permis de déterminer que la stratégie optimale permettant de minimiser le délai avant l'instant d'ovulation (défini à partir de conditions sur l'effectif différencié) consistait à utiliser des taux de transfert de type bang-bang, et de préciser la chronologie des événements ovulatoires dans ce cas (instants de commutation, de déclenchement de la décharge ovulatoire et d'ovulation) [21].

A ce stade, la difficulté majeure pour représenter la dynamique cellulaire, non pas en fonction de l'âge folliculaire, mais de l'historique d'exposition d'un follicule en terme d'environnement hormonal, venait de ce que l'action de cet environnement (c'est-à-dire de l'hormone FSH) varie suivant le degré de maturité cellulaire. Schématiquement, cette hormone favorise la prolifération des cellules moins matures (majoritaires dans les follicules en début de développement terminal) tandis qu'elle favorise la différenciation terminale dans les cellules plus matures. Il fallait donc pouvoir distinguer les effets de FSH en terme de prolifération et en terme de différenciation et les moduler selon l'état cellulaire.

Une deuxième étape a donc consisté à étudier plus finement la transduction du signal FSH, en suivant une approche de type modélisation biochimique, tout en tenant compte de modifications à long terme dans cette signalisation (phénomène d'auto-amplification) [28].

La troisième et dernière étape a consisté à réintégrer les dynamiques précédemment élaborées au niveau intracellulaire et cellulaire, dans un formalisme spatialisé (EDP en 2D), ce qui permet de suivre l'évolution des populations cellulaires d'un follicule en fonction des conditions d'environnement hormonal, mais aussi de coupler entre elles les trajectoires folliculaires, pour rendre compte du rétrocontrôle exercé par l'ensemble de la population de follicules en développement sur les niveaux de FSH.

Ce modèle EDP a fait l'objet d'un ensemble de résultats théoriques. Dans [40], les auteurs ont étudié un problème d'atteignabilité, défini à partir de la formulation du modèle sous forme de courbes caractéristiques, pour délimiter les conditions initiales compatibles respectivement avec un état final de type ovulatoire ou atrétique. Il s'est avéré que ces conditions initiales sont peu distinguables et que le contrôle est déterminant dans la divergence des trajectoires ovulatoires ou atrétiques.

Dans le même article, les auteurs avaient avancé des arguments basés sur les traces des solutions pour aborder la question du caractère bien posé du modèle. Cette

question a ensuite été étudiée rigoureusement dans [71], en utilisant un argument de type point fixe combiné à la méthode des caractéristiques. Ce résultat assure, dans un cadre général, l’existence et l’unicité de la solution du problème, sans hypothèses simplificatrices. Un problème de contrôle optimal a ensuite été étudié sur une expression un peu simplifiée du modèle (contrôle s’exerçant en boucle ouverte sur les termes de vitesse, omission du second membre) [24]. Les résultats montrent qu’il existe au moins une stratégie de contrôle de type bang-bang, appliquée sur les termes de vitesse, qui permette de maximiser la maturité folliculaire (moment d’ordre un en maturité) en un temps final fixé. Ces résultats, qui généralisent en quelque sorte ceux qui avaient été obtenus sur le modèle compartimental [21], sont établis pour une formulation du modèle exprimée sous la forme d’une somme (possiblement infinie) de masses de Dirac, et restent à l’état de conjecture pour la formulation EDP. Une autre approche s’est focalisée sur le phénomène de compétition entre follicules, envisagé sous l’angle de la théorie des jeux [60]. En se basant sur des résultats de convergence asymptotique de la maturité, et en formulant quelques hypothèses simplificatrices (une seule phase pour le cycle cellulaire et mitose distribuée), l’auteur parvient à réduire le modèle originel, d’abord de 2D en 1D, puis comme un système d’ODEs couplées et fortement non linéaires. L’étude de ce système permet d’étudier la pression des autres follicules sur une trajectoire folliculaire et de classer les follicules dans différentes catégories (“saved”, “sensitive” et “doomed”).

Ce modèle a aussi fait l’objet d’études numériques exposées dans [39] et [47], qui se basent sur des schémas numériques de type volumes finis, implémentés jusqu’ici dans l’environnement académique Bearclaw (Boundary Embedded Adaptive Refinement Conservation LAW package) [55]. Non seulement ces approches ont dû proposer des solutions alternatives pour éviter de traiter directement les conditions de transmission discontinues (par exemple en les régularisant), mais elles ont été confrontées à un problème de temps de calcul pénalisant, qui constituait un obstacle majeur à la réalisation de simulations en grand nombre, ou de grande dimension (c’est-à-dire avec des tailles de cohorte folliculaire réalistes), pourtant indispensables à l’exploitation complète du modèle en terme d’interprétation biologique, que ce soit pour calibrer les valeurs de paramètres en fonction des connaissances disponibles ou pour tester différents scénarios de sélection.

Ce sont ces défis numériques que nous proposons de relever dans cette thèse, en développant une stratégie combinant l’élaboration d’un schéma numérique dédié, la construction d’un maillage adaptatif compatible avec ce schéma et leur mise en œuvre dans un cadre de calcul intensif sur une machine parallèle.

## 1.2 Modèle multiéchelles

**Schéma fonctionnel (Block Diagram).** Le modèle EDP de processus de sélection des follicules ovariens peut être représenté sous la forme générale d’un schéma

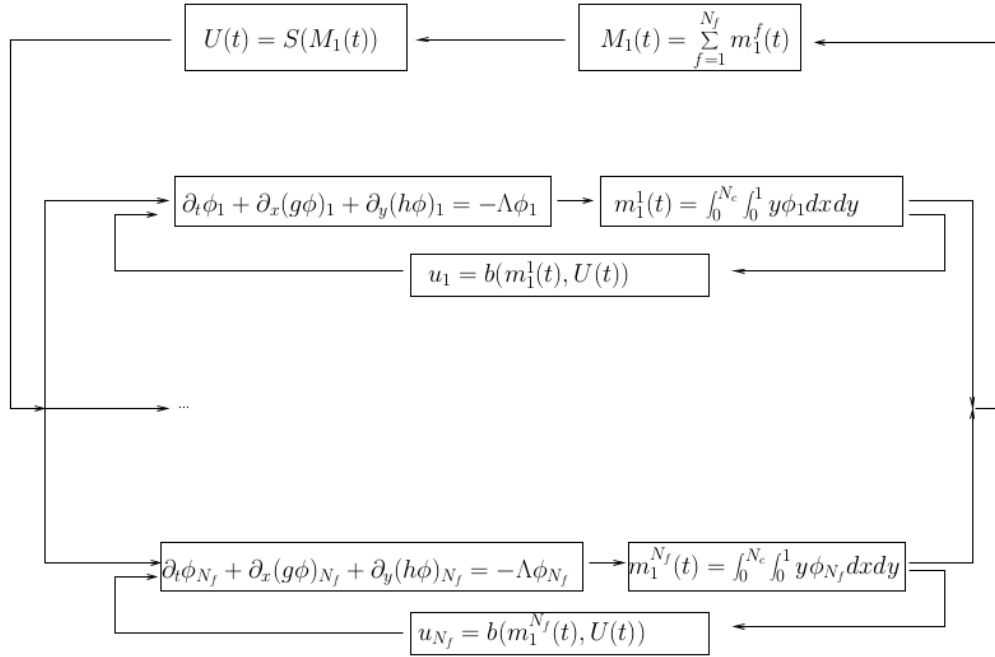


FIGURE 1.2 – Représentation du modèle multiéchelle du processus de sélection des follicules ovariens sous la forme d’un schéma fonctionnel (block diagram). Au sommet, la boucle de rétro-contrôle global. Le chemin se divise ensuite, chaque sous-chemin correspondant à un follicule, et comportant une boucle de rétro-contrôle au niveau mésoscopique. Puis les chemins se réunissent à nouveau pour définir la maturité globale, qui entre dans le terme de contrôle global.

fonctionnel, que l’on donne dans la Figure 1.2. Les paragraphes suivants sont consacrés à la description des différentes composantes de ce modèle.

### 1.2.1 Inconnues du problème

**Cellules de granulosa.** Parmi les différentes couches qui constituent la paroi des follicules ovariens se trouve la couche de cellules de granulosa. Au cours du temps les cellules en prolifération gagnent en maturité. Passé un certain seuil, elles se différencient et sortent du cycle cellulaire (voir Figure 1.3). Dès lors, il n’y a plus de mitose, et les cellules ne peuvent que vieillir. Au stade de prolifération chaque cycle peut lui-même être décomposé en deux phases : la phase G1, où les cellules se préparent à l’apoptose ou à la différenciation, et où elles sont sensibles à l’hormone FSH, et la phase SM de préparation à la division cellulaire, où au contraire les cellules sont insensibles à la FSH. A chaque fin de cycle G1-SM, la cellule entre en mitose, et une cellule mère se divise en deux cellules filles.



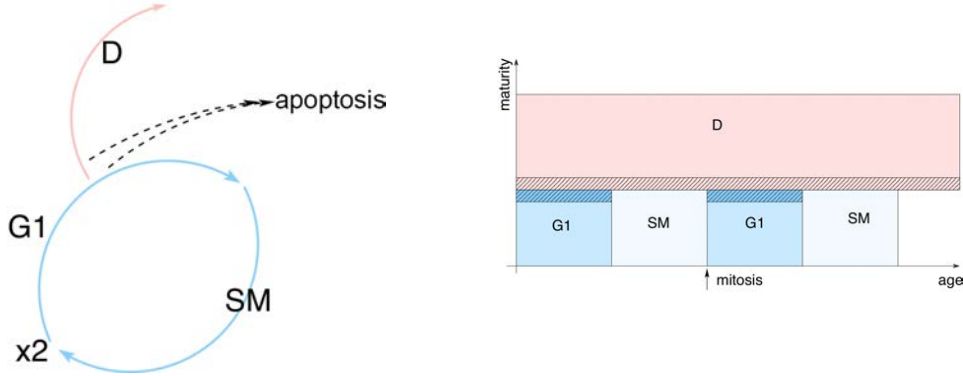


FIGURE 1.3 – Plan age-maturité. En abscisse, l'âge (position dans le cycle cellulaire), en ordonnée la maturité cellulaire. A chaque sous-domaine correspond une phase du cycle cellulaire des cellules de granulosa. Les zones hachurées correspondent aux zones où les cellules peuvent disparaître par apoptose, pendant les phases G1 ou D.

**Domaine de calcul 2D.** Le domaine de calcul est un domaine compact du plan  $R^2$ , appelé plan âge-maturité (voir Figure 1.3). L'axe des abscisses représente l'âge, et permet de repérer la position des cellules dans leur cycle cellulaire. Celui des ordonnées représente la maturité cellulaire. Cette définition recouvre plusieurs notions : sensibilité à la FSH et à la LH, sensibilité à l'apoptose, position par rapport à la sortie du cycle cellulaire. Le domaine est divisé en sous domaines, chacun correspondant à une phase des cellules de granulosa.

**Inconnues.** Dans ce modèle, les inconnues sont les densités de cellules de granulosa des follicules

$$\begin{cases} \phi_1(a, \gamma, t), \\ \dots \\ \phi_{N_f}(a, \gamma, t). \end{cases}$$

avec  $N_f$  le nombre de follicules.

### 1.2.2 Système de lois de conservations

**Dynamique d'un follicule.** A chaque follicule  $f$  est associé une dynamique, donnée par :

- une loi de conservation avec flux contrôlés

$$\frac{\partial \phi_f}{\partial t} + \frac{\partial g_f(u_f) \phi_f}{\partial a} + \frac{\partial h_f(u_f) \phi_f}{\partial \gamma} = -\lambda(U) \phi_f(a, \gamma, t),$$

- une condition initiale :

$$\phi_f(a, \gamma, 0) = \phi_f^0(a, \gamma),$$

- des conditions aux limites :

- condition de périodicité sur l'axe des abscisses  $\phi_f(0, \gamma, t) = \phi_f(N_c, \gamma, t)$  ;

## Chapitre 1. Introduction

---

- densité nulle  $\phi_f = 0$  aux bords inférieur  $\gamma = 0$  et supérieur  $\gamma = 1$  du domaine ;
- des termes de contrôle :
  - $u_f$  contrôle propre au follicule  $f$ , dit mésoscopique ;
  - $U$  contrôle commun à tous les follicules, dit global ou macroscopique ;
- des conditions de transmission entre les sous domaines.

**Conditions de transmission.** Les fonctions de vieillissement  $g_f$  et de maturation  $h_f$  sont définies par sous-domaines, et sont en général discontinues aux interfaces séparant ces derniers. Pour que le problème soit mathématiquement bien posé, il faut imposer des conditions de transmission (voir [71]).

Ces conditions, qui peuvent s'exprimer en 1D sans perte de généralité, prennent la forme suivante (voir Figure 1.4)

$$f(\phi(x_s^+, t)) = \psi(f(\phi(x_s^-, t))) \quad \text{frontière en } x_s.$$

Le flux à droite de l'interface positionnée en  $x_s$ ,  $f(\phi(x_s^+, t))$ , est une fonction  $\psi$  du flux à gauche de l'interface,  $f(\phi(x_s^-, t))$ .

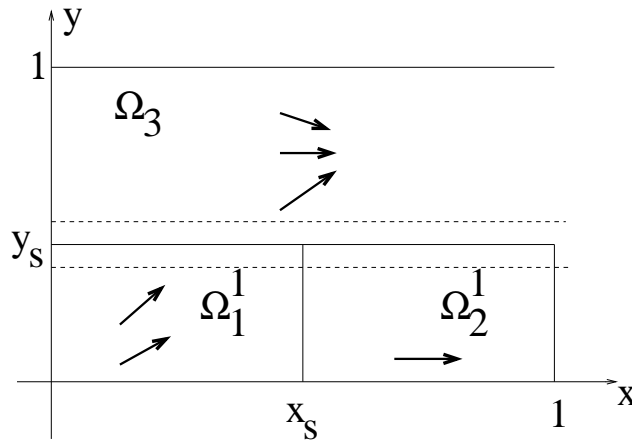


FIGURE 1.4 – Interfaces et conditions de transmission (Figure extraite de [8]). Le champ de vitesse est régulier sur chaque sous-domaine  $\Omega_1, \Omega_2, \Omega_3$ , mais est potentiellement discontinu aux interfaces séparant ces derniers (en  $x_s$  et en  $y_s$ ). Afin de fermer mathématiquement le problème, il est nécessaire d'imposer des conditions de transmission.

Précisons les conditions de transmission dans notre cas.

- continuité de flux entre les phases G1 et SM :  $\psi(F) = Id(F)$ ,
- doublement de flux (mitose en cinétique cellulaire) entre les phases SM et G1 :  $\psi(F) = 2F$ ,
- paroi étanche entre SM et D :  $\psi(F) = 0$ .

**Vitesse de propagation selon l'axe des abscisses.** La fonction  $g_f$  définit la vitesse de vieillissement (propagation selon l'axe des abscisses). En zone G1, où les

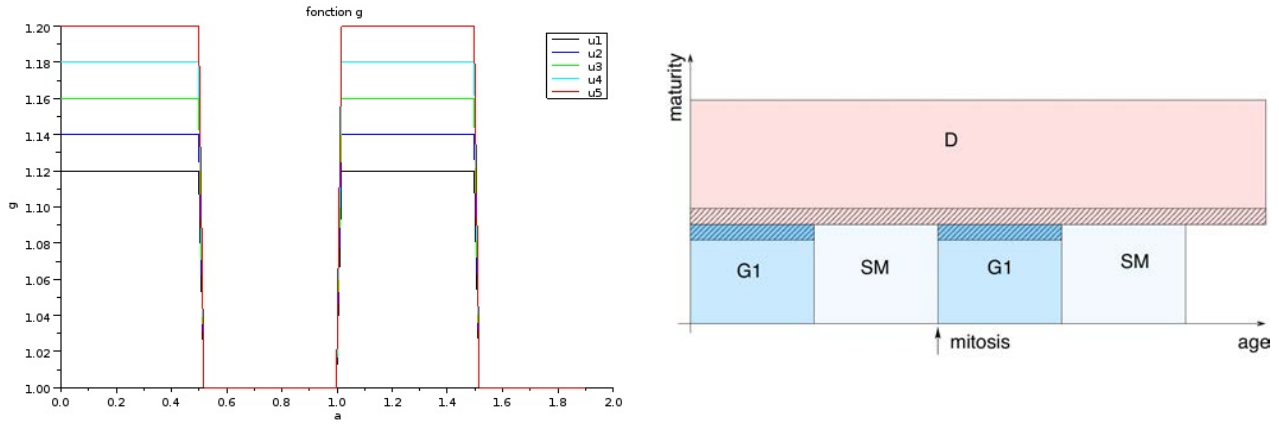


FIGURE 1.5 – Fonction de vieillissement  $g_f$  sur une coupe horizontale (à gauche), en zone de prolifération, du domaine de calcul 2D (à droite), pour différentes valeurs de contrôle mésoscopique  $u_f$ . Un contrôle élevé entraîne une vitesse élevée, favorisant la production de cellules. En effet, dans ce cas, les cellules ont alors un cycle plus court.

cellules de granulosa sont en croissance, cette vitesse est contrôlée, afin de modéliser la sensibilité des cellules à la FSH, où les paramètres  $g_1$  et  $g_2$  peuvent varier d'un follicule à l'autre. En phase SM et D, ces dernières étant insensibles à la FSH, cette vitesse est constante et égale à un, les variables âge et temps étant alors identiques.

$$g_f(a, \gamma, u_f) = \begin{cases} g_1 u_f + g_2 & \text{en phase } G1, \\ 1 & \text{en phase } SM \cup D. \end{cases}$$

Sur la Figure 1.5, on représente une coupe horizontale (à gauche) du domaine de calcul 2D (à droite), pour différentes valeurs de contrôle mésoscopique  $u_f$ . Plus le contrôle est élevé, plus la vitesse est élevée, favorisant la production de cellules de granulosa.

**Vitesse de propagation selon l'axe des ordonnées.** La vitesse de maturation des cellules de granulosa peut être définie comme l'évolution de la réponse d'une cellule à la FSH en terme de synthèse d'AMP cyclique. En phase SM, en notant  $\gamma$  la maturité d'une cellule de granulosa, on a forcément

$$\frac{d\gamma}{dt} = 0,$$

elle se prépare à la mitose. Les cellules sont insensibles à la FSH, et n'évoluent pas en maturation. En phase G1 et D, le niveau évolue selon la sensibilité de la cellule à FSH. Dans [28], la dynamique du couplage entre les récepteurs à FSH et la synthèse de l'AMP cyclique a été modélisée par un système d'EDO non linéaires. L'étude de l'état d'équilibre stable de ce système a conduit, moyennant quelques simplifications, à la fonction de maturation suivante

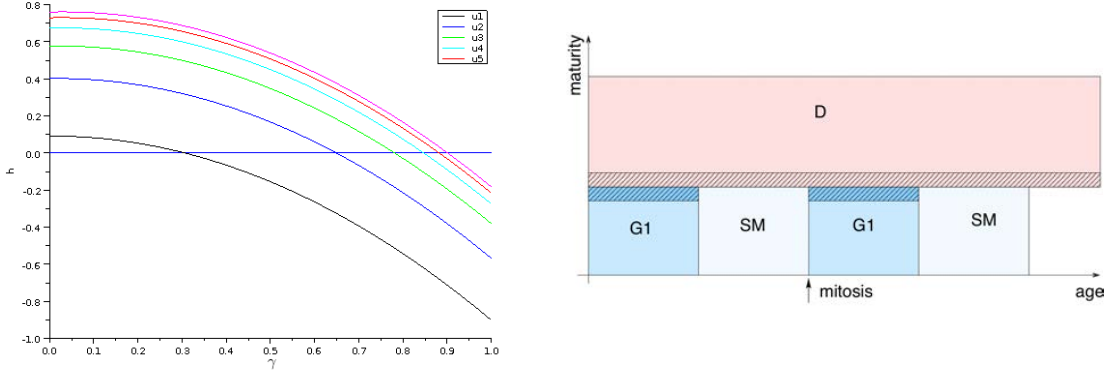


FIGURE 1.6 – Fonction de maturation  $h_f$  sur une coupe verticale (à gauche) du domaine de calcul (à droite), pour différentes valeurs de contrôle mésoscopique  $u_f$ , croissante de  $u_1$  à  $u_5$ . Le contrôle mésoscopique règle la position du zéro de  $h_f$ , qui définit une maturité asymptotique.

$$h_f(a, \gamma, u_f) = \begin{cases} \tau_f(-\gamma^2 + (c_1\gamma + c_2)(1 - \exp(\frac{-u_f}{\bar{u}}))) & \text{en phases } G1 \cup D, \\ 0 & \text{en phase } SM. \end{cases}$$

Sur la Figure 1.6, on a représenté la fonction  $h_f$  sur une coupe verticale (à gauche) du domaine de calcul (panel de droite), pour différentes valeurs de contrôle. A  $\gamma$  constant,  $h_f$  est une fonction croissante de  $u_f$ , et bornée. Un contrôle élevé favorise une maturation rapide.

**Terme source : perte par apoptose.** La perte de cellules par apoptose est modélisée par un terme source de la forme

$$-\Lambda(a, \gamma, U)\phi_f(a, \gamma, t),$$

avec

$$\Lambda(a, \gamma, U) = K \exp\left(-\frac{(\gamma - \gamma_s)^2}{\bar{\gamma}}\right) \frac{(U_{\max} - U)}{U_{\min}}.$$

Il est composé de plusieurs termes : le paramètre  $K$  règle l'intensité de l'apoptose, le terme en exponentielle assure la nullité du terme source loin de l'interface G1-D, et le terme de contrôle modélise la réponse au contrôle global. Lorsque le contrôle global est maximal, la fonction  $\Lambda$  est nulle, ce qui traduit un environnement favorable à la croissance cellulaire (et donc folliculaire), car il n'y a pas de perte par apoptose. Inversement, lorsque le contrôle est minimal, la fonction  $\Lambda$  atteint un maximum, ce qui traduit un environnement défavorable pour l'accroissement cellulaire, avec beaucoup de pertes par apoptose.

Sur la Figure 1.7, on a représenté la fonction  $\Lambda$  sur une coupe (à gauche) du domaine de calcul (à droite), pour différentes valeurs de contrôle global  $U$ .

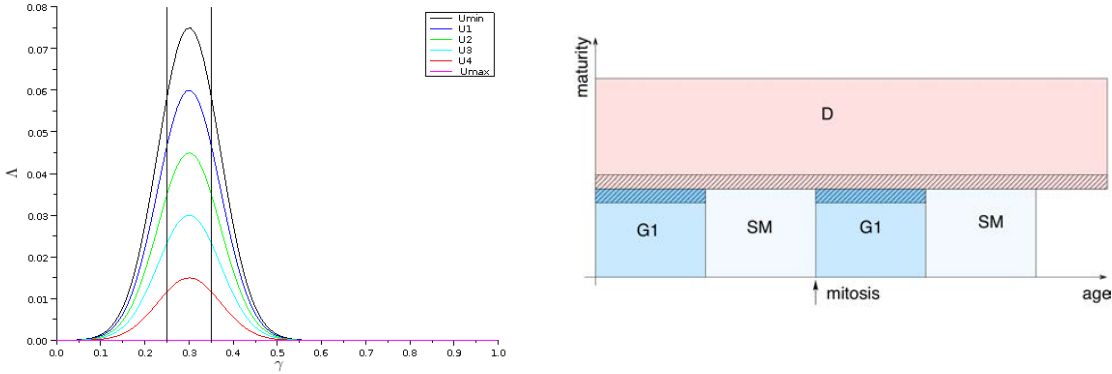


FIGURE 1.7 – Fonction de perte par apoptose  $\Lambda$  sur une coupe (à gauche) du domaine de calcul (à droite) pour différentes valeurs du contrôle global  $U$ . A contrôle maximal, ce terme est nul. Il s’active lorsque le contrôle baisse, ce qui entraîne une perte de cellules par apoptose.

Jusqu’à présent, nous avons parlé de dynamique pour chaque follicule séparément, ainsi que de contrôle, mais nous n’avons pas précisé comment s’opère le couplage entre les follicules. C’est l’objet du paragraphe suivant.

### 1.2.3 Contrôle non local

**Couplage des follicules par les moments de la solution.** Le couplage des follicules se fait via les termes de contrôle, définis par les moments des solutions. On définit les moments d’ordre 1 en  $\gamma$  pour chaque follicule, que l’on appelle maturités folliculaires (échelle mésoscopique),

$$\begin{cases} m_1^1(t) = \iint_{[0, N_c] \times [0, 1]} \gamma \phi_1(t, a, \gamma) d\gamma da, \\ \dots \\ m_1^{N_f}(t) = \iint_{[0, N_c] \times [0, 1]} \gamma \phi_{N_f}(t, a, \gamma) d\gamma da. \end{cases}$$

Mathématiquement, la maturité folliculaire va représenter la position verticale moyenne du nuage de densité. Biologiquement, elle quantifie la production d’œstradiol et d’inhibin de chaque follicule, ainsi que la sensibilité à la FSH et à la LH.

A partir des maturités folliculaires, on peut définir la maturité ovarienne (échelle macroscopique)

$$M_1(t) = \sum_{f=1}^{N_f} m_1^f(t).$$

Cette quantité permet de modéliser la quantité d’œstradiol et d’inhibin sécrétée par l’ovaire, information reçue par l’axe hypothalamus-hypophyse.

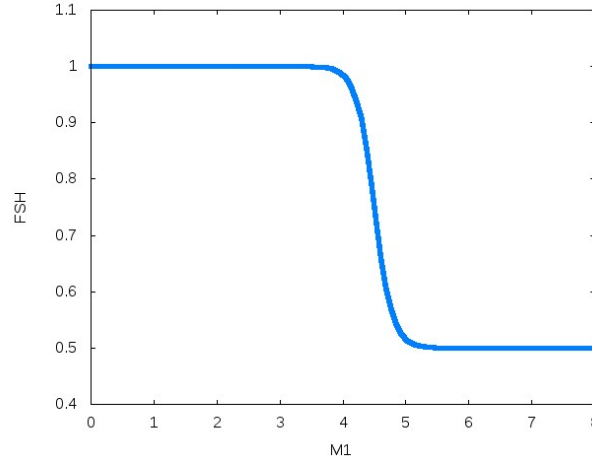


FIGURE 1.8 – *Contrôle hormonal : niveau plasmatique de FSH en fonction de la maturité ovarienne. Tant qu'une maturité seuil n'est pas atteinte, le contrôle est maximal. A l'atteinte du seuil, le contrôle chute, vers une seconde valeur asymptotique.*

**Contrôle macroscopique : taux plasmatique de FSH.** A partir de la maturité ovarienne, on définit le terme de contrôle global

$$U(t) = \frac{U_{\max} - U_{\min}}{1 + \exp(c(M_1(t) - \bar{M}))} + U_{\min},$$

Ceci définit un premier couplage entre les follicules, via le terme source de perte par apoptose.

Sur la Figure 1.8, on représente le niveau plasmatique de FSH (terme  $U$ ) en fonction de la maturité globale  $M_1$ . Au départ, le contrôle est maximal, ce qui favorise la croissance folliculaire, et entraîne donc un accroissement de la valeur de  $M_1$ . une fois une valeur seuil atteinte, le niveau plasmatique de FSH décroît vers une valeur minimale, activant le terme de perte par apoptose (avec une intensité de ce terme commune à tous les follicules).

**Contrôle mésoscopique : vascularisation et accès du follicule à la FSH.**

Une fois le contrôle global défini, on peut définir le contrôle mésoscopique, propre à chaque follicule. Ce dernier représente la vascularisation, la capacité d'accès à la ressource FSH. On définit pour cela une fonction de vascularisation  $b_f$  :

$$b_f(m_f) = \min \left( b_1 + \frac{e^{b_2 m_1(f,t)}}{b_3}, 1 \right),$$

puis les termes de contrôle mésoscopiques, qui sont des fractions du terme de contrôle macroscopique

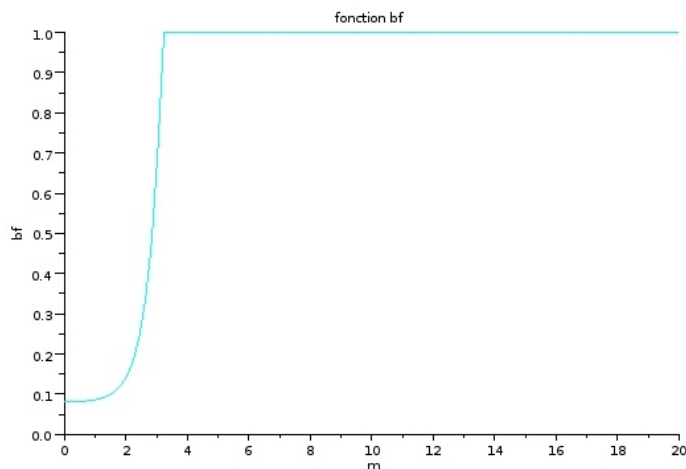


FIGURE 1.9 – *Fonction de vascularisation  $b_f$ . Fonction croissante de la maturité folliculaire, jusqu'à l'atteinte de la valeur 1, signe d'un accès sans contrainte au niveau plasmatique de FSH.*

$$u_f(t) = b_f(m_1^f)U(t)$$

Sur la Figure 1.9, on représente la vascularisation  $b_f$  en fonction de la maturité folliculaire  $m_1^f$ . La vascularisation est une fonction croissante de la maturité folliculaire, jusqu'à atteindre la valeur 1, signifiant que le niveau de FSH mésoscopique est égal au niveau macroscopique de FSH.

**Temps d'arrêt.** Le processus prend fin lorsque le niveau de maturité globale,  $M_1(t)$ , dépasse un seuil, que l'on note  $M_o$ . Cela définit un temps d'arrêt

$$t_{final} = \inf\{t, M_1(t) \geq M_o\}.$$

Les follicules sont alors triés en deux classes, suivant leur niveau de maturité folliculaire. Ceux dont la maturité dépasse un seuil, noté  $M_f$ , sont dits ovulatoires, les autres étant atrétiques.

### 1.3 Apports de la thèse

Dans cette section, nous passons en revue les résultats obtenus dans le cadre de la thèse.

#### 1.3.1 Numerical simulation of the selection process of the ovarian follicles

La première étape de la thèse fut l'écriture et l'implémentation d'un schéma numérique dédié au problème. Le point de départ était le suivant : le modèle est bien posé,

et des simulations numériques faites précédemment ([39], [47]) ont permis d'identifier des valeurs de plages de paramètres donnant des sorties plausibles. Ces simulations étaient réalisées avec le logiciel BearClaw ([55]), utilisant divers schémas volumes finis en maillage adaptatif : CTU ([32]), et F-Wave ([9]).

Le principal écueil de cette stratégie numérique était la prise en compte des conditions de couplage entre les sous-domaines, qui était contournée en régularisant les discontinuités inhérentes au modèle dans les coefficients de vitesse et du terme source. Cette approximation introduisait de l'imprécision numérique et pénalisait énormément les performances en temps de calcul, car elle nécessitait un raffinement excessif du maillage dans les zones de régularisation.

À partir de ce constat, un des buts de la thèse était de définir puis implémenter une méthode numérique dédiée pour ce type de modèle incluant des équations couplées de manière non locale et posées sur un domaine de calcul avec des conditions de transmissions non triviales aux interfaces entre les sous-domaines.

Nous avons pensé dès le début à l'implémentation de la méthode numérique sur une architecture parallèle, en nous inspirant du schéma fonctionnel représenté sur la Figure 6.1, qui met en relief le comportement "en parallèle" des follicules au sein de l'ovaire.

Nous sommes repartis des équations du modèle. Ces dernières sont des équations de transport scalaires couplées de manière non locales, définies sur une géométrie simple. Nous avons adopté dans un premier temps un schéma volumes finis upwind sur des mailles cartésiennes uniformes.

En tirant parti du couplage non local des équations, via les moments de la solution, nous avons proposé un algorithme de type SIMD Single Instruction, Multiple Data). On peut citer deux principaux résultats obtenus (voir Figure 1.10). Le schéma proposé est bien convergent (voir Figure 1.10, panel de gauche), et la méthode de parallélisation donne de très bons résultats en test de scalabilité faible (voir Figure 1.10, panel de droite). Il est important de noter que, dans le cadre du test de scalabilité faible effectué pour cette figure, la distribution des paramètres est la même pour tous les processeurs.

Ces résultats ont fait l'objet d'une publication dans ESAIM Proceedings ([6]), et constituent le chapitre 1.

### 1.3.2 A numerical method for transport equations with discontinuous flux functions : application to mathematical modeling of cell dynamics

Une fois obtenue une méthode de calcul multiéchelle à l'ordre 1, nous avons cherché à augmenter la précision du schéma. Pour cela, nous avons utilisé des techniques de limitation de flux, pour monter en précision en espace, et des méthodes de type Runge-Kutta, pour monter en ordre en temps.

Les méthodes de limitation de flux consistent à écrire les flux numériques aux interfaces séparant les mailles sous la forme

$$F = F^L + l(r)(F^H - F^L)$$



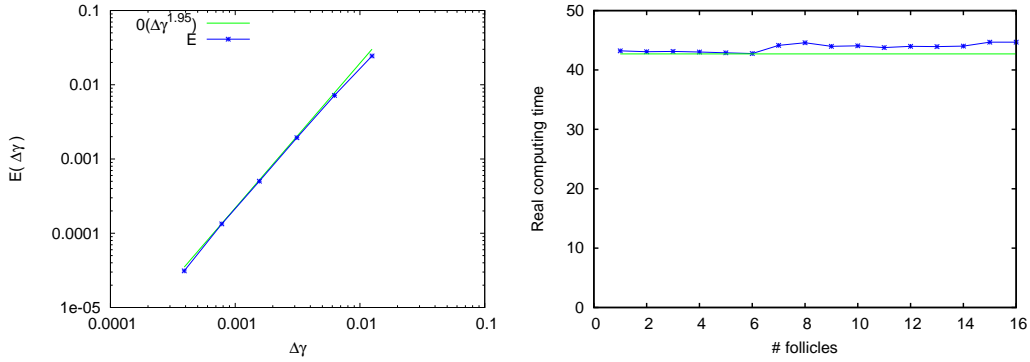


FIGURE 1.10 – *Figures extraites de [6]. Test de convergence en zone G1 (à gauche), et test de scalabilité faible (à droite). Le schéma est bien convergeant, loin des interfaces, et la parallélisation est excellente.*

avec  $F^L$  un flux numérique d'ordre 1,  $F^H$  un flux numérique d'ordre élevé,  $l$  une fonction limiteur et  $r$  un rapport, indicateur de régularité locale. Il est connu que les schémas d'ordre 1 sont diffusifs, et ont tendance à trop lisser les solutions. Les schémas d'ordre élevé n'ont pas ce problème, mais ils génèrent des oscillations aux voisinages des discontinuités. La limitation de flux (ou de pente), permet de pallier ce problème. L'indicateur de régularité est proche de 1 dans les régions où la solution varie peu, et proche de 0 ou de l'infini dans les régions à fort gradient. La fonction limiteur vaut 1 quand  $r = 1$  et tend vers 0 en 0 et en l'infini. Le schéma ainsi construit est hybride. Il est à l'ordre 1 proche des discontinuités, à l'ordre élevé dans les régions où la solution est régulière.

Cette méthode ne fonctionne cependant pas aux interfaces non conservatives de type mitose dans notre modèle. Ces interfaces créent localement des discontinuités, ce qui dégrade l'ordre du schéma à l'ordre 1 (vérifié numériquement). Nous avons proposé une solution pour éviter ce problème. Tout d'abord, on ne calcule pas un, mais deux flux par interface (un flux à gauche, et un flux à droite). Selon le signe de la vitesse, on calcule d'abord l'un des deux flux, puis le second, en prenant en compte la condition de transmission  $\psi$ . Pour que le flux soit calculé à l'ordre élevé, on injecte dans le limiteur non plus la solution, mais la solution via un changement de variable, qui permet de manipuler une quantité conservée. Typiquement, à gauche de l'interface, le schéma numérique ne voit pas la condition de transmission (on donne aux limiteurs des quantités non doublées à l'interface). Pour avoir pleinement l'ordre 3, dans notre cas, il faut passer au limiteur des quantités conservées, sur les interfaces de doublement, mais aussi sur les interfaces directement voisines, à gauche et à droite.

Afin de valider notre méthode, nous avons créé des cas tests directement inspirés du modèle EDP de processus de sélection des follicules ovariens. Nous avons isolés quatre cas représentatifs. Une fois ces problèmes posés, nous avons proposé des solutions exactes, Pour ce faire, l'idée est d'adapter la méthode des caractéristiques (voir Figure 1.11). Loin des interfaces, on peut remonter les caractéristiques pour

## Chapitre 1. Introduction

---

atteindre l'axe  $t = 0$  Lorsqu'une caractéristique atteint une interface, on relève le temps d'atteinte, puis on utilise la condition de transmission pour passer de l'autre côté de l'interface, et enfin on remonte la nouvelle caractéristique jusqu'à l'axe  $t = 0$ .

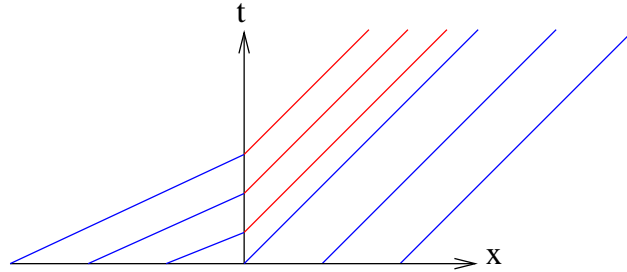


FIGURE 1.11 – *Caractéristiques correspondant à deux équations de transports linéaires à coefficients constants avec un contraste de vitesse en  $x = 0$ , avec  $v > 1$  pour  $x < 0$  et  $v < 1$  pour  $x > 0$ . La solution est constante sur les droites caractéristiques.*

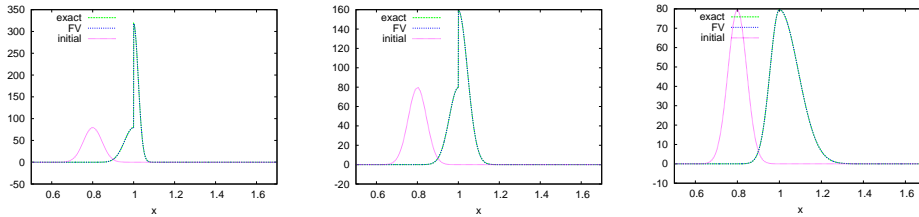
Sur la Figure 1.12, on compare la solution exacte d'un problème d'équation de transport avec doublement de flux à l'interface avec la solution approchée utilisant notre méthode, et ce pour différentes valeurs de contraste de vitesse (accélération, vitesse constante ou ralentissement de la densité au passage de l'interface). À chaque fois, on représente à gauche du domaine la densité au temps initial (il s'agit d'une gaussienne). On note que les courbes se superposent, ce qui semble indiquer que notre méthode fonctionne qualitativement.

Ce résultat est confirmé sur la Figure 1.13, panel b), où l'on fait une courbe d'erreur au temps final en fonction du pas de discrétisation. Il y a plusieurs courbes, indexées par  $g_R$ , qui représente la vitesse à droite de l'interface, en supposant que la vitesse à gauche est de 1. On note que l'ordre numérique de convergence est supérieur à 2. Sur le panel a), on trace la masse (moment d'ordre 0 de la densité) au cours du temps. On note qu'elle a bien doublé.

On pourra consulter le chapitre 2 pour plus de détails.

Ces résultats ont fait l'objet d'une publication dans *SIAM Journal of Scientific Computing* ([7]).

Une fois obtenu un schéma numérique parallèle, d'ordre élevé, même aux interfaces, nous avons cherché à accélérer encore les calculs, en utilisant des techniques d'adaptation de maillage. En effet, on remarque que les solutions du système sont des fonctions de densités très localisées dans le domaine de calcul. Ainsi, l'utilisation d'un maillage uniforme n'est pas optimale. En effet, le maillage est très raffiné dans des zones où la densité est, et reste, la plupart du temps, nulle. Cela fait beaucoup de calculs inutiles, et beaucoup de mémoire utilisée pour rien. Nous avons utilisé des techniques de multirésolution (voir [10], [30]), pour faire des simulations sur des maillages adaptatifs en temps, localement raffinés dans les zones où la solution varie. Nous avons développé un code de calcul dédié, en C++/MPI appelé CodeFollicleMR. Sur la Figure 1.14, on donne l'exemple d'un maillage adaptatif obtenu avec



a) Ralentissement au passage de l'interface    b) Pas de saut de vitesse au passage de l'interface.    c) Accélération au passage de l'interface.)

FIGURE 1.12 – *Cas test 1D (Figures extraites de [7]) : doublement de flux au passage de l'interface  $x_s = 0.5$ . Profils de la densité  $\phi(x,t)$  au temps initial et lorsqu'elle passe l'interface séparant les zones G1 et SM ( $t = 0.2$ ).*

le code.

Une notice utilisateur est disponible en Annexe A de la thèse.

Une notice programmeur, détaillant la structure du code source, les formats de fichier d'entrée et de sortie, est disponible en Annexe C de la thèse.

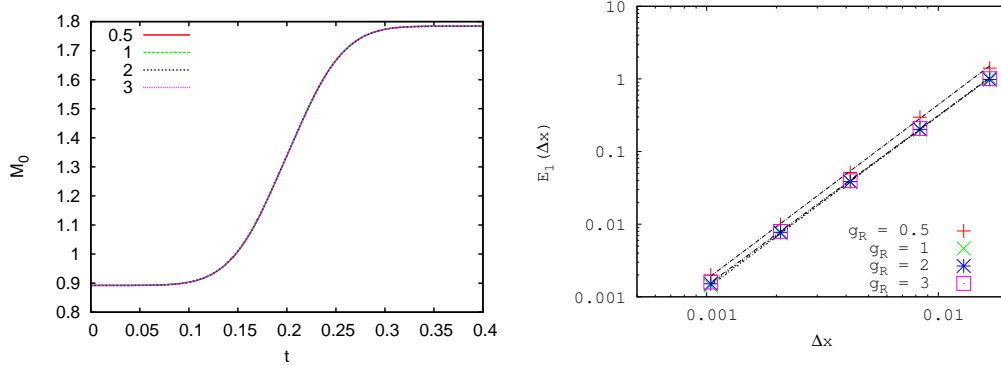
Après le problème du maillage et de la méthode de calcul s'est posé le problème de la visualisation des résultats. Le problème étant assez spécifique, nous avons proposé un code de visualisation dédié. Il a fallu pour cela définir un cahier des charges, qui s'est précisé avec le temps, au fil des échanges avec les biologistes.

Nous avons besoin de visualiser des résultats sur deux niveaux d'échelle différents : micro et macro.

Au niveau micro, on veut visualiser les densités de chacun des follicules (champs de scalaire dans un domaine borné pour chaque follicule). Une première idée fut de pouvoir visualiser dans une fenêtre les résultats pour un follicule, avec une option pour passer d'un follicule à l'autre. Cette méthode est pratique, mais n'est pas adaptée à la comparaison inter follicules. Nous avons alors proposé une visualisation globale et simultanée de tous les follicules, en affichant les densités les unes au dessous des autres.

Pour le niveau macro, nous voulons visualiser des courbes temporelles (masses, maturités, contrôle macroscopique et contrôles mésoscopiques, ...). Nous avons proposé de visualiser ces quantités simultanément à l'aide du logiciel gnuplot (voir par exemple Figure 1.19).

Le code couleur au niveau micro a soulevé plusieurs questions, et a beaucoup changé au cours du temps. Au départ, il était redéfini à chaque pas de temps, pour chaque follicule, à partir du maximum de la densité au temps courant. Afin de comparer plus facilement les follicules entre eux, nous avons défini un code couleur valable pour tous les follicules, à tous les pas de temps, basé sur la densité maximale, tous follicules confondus, au cours de la simulation. Le problème est que cette méthode favorise la visualisation des densités pour des temps proches du temps final. Nous avons alors proposé un code couleur adapté à notre problème de dynamique de



a) Masse en fonction du temps.      b) Erreur en norme  $L_1$  en fonction de la taille de maille  $\Delta x$

FIGURE 1.13 – *Cas test 1D (Figures extraites de [7]) : doublement de flux au passage de l'interface  $x_s = 0.5$ . La masse (à gauche) a doublé lorsque toute la densité a passé l'interface, comme attendu. L'erreur en norme  $L_1$  (à droite) en fonction de la taille des mailles décroît à zéro avec un taux de 2.4.*

population. Il est non linéaire, et permet de visualiser tous les follicules, à tout temps, sans favoriser le temps final.

Sur la Figure 1.15, on donne un exemple de visualisation obtenu avec le code de visualisation dualMR (une notice utilisateur est disponible en Annexe B de la thèse). On affiche avec le même code couleur les densités de cellules de deux follicules dans le plan âge (abscisse  $x$ ), maturité (ordonnée  $y$ ) au même instant  $t$ . Les paramètres définissant les fonctions de vitesses  $g_f$  et  $h_f$ ,  $f = 1, 2$  ont des valeurs différentes et les densités de cellules des deux follicules sont dans des régions différentes de l'espace fonctionnel. De plus, l'effectif cellulaire du follicule sur le panel du bas ( $M_0 = 20.46$ , en dessous à gauche du panel) est plus de trois fois plus grand que celui du follicule sur le panel du haut ( $m_0 = 6.33$ ).

### 1.3.3 Adaptive mesh refinement strategy for a non conservative transport problem

En l'absence de soin particulier la stratégie de maillage en présence de doublement de flux met en évidence un problème en temps long. En effet, le maillage adaptatif se raffine dans des zones où il n'y avait pas de densité significative (voir Figure 1.16, a) ).

Ce phénomène est dû au couplage entre l'adaptation de maillage par multirésolution et les conditions de doublement aux interfaces (modélisant le phénomène de mitose). En raison des opérations de raffinement et déraffinement successifs, la méthode adaptative introduit des perturbations par rapport à la solution calculée sur une grille uniforme. La Figure 1.17 visualise les différences entre les deux solutions dans le cas d'une équation de transport, pour une valeur exagérément grande du

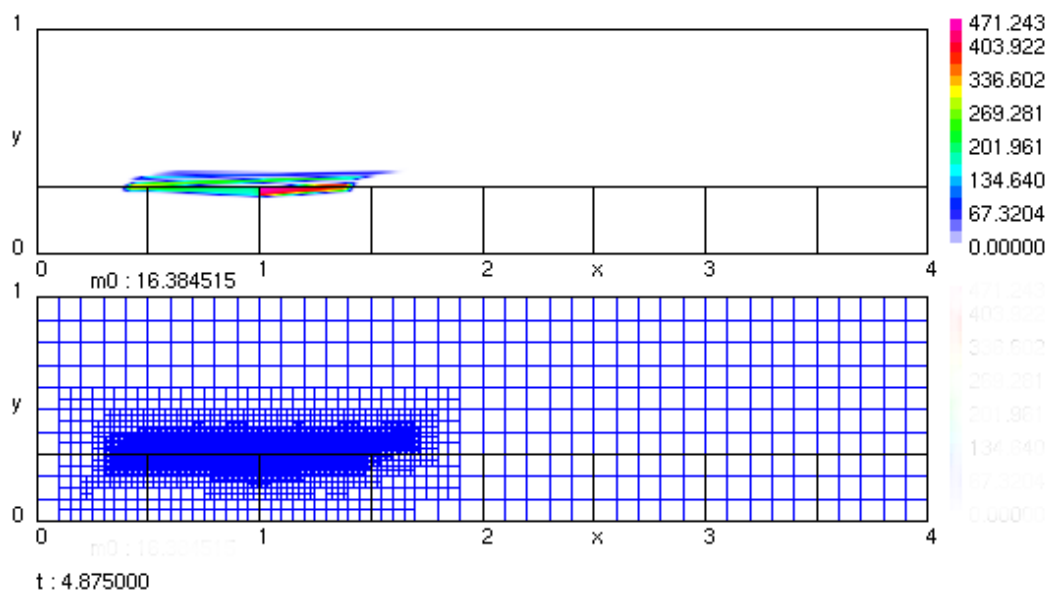


FIGURE 1.14 – *maillage adaptatif au temps  $t^n$  du plan âge-maturité obtenu avec CodeFollicleMR (et visualisé avec le logiciel dualMR). On représente la densité  $\phi_f$  d'un follicule  $f$  avec en abscisse, l'âge  $a$ , et en ordonnée, la maturité cellulaire  $\gamma$ . Le maillage est raffiné seulement dans les zones où la solution varie, ce qui économise beaucoup de calculs lorsque l'on utilise la méthode des volumes finis.*

paramètre de "seuillage" (les détails de la solution entre deux niveaux successifs de discrétisation sont négligés en dessous de ce seuil). Dans le cas conservatif où elle a été développée ([30]), ces perturbations restent contrôlées au cours du temps par un paramètre de seuillage.

Mais les conditions de doublement de flux agissent comme un amplificateur, et un bruit numérique, si petit soit-il au départ, est amplifié au cours du temps, jusqu'à être détecté par le maillage, si l'on attend assez longtemps. Nous avons proposé une méthode de réduction de bruit, en modifiant les conditions de transmission. Nous avons validé qualitativement, puis quantitativement cette méthode.

Sur la Figure 1.16 b), nous avons tracé les résultats de la même simulation que dans la Figure 1.16 a), en utilisant notre méthode de réduction de bruit. Les mailles parasites ont disparues.

Ces résultats ont fait l'objet d'une publication dans ESAIM Mathematical Modelling and Numerical Analysis (M2AN) ([8]), qui constitue le chapitre 3.

### 1.3.4 Calibration

La mise en place d'un outil de simulation performant et fiable fut une réelle avancée, ouvrant de nouvelles perspectives. Nous étions alors capables de faire des simulations sur un nombre de follicules réaliste, en un temps raisonnable. Cela nous amenait naturellement à l'étape de calibration des paramètres.

## Chapitre 1. Introduction

---

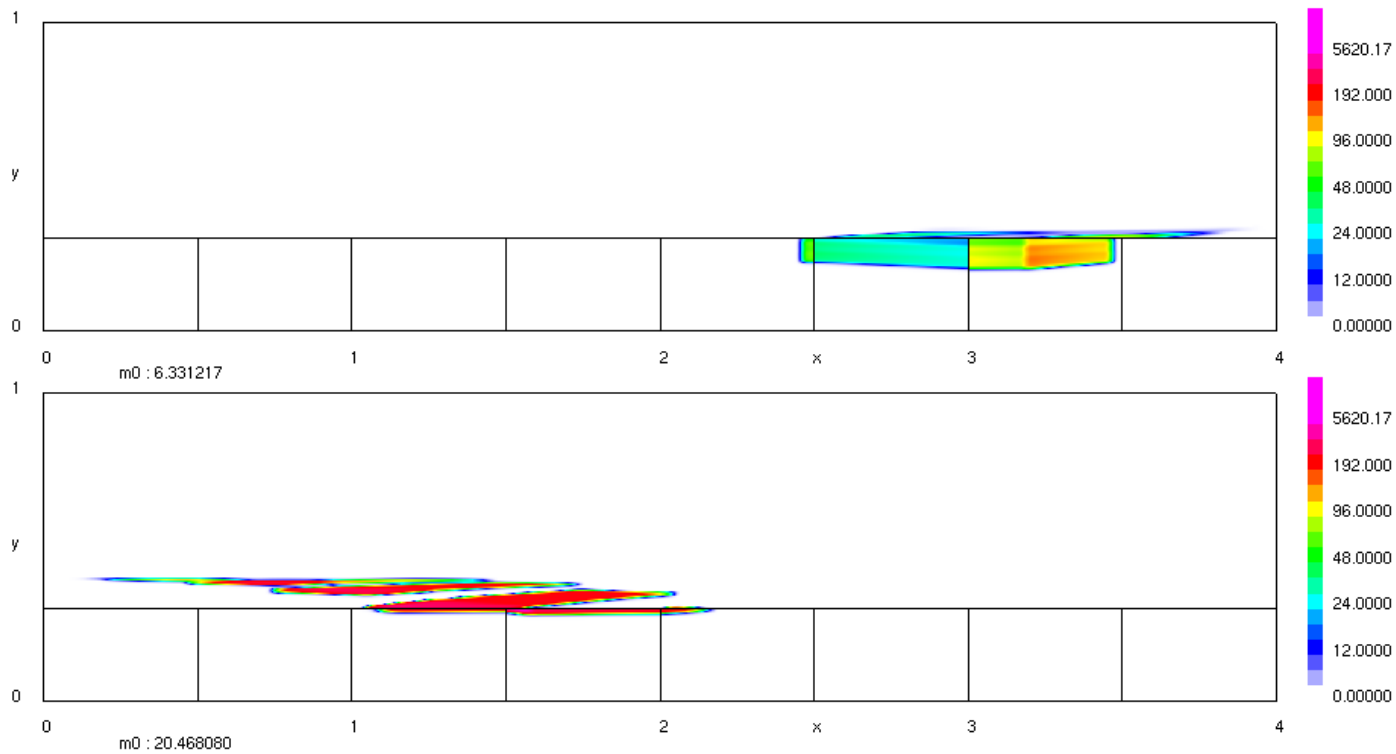
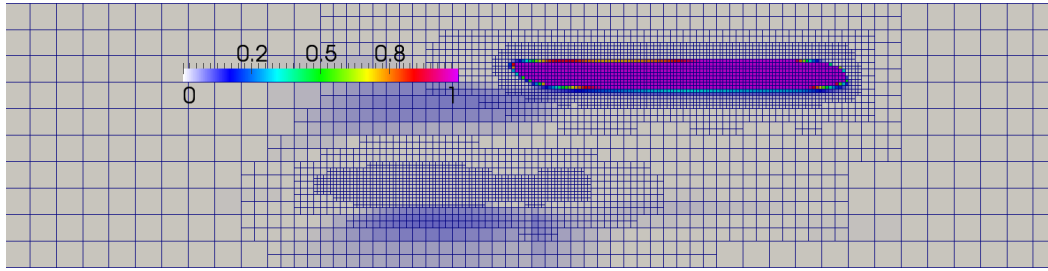


FIGURE 1.15 – Exemple de visualisation obtenue avec le code de visualisation C++ - OpenGL dualMR, dédié à la visualisation des résultats du modèle EDP de processus de sélection des follicules ovariens. On affiche les densités de deux follicules correspondant à des valeurs différentes des paramètres, dans le plan âge (abscisse  $x$ ), maturité (ordonnée  $y$ ) au même instant  $t$ . Le code couleur est adapté à la visualisation de notre problème : il est non linéaire, avec paliers, correspondant à différents doublements de population.

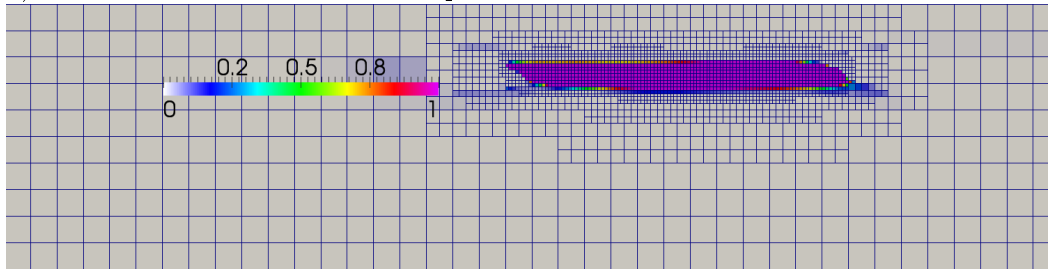
A la suite de nombreux échanges avec les biologistes, et en particulier Danielle Monniaux (INRA Tours) nous avons proposé plusieurs situations de référence.

Nous avons commencé par définir des contraintes permettant de réduire l'espace des paramètres, afin de respecter les spécifications biologiques. Puis, nous avons défini les notions de trajectoire ovulatoire et de trajectoire atrétique. Une première étape fut achevée en proposant une situation à un follicule ovulatoire seul. Nous avons utilisé des abaques de visualisation et des cubes de visualisation pour définir les paramètres réglant le comportement en temps long de la solution. Ensuite nous avons réglé les paramètres de vitesse, tout d'abord en introduisant des paramètres de travail permettant de régler plus facilement les fonctions  $g_f$  et  $h_f$ , puis par des méthodes de tir.

Au cours des simulations nous avons été amenés à définir de nouvelles sorties macroscopiques sur le modèle, afin de le calibrer au mieux. Les paramètres réglant les



a) Sans réduction de bruit numérique.



b) Avec réduction de bruit numérique.

FIGURE 1.16 – *Apparition de bruit numérique en temps long : des mailles parasites apparaissent dans le maillage dans des zones où il n’y a pas de densité significative (panel a) ). Avec notre méthode de réduction de bruit, ces mailles parasites disparaissent (panel b) ).*

vitesses de vieillissement et de maturation  $g_f$  et  $h_f$  étant précédemment établis, nous avons défini la notion de perte de masse instantanée, et celle de perte cumulée, en s’inspirant de la notion de taux de risque (hazard rate, on pourra consulter [67] pour plus de détails), pour mieux calibrer le terme de perte par apoptose (voir Figure 1.19). D’autre part, nous avons défini la notion d’index mitotique numérique, en s’inspirant de la notion biologique d’index mitotique (voir Figure 1.19). Il est important de noter que cette sortie supplémentaire peut être définie dans notre modèle où la mitose est bien représentée à l’interface entre les zones SM et G1 de deux cycles cellulaires consécutifs, contrairement aux modélisations plus courantes dans la littérature où la mitose est prise en compte par un terme source distribué sur toute la durée du cycle.

Après avoir modélisé des follicules ovulatoires, nous avons cherché à simuler des follicules atrétiques, c’est-à-dire des follicules dont l’effectif de cellules de granulosa commence par augmenter puis décroît en raison de conditions hormonales défavorables. Ce faisant, nous avons pris conscience de l’importance de choisir les valeurs des paramètres dans une gamme où le comportement en temps long dépendait effectivement du contrôle, c’est à dire ici de l’évolution de l’ensemble des follicules de la cohorte. En effet, la plupart des follicules atrétiques obtenus jusqu’ici étaient atrétiques quel que soit le contrôle. Nous avons mis au point une stratégie consistant à vérifier par une simulation numérique préliminaire que chaque jeu de paramètres pouvait conduire à un comportement ovulatoire, dans un environnement favorable,

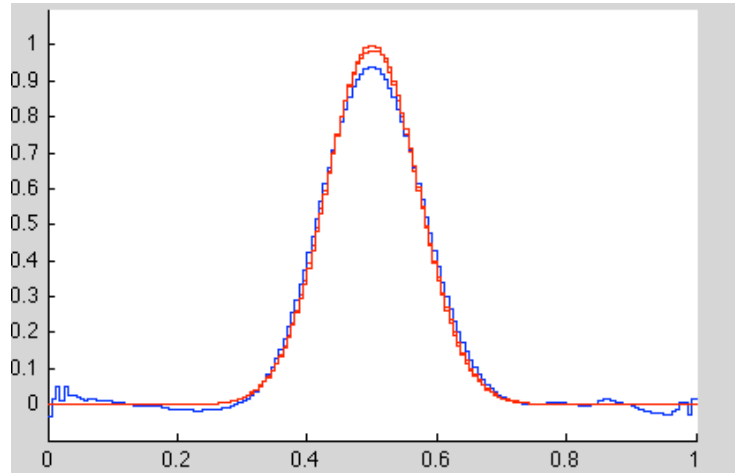


FIGURE 1.17 – *Origine du bruit numérique. Approximation numérique d’une équation de transport  $\partial_t \phi + \partial_x \phi = 0$  avec adaptation de maillage par multirésolution. En abscisse, l’espace  $x$ , en ordonnée, la densité  $\phi$ , après une période. Les deux courbes rouges représentent la solution exacte et la solution volumes finis avec un maillage uniforme. La courbe bleue représente la solution multirésolution reconstruite au temps final sur le maillage le plus fin (pour ce maillage, on a pris  $\epsilon = 0.05$ ,  $N_m = 10$ , et  $L = 4$ ). On a fixé le paramètre  $\epsilon$  contrôlant la précision de l’algorithme multirésolution à une valeur exagérément grande de manière à ce que les erreurs dues au déraffinement puis à la reconstruction soient visibles à l’œil nu.*

correspondant à un contrôle global maximal constant. On verra dans le chapitre 5 que si les paramètres d’un follicule correspondent à une concentration en temps long de la densité de cellules dans la zone de l’espace âge  $\times$  maturité où le terme source modélisant l’apoptose est actif, il est en quelque sorte "prédestiné" à disparaître. Nous avons alors proposé un jeu de paramètres de référence à 20 follicules, un ovulatoire et 19 atrétiques (voir Figure 1.19). D’autres modifications par rapport aux simulations numériques antérieures ont concerné la condition initiale du modèle. Nous avons adopté une condition initiale correspondant à une population initiale désynchronisée (voir Figure 1.18), définie par morceaux en maturité (composée de trois morceaux), exponentielle décroissante en âge, décalée de l’interface entre phase de prolifération et phase différenciée.

### 1.3.5 Schéma numérique multiéchelles et analyse de performances par propagation d’incertitudes

Afin d’évaluer les performances de notre code de calcul, nous avons procédé à un test classique en calcul parallèle : le test de scalabilité faible.

Cela consiste à tracer le temps de calcul en fonction du nombre de processeurs. Il est important de noter que la taille du problème considéré augmente avec le nombre



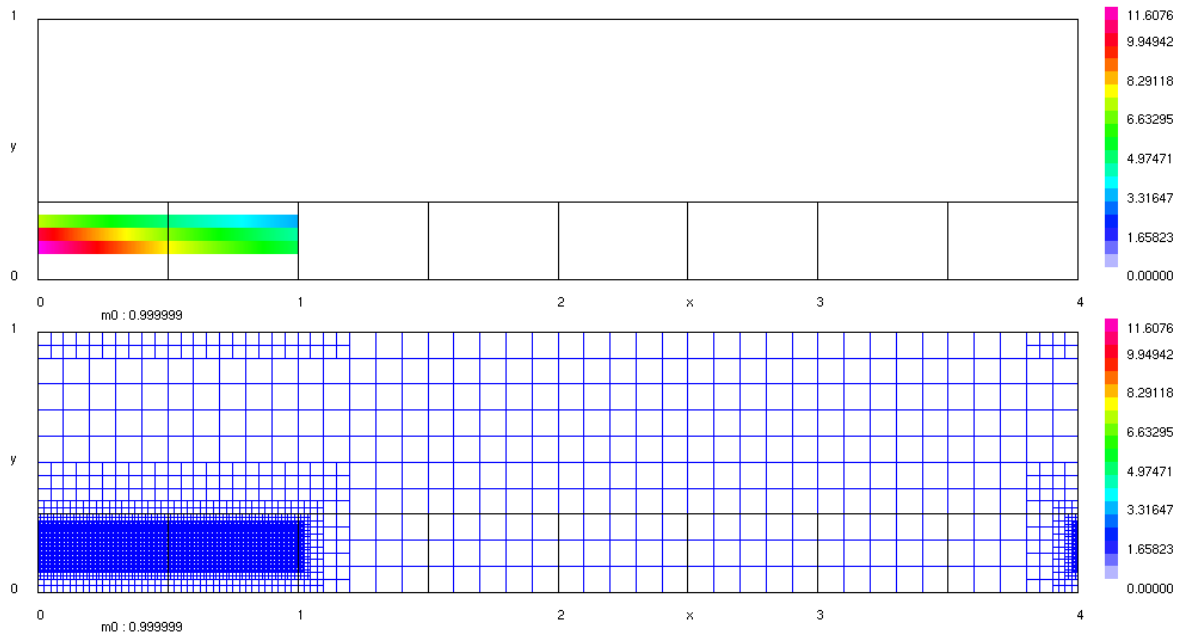


FIGURE 1.18 – Condition initiale à décroissance exponentielle en âge.

de processeurs, contrairement à un test de scalabilité forte, où il reste constant (test classique en décomposition de domaine). La courbe correspondant à la scalabilité faible optimale est la droite horizontale correspondant au temps de calcul d'un follicule sur un processeur. Les communications, ainsi que les phénomènes d'attente entre les différents processeurs, ajoutent du temps de calcul, et éloignent la courbe de performance de l'optimal.

Par ailleurs, les paramètres du modèle influent eux aussi sur les performances en temps de calcul et en occupation mémoire, dans le cas d'un schéma volumes finis adaptatif explicite en temps, dont les paramètres de discrétisation doivent satisfaire une contrainte de stabilité.

On peut obtenir d'excellents résultats avec un test de scalabilité faible, effectué sur un jeu de paramètres de référence. Cependant, si une perturbation sur les paramètres en entrée (erreur de mesure, valeur modifiée par un utilisateur, ...) entraîne une diminution du pas de temps ou bien un raffinement du maillage, les performances réelles seront loin de celles de la situation de référence.

Afin d'avoir des résultats réalistes, du point de vue de l'utilisateur, nous avons proposé une méthode plus générale, inspirée des méthodes de propagation d'incertitudes. Le principe est le suivant : on définit un jeu de paramètres de référence, et des lois de probabilités perturbant ces paramètres. Par une méthode de type Monte Carlo, on effectue un grand nombre de réalisations (simulations numériques basées sur ces paramètres), et on évalue les valeurs moyennes de certaines sorties, comme le taux d'occupation moyen du maillage adaptatif et le temps utilisateur moyen. Cela définit un test de scalabilité faible en moyenne.

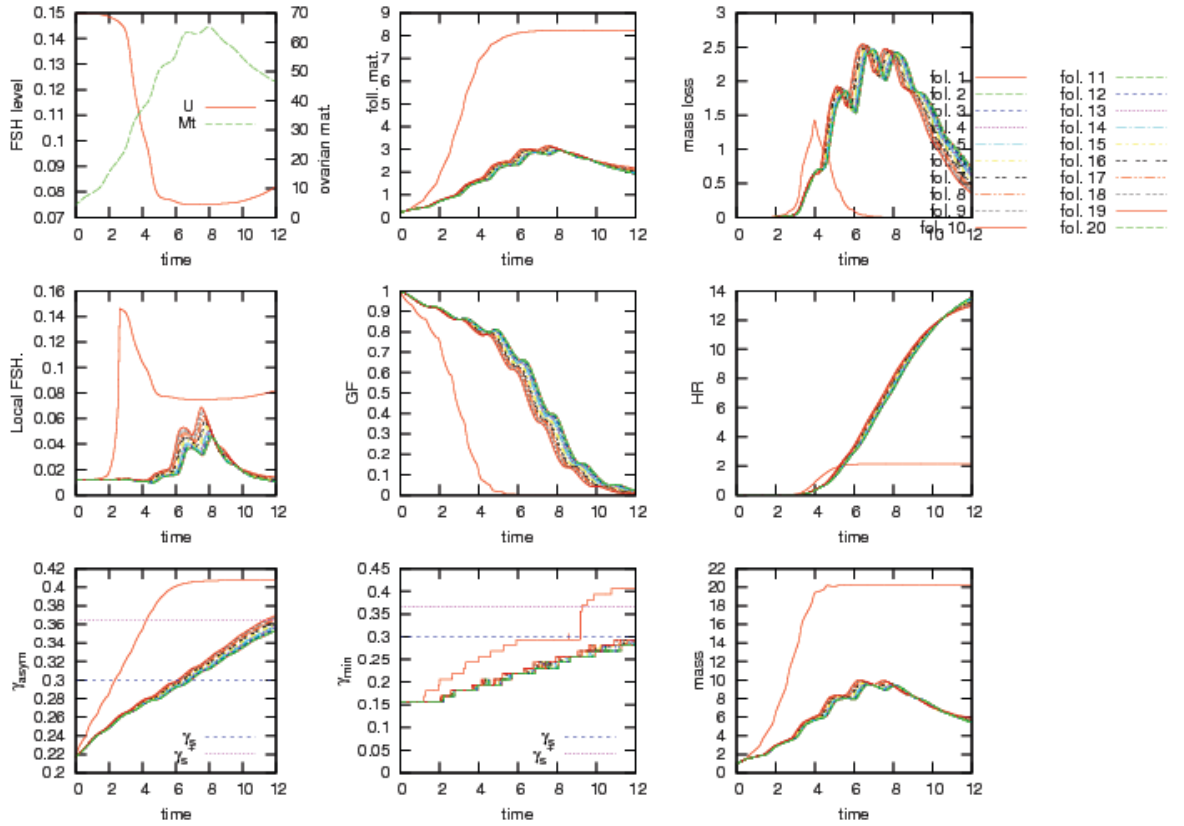


FIGURE 1.19 – *Situation de référence à 20 follicules : l'un remplit tout les critères d'un follicule ovulatoire, tandis que les 19 autres remplissent les critères de follicules atrétiques. De haut en bas, de gauche à droite : contrôle global  $U$  et maturité globale  $M_1$ , maturités folliculaires  $m_1^f$ , pertes instantanées, contrôles mésoscopiques  $u_f$ , fractions de croissance, pertes cumulées, fonctions  $\zeta$  des follicules, fonctions  $\gamma_{\min}$  des follicules, masses des follicules  $m_0^f$ . Pour plus de détails, on pourra consulter le chapitre 5 dévolu à la calibration du modèle.*

Ce test permet à l'utilisateur de connaître, de façon plus réaliste et plus sûre qu'un simple test de scalabilité, les performances qu'il peut attendre d'un code de calcul. Sur la Figure 1.20, on présente les résultats d'un test de scalabilité faible moyenne, appliqué à notre code de calcul : CodeFollicleMR. Sur le panel de gauche, on représente le taux d'occupation moyen (à valeurs dans  $[0, 1]$ ) du maillage adaptatif, par rapport au maillage uniforme avec la discrétisation la plus fine présente dans le maillage adaptatif. On note que ce dernier se situe dans l'intervalle  $[0.195, 0.225]$ . Sur le panel de droite, on représente le temps de calcul moyen en fonction du nombre de processeurs. Cette courbe est proche de la limite théorique, ce qui signifie que le test de scalabilité faible moyenne donne de bons résultats dans ce cas.

On étudie plus en détails dans le chapitre 6 les relations entre la variabilité des paramètres et celles des performances numériques, occupation mémoire et temps de

calcul.

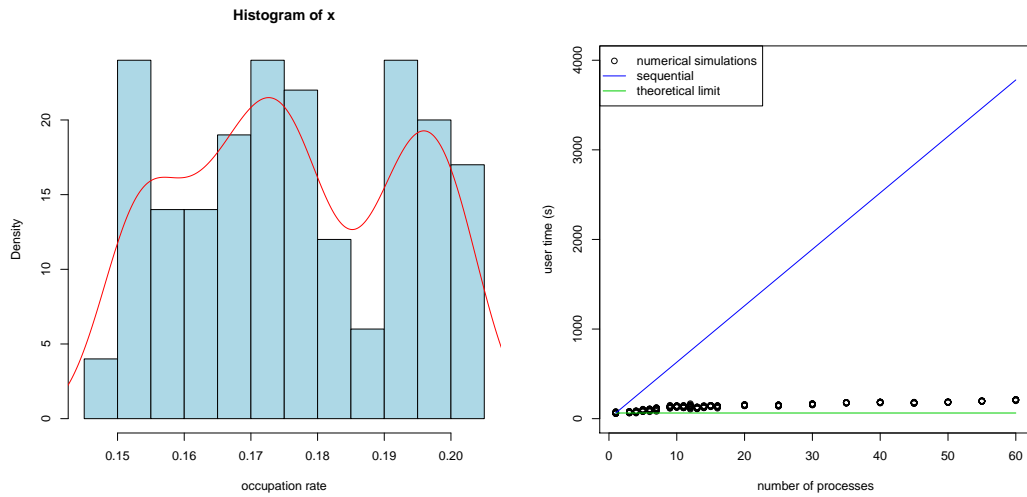


FIGURE 1.20 – *Test de scalabilité faible moyenne (résultats visualisés avec le logiciel R). On effectue  $N = 20$  réalisations à partir de paramètres perturbés, et on récupère le taux moyen d'occupation du maillage adaptatif pour un follicule (à gauche), et le temps de calcul moyen en fonction du nombre de processeurs (à droite). La courbe verte (th.lim.) représente la limite théorique d'optimalité, la courbe bleue (seq.) représente le temps de calcul séquentiel théorique, et les points noirs sont des résultats des simulation. Le taux d'occupation mémoire, par rapport à un maillage uniforme sur la grille la plus fine présente dans le maillage adaptatif, est de  $21\% \pm 1\%$ . L'équilibrage des charges reste raisonnable et le test de scalabilité faible moyenne donne de bons résultats.*

# Numerical simulation of the selection process of the ovarian follicles

Ce chapitre a fait l'objet d'une publication dans ESAIM Proceedings [6] lors du CEMRACS 2011.

## Sommaire

<b>2.1</b>	<b>Biological and biomathematical background</b>	<b>27</b>
2.1.1	Biological background	27
2.1.2	Biomathematical model	29
<b>2.2</b>	<b>Numerical Method</b>	<b>32</b>
2.2.1	Discretization	32
2.2.2	Macroscopic scale : piecewise constant approximation of the hormonal control	33
2.2.3	Mesoscopic scale : finite volume scheme	33
2.2.4	Macro/Meso scale Parallelization	35
2.2.5	Parallel algorithm and improvement to order 2 in time	36
<b>2.3</b>	<b>Numerical simulations</b>	<b>36</b>
2.3.1	Initial condition	36
2.3.2	Sets of default parameters	37
2.3.3	Time evolution of the cell density for one follicle	38
2.3.4	Competition between ten follicles	39
2.3.5	Convergence rate test	40
2.3.6	HPC test	41

**Abstract** This paper presents the design and implementation of a numerical method to simulate a multiscale model describing the selection process in ovarian follicles. The PDE model consists in a quasi-linear hyperbolic system of large size, namely  $N_f \times N_f$ , ruling the time evolution of the cell density functions of  $N_f$  follicles (in practice  $N_f$  is of the order of a few to twenty). These equations are weakly coupled through the sum of the first order moments of the density functions. The time-dependent equations make use of two structuring variables, age and maturity, which play the roles of space variables. The problem is naturally set over a compact domain of  $\mathbf{R}^2$ . The formulation of the time-dependent controlled transport

## Chapitre 2. Numerical simulation of the selection process of the ovarian follicles

---

coefficients accounts for available biological knowledge on follicular cell kinetics. We introduce a dedicated numerical scheme that is amenable to parallelization, by taking advantage of the weak coupling. Numerical illustrations assess the relevance of the proposed method both in term of accuracy and HPC achievements.

**Résumé** Ce document présente la conception et l’implémentation d’une méthode numérique servant à simuler un modèle multiéchelle décrivant le processus de sélection des follicules ovariens. Le modèle EDP consiste en un système hyperbolique quasi linéaire de grande taille, typiquement  $N_f \times N_f$ , gouvernant l’évolution des fonctions de densité cellulaire pour  $N_f$  follicules (en pratique  $N_f$  est de l’ordre de quelque uns à une vingtaine). Ces équations d’évolution utilisent deux variables structurantes, l’âge et la maturité, qui jouent le rôle de variables d’espace. Le problème est naturellement posé sur un domaine compact de  $\mathbf{R}^2$ . La formulation du transport à coefficients variables au cours du temps en fonction du contrôle est issue des connaissances disponibles sur la cinétique cellulaire au sein des follicules ovariens. Nous présentons un schéma numérique dédié au problème parallélisable en tirant parti du faible couplage. Des simulations numériques permettent d’évaluer la pertinence de la méthode proposée tant en terme de précision que de calcul haute performance.

## Introduction

This work is motivated by a mathematical modelling approach of a complex physiological system, the development of ovarian follicles. The model describes both the cell dynamics within each follicle and the competition process within the population of follicles. The resulting model ([41]) is a large scale system of weakly coupled quasi-linear transport equations, where integro-differential terms occur both in the velocity and source term. The coupling terms account for the endocrine-based dependence of one follicle dynamics on all other developing follicles. The well-posedness of the Cauchy problem is established in [71]. Existence and uniqueness of weak solutions is proved for bounded initial conditions. The competition process was investigated in a game theory approach after reducing the PDE model to coupled ODE systems [60]. Control problems associated with this model are investigated in [40] (computation of backwards reachability sets) and [24] (optimal control in minimal time).

We are specifically interested in the numerical issues raised by this multiscale model. In previous works [23, 39], a CTU numerical scheme has been implemented in the Bearclaw<sup>1</sup> environment, which is based on adaptive mesh refinement using wave-propagation algorithms [53, 54]. Here, we develop a dedicated code, which allows us (i) to handle the conservative form of the equations, (ii) to deal with the discontinuous coefficients and (iii) to use high performance computing (HPC) techniques in order to speed up the computing and be able to simulate as many as twenty follicles.

---

1. <http://www.pas.rochester.edu/~bearclaw/>

## Chapitre 2. Numerical simulation of the selection process of the ovarian follicles

---

The paper is organized as follows. In section 2.1, we introduce the biological background and the multiscale model. In section 2.2, we describe the numerical scheme in detail. In section 2.3, we illustrate the simulation outputs and assess the algorithm robustness, accuracy and scalability on parallel architectures.

### 2.1 Biological and biomathematical background

#### 2.1.1 Biological background

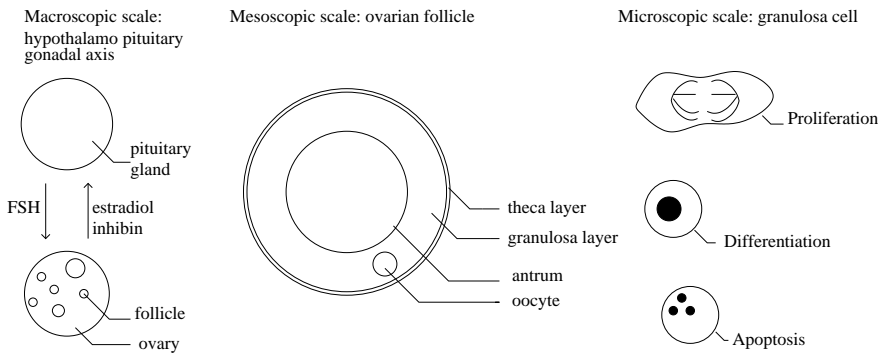


FIGURE 2.1 – *Follicle development as a multiscale process. Left : endocrine feedback loop between the ovaries and the hypothalamo-pituitary-axis. Middle : schematic 2D view of an ovarian follicle. Right : different cell states encountered by the granulosa cells during follicular development.*

##### 2.1.1.1 Mesoscopic scale : ovarian follicles

Ovarian follicles are spheroidal tissular structures sheltering the oocytes (figure 2.1). Each follicle goes through a development process, composed of two main parts : the basal development and the terminal development (figure 2.2). During basal development, the follicles are independent of hormonal support from the pituitary gland, while, during the terminal part of their development, they become dependent on FSH (follicle stimulating hormone) supply. Follicle development can end up either by ovulation, in the best case, or much more often by degeneration through atresia. The commitment of a follicle to either ovulation or atresia is driven by the changes occurring in the follicular cell population.

##### 2.1.1.2 Macroscopic scale : competition process and endocrine feedback loop

In some sense, the terminal part of follicle development can be considered as a competition between follicles for FSH resource. FSH controls follicle development, while FSH levels are in turn controlled by two hormones secreted by the ovary, inhibin

## Chapitre 2. Numerical simulation of the selection process of the ovarian follicles

---

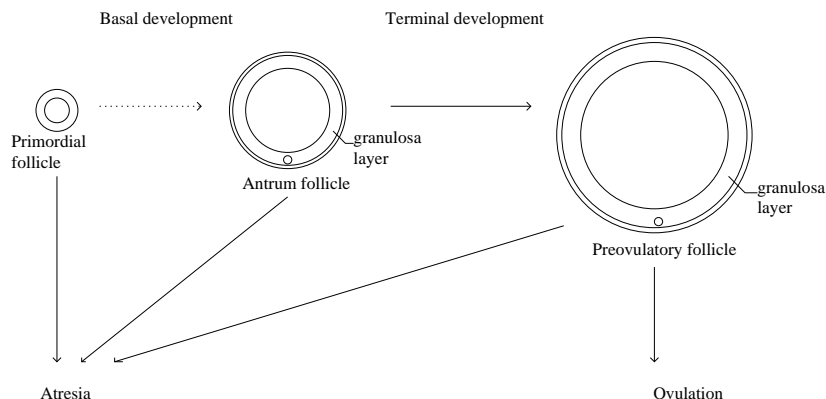


FIGURE 2.2 – *Follicular development : from basal to terminal development. After exiting the pool of quiescent primordial follicles, ovarian follicles enter a several-month long process of growth and maturation, that either ends up by ovulation or degeneration through atresia. In the basal part of development, follicle growth is mainly due to the oocyte enlargement and increase in the number of granulosa cell layers. In the final part of development, granulosa cells progressively stop proliferating and follicle growth is mainly due to the enlargement of the antrum, a liquid-filled cavity in the center of the follicle.*

and estradiol, whose plasma levels are determined by the summed contributions of all maturing follicles (figure 2.1). This ovarian hormonal feedback induces a drop in FSH levels that will penalize the follicles, except those (in the poly-ovulating situations) or that (in the mono-ovulating ones) that are sufficiently mature to survive in a FSH poor environment. The rising levels of estradiol finally triggers the ovulatory surge that leads to the ovulation of the surviving follicles.

### 2.1.1.3 Microscopic scale : granulosa cell kinetics

Each granulosa cell can be encountered in either of three different cell states : proliferation, differentiation or apoptosis (programmed cell death) (figure 2.3). At the beginning of terminal development, most granulosa cells are progressing along the cell division cycle, that can be split into a  $G1$  phase, where cells are sensitive to FSH control, and a  $SM$  phase, where cells are preparing for mitosis and are insensitive to FSH control. At the end of mitosis, one single mother cell gives birth to two daughter cells. During the FSH sensitive phase, cells become more and more mature, up to a point where they reach a threshold maturity and exit the cell division cycle. At this time they are exposed to a great risk of apoptosis, and may die if the FSH environment is not favorable. After exiting the cell cycle, the cells stop proliferating definitively, but their maturity still increases, so that they contribute more and more to hormone (and especially estradiol) secretion.

## 2.1.2 Biomathematical model

### 2.1.2.1 Computing domain

Let us introduce the variables

$$\begin{cases} a & \text{age,} \\ \gamma & \text{maturity,} \\ t & \text{time,} \end{cases}$$

the vector of granulosa cell densities for the  $N_f$  follicles

$$\Phi(a, \gamma, t) = (\phi_1(a, \gamma, t), \dots, \phi_{N_f}(a, \gamma, t)),$$

and the computing domain  $\Omega$  in the  $(a, \gamma)$  plane,

$$\Omega = \{(a, \gamma), 0 \leq a \leq N_c \times D_a, 0 \leq \gamma \leq 1\}$$

where  $N_c$  is the number of cell cycles and  $D_a$  is the duration of one cycle. The different cell states described in paragraph 2.1.1.3 take place in three subdomains as illustrated in Figure 2.3. The differentiation phase  $D$  corresponds to the area of the domain where the maturity overcomes the cellular maturity threshold  $\gamma_s$

$$\begin{cases} G1 = \{(a, \gamma) \in \Omega, & pD_a \leq a \leq (p + 1/2)D_a, & p = 0, \dots, N_c - 1, & 0 \leq \gamma \leq \gamma_s\}, \\ SM = \{(a, \gamma) \in \Omega, & (p + \frac{1}{2})D_a \leq a \leq (p + 1)D_a, \\ & p = 0, \dots, N_c - 1, & 0 \leq \gamma \leq \gamma_s\}, \\ D = \{(a, \gamma) \in \Omega, & \gamma_s \leq \gamma\}. \end{cases}$$

### 2.1.2.2 Hyperbolic system

If we consider, for each follicle, a conservation law for the granulosa cell population density, we obtain the following hyperbolic system, satisfied by  $\Phi$  :

$$\begin{cases} \frac{\partial \phi_1(a, \gamma, t)}{\partial t} + \frac{\partial (g(a, \gamma, u_1(t)) \phi_1(a, \gamma, t))}{\partial a} + \frac{\partial (h(a, \gamma, u_1(t)) \phi_1(a, \gamma, t))}{\partial \gamma} \\ \quad \quad \quad = -\lambda(a, \gamma, U(t)) \phi_1(a, \gamma, t) \\ \quad \quad \quad \vdots \\ \frac{\partial \phi_f(a, \gamma, t)}{\partial t} + \frac{\partial (g(a, \gamma, u_f(t)) \phi_f(a, \gamma, t))}{\partial a} + \frac{\partial (h(a, \gamma, u_f(t)) \phi_f(a, \gamma, t))}{\partial \gamma} \\ \quad \quad \quad = -\lambda(a, \gamma, U(t)) \phi_f(a, \gamma, t) \\ \quad \quad \quad \vdots \\ \frac{\partial \phi_{N_f}(a, \gamma, t)}{\partial t} + \frac{\partial (g(a, \gamma, u_{N_f}(t)) \phi_{N_f}(a, \gamma, t))}{\partial a} + \frac{\partial (h(a, \gamma, u_{N_f}(t)) \phi_{N_f}(a, \gamma, t))}{\partial \gamma} \\ \quad \quad \quad = -\lambda(a, \gamma, U(t)) \phi_{N_f}(a, \gamma, t) \end{cases} \quad (2.1)$$



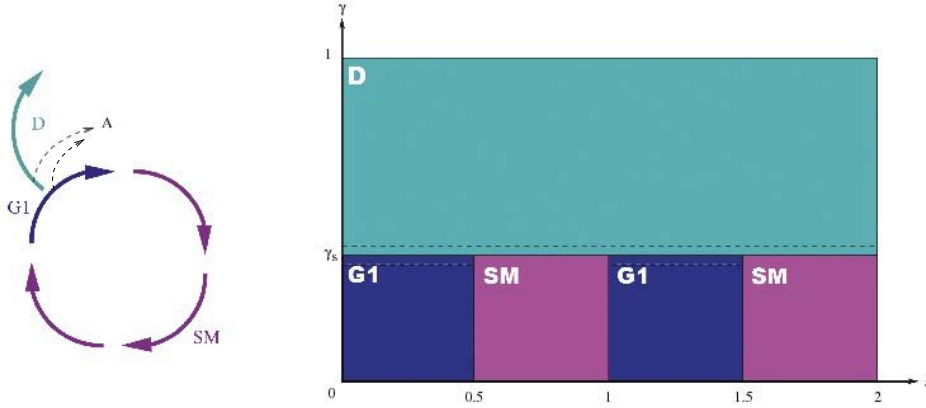


FIGURE 2.3 – Division of the spatial domain according to the cell phases for unit cycle duration  $D_a = 1$ . Left : schematic view of a granulosa cell cycle. Right :  $(a, \gamma)$  plane. The bottom part represents the successive cell cycles, each composed of the G1 and SM phases. The top part corresponds to the differentiation phase.

The initial condition is given by the distribution of the granulosa cell populations at the initial time

$$\Phi(a, \gamma, 0) = \Phi_0(a, \gamma)$$

where  $\Phi_0$  is compactly supported in  $]0, N_c D_a[ \times ]0, 1[$ . Since the functional domain  $\Omega$  is compact, we need to express boundary conditions. Considering some qualitative properties of the maturation function  $h$ , defined below, the horizontal top boundary  $\gamma = 1$  is never reached. Similarly, we can fix the number of cell cycles  $N_c$  in accordance with the ageing function  $g$ , so that the right vertical boundary  $a = N_c D_a$  is never reached. For sake of computing simplicity, we therefore assume spatial periodicity on the outer boundaries of  $\Omega$  in the numerical simulations.

The ageing function  $g$  appearing in (4.38) is defined by

$$g(a, \gamma, u) = \begin{cases} g_1 u + g_2 & \text{for } (a, \gamma) \in G1 \\ 1 & \text{for } (a, \gamma) \in SM \cup D \end{cases} \quad (2.2)$$

where  $g_1, g_2$  are real positive constants. The maturation function  $h$  is defined by

$$h(a, \gamma, u) = \begin{cases} \tau_h (-\gamma^2 + (c_1 \gamma + c_2)(1 - \exp(\frac{-u}{\bar{u}}))) & \text{for } (a, \gamma) \in G1 \cup D \\ 0 & \text{for } (a, \gamma) \in SM \end{cases} \quad (2.3)$$

where  $\tau_h, c_1, c_2$  and  $\bar{u}$  are real positive constants. The source term, that represents cell loss through apoptosis, is defined by

$$\lambda(a, \gamma, U) = \begin{cases} K \exp(-(\frac{\gamma - \gamma_s}{\bar{\gamma}})^2) \times (1 - U) & \text{for } (a, \gamma) \in G1 \cup D \\ 0 & \text{for } (a, \gamma) \in SM \end{cases} \quad (2.4)$$

## Chapitre 2. Numerical simulation of the selection process of the ovarian follicles

---

where  $K$ ,  $\gamma_s$  and  $\bar{\gamma}$  are real positive constants.

Remark. The precise values of the constants depends on species and breeds; they will be fixed later.

### 2.1.2.3 Closure equations

The equations in the PDE system (4.38) are linked together through the argument  $u_f(t)$  appearing in the speeds  $g(a, \gamma, u_f)$  and  $h(a, \gamma, u_f)$  and the argument  $U(t)$  in the source term  $\lambda(a, \gamma, U)$ .  $U(t)$  and  $u_f(t)$  represent respectively the plasma FSH level and the locally bioavailable FSH level and depend on some maturation moments of the densities.

– follicular cell mass (local, one by follicle)

$$m_0(f, t) = \int_0^1 \int_0^{N_c D_a} \phi_f(a, \gamma, t) da d\gamma, \quad 1 \leq f \leq N_f \quad (2.5)$$

This moment is an interesting quantity since it corresponds to an observable variable (the total cell number in one follicle) even if it does not come as such into play in the equations.

– follicular maturity (local, one by follicle)

$$m(f, t) = \int_0^1 \int_0^{N_c D_a} \gamma \phi_f(a, \gamma, t) da d\gamma, \quad 1 \leq f \leq N_f \quad (2.6)$$

– ovarian maturity (global, shared by all follicles)

$$M(t) = \sum_{f=1}^{N_f} m(f, t), \quad (2.7)$$

– plasma FSH level (global, shared by all follicles)

$$U(t) = U_{\min} + \frac{1 - U_{\min}}{1 + \exp(c(M(t) - \bar{M}))}, \quad (2.8)$$

where  $U_{\min}$ ,  $c$  and  $\bar{M}$  are real positive constants

– locally bioavailable FSH level

$$u_f(t) = \min \left( b_1 + \frac{e^{b_2 m(f, t)}}{b_3}, 1 \right) U(t), \quad (2.9)$$

where  $b_1$ ,  $b_2$  and  $b_3$  are real positive constants.

It is important to notice that the ageing function (5.2) is discontinuous (in general) on the interfaces  $G1 - SM$ , and that the maturation function (4.40) is discontinuous (in general) on the interfaces  $SM - D$  (Figure 2.3). It is necessary to introduce transmission conditions in order to overcome possible failure of uniqueness due to the discontinuities in coefficients (see [43] or [12] for instance). The precise definition of the required transmission conditions has been addressed in the paper by Peipei

## Chapitre 2. Numerical simulation of the selection process of the ovarian follicles

---

Shang ([71]). We suppose that, for each cycle  $p = 1, \dots, N_c$  and each follicle  $f = 1, \dots, N_f$ , the flux on the  $a$ -axis is continuous between the phases  $G1$  and  $SM$

$$\phi_f(t, a^+, \gamma) = (g_1 u_f + g_2) \phi_f(t, a^-, \gamma), \quad a = (p - 1/2)D_a, \quad 0 \leq \gamma \leq \gamma_s, \quad (2.10)$$

and that the flux is doubling on the interfaces  $SM - G1$ , which accounts for the birth of new cells at the end of each cell cycle

$$(g_1 u_f + g_2) \phi_f(t, a^+, \gamma) = 2 \phi_f(t, a^-, \gamma), \quad a = pD_a, \quad 0 \leq \gamma \leq \gamma_s. \quad (2.11)$$

Finally, we suppose homogeneous Dirichlet condition to the north of the interface  $SM - D$

$$\phi_f(t, a, \gamma_s^+) = 0, \quad (p - 1/2)D_a \leq a \leq pD_a. \quad (2.12)$$

The time dependent ovarian maturity  $M(t)$  defined in (2.7) is compared to the ovarian maturity threshold, denoted by  $M_o$ , to define the final time, beyond which the competition is over

$$t_{final} = \inf\{t, M(t) \geq M_o\}. \quad (2.13)$$

The follicles are then sorted into two classes : ovulatory if the maturity is higher than the follicle maturity threshold, denoted by  $M_f$ , atretic otherwise.

## 2.2 Numerical Method

### 2.2.1 Discretization

In this paragraph the cycle duration is set to  $D_a = 1$ . We denote by  $\Delta a$  (respectively  $\Delta \gamma$ ) the space step in the age (respectively maturity) direction. In practice we choose  $\Delta a = \Delta \gamma$ . The discretization step  $\Delta \gamma$  and the cellular maturity threshold  $\gamma_s$  must be chosen so that the interfaces where the speed coefficients (5.2,4.40) are discontinuous fall on grid points covering the computational domain of size  $[N_c, 1]$ . Denoting by  $N_m$  the number of grid cells by half granulosa cell cycle, we set  $N_\gamma = 2N_m$  and  $\Delta a = \Delta \gamma = 1/N_\gamma$  and we introduce the dedicated<sup>2</sup> notations for grid points  $(a_k, \gamma_l)$  and mesh centers  $(a_{k+1/2}, \gamma_{l+1/2})$

$$a_k = k\Delta a, \quad a_{k+1/2} = (k + 1/2)\Delta a, \quad \text{for } k = 0, \dots, N_c \times N_\gamma, \quad (2.14)$$

$$\gamma_l = l\Delta \gamma, \quad \gamma_{l+1/2} = (l + 1/2)\Delta \gamma, \quad \text{for } l = 0, \dots, N_\gamma. \quad (2.15)$$

Considering that the time step  $\Delta t^n$  may change at every iteration, in order to preserve stability, the time discretization is defined by

$$t^0 = 0, \quad t^{n+1} = t^n + \Delta t^n, \quad \text{for } n = 0, \dots, N_t \quad (2.16)$$

with  $N_t$  such that  $t^{N_t} = t_{final}$ . The unknowns are the approximate mean values of the density vector in each grid mesh

$$\left\{ \begin{array}{l} \Phi_{k,l}^n \approx \frac{1}{\Delta a \Delta \gamma} \int_{a_k}^{a_{k+1}} \int_{\gamma_l}^{\gamma_{l+1}} \Phi(a, \gamma, t^n) d\gamma da, \quad \text{for } k = 0, \dots, N_c N_\gamma - 1, \\ \text{and } l = 0, \dots, N_\gamma - 1 \end{array} \right.$$

whose components  $\phi_{f,k,l}^n$  are the discrete density values for each follicle.

---

2. We use this index notation for the use of the forthcoming Multiresolution development

### 2.2.2 Macroscopic scale : piecewise constant approximation of the hormonal control

We define the approximation of the control terms (2.5 - 4.44) at each time step  $n = 0, \dots, N_t$

$$m_{0,f}^n = \Delta a \Delta \gamma \sum_{l=0}^{N_\gamma-1} \sum_{k=0}^{N_c N_\gamma-1} \phi_{f,k,l}^n, \quad \text{for } f = 1, \dots, N_f, \quad (2.17)$$

$$m_f^n = \Delta a \Delta \gamma \sum_{l=0}^{N_\gamma-1} \gamma_{l+1/2} \sum_{k=0}^{N_c N_\gamma-1} \phi_{f,k,l}^n, \quad \text{for } f = 1, \dots, N_f, \quad (2.18)$$

$$M^n = \sum_{f=1}^{N_f} m_f^n, \quad (2.19)$$

$$U^n = U_{\min} + \frac{1 - U_{\min}}{1 + \exp(c(M^n - \bar{M}))}, \quad (2.20)$$

$$u_f^n = \min(b_1 + \frac{e^{b_2 m_f^n}}{b_3}, 1) U^n, \quad \text{for } f = 1, \dots, N_f. \quad (2.21)$$

### 2.2.3 Mesoscopic scale : finite volume scheme

We use a splitting strategy in order to compute the solution of the PDE system (4.38), which amounts to a convective equation

$$\begin{cases} \partial_t \phi_f(a, \gamma, t) + \partial_a(g(a, \gamma, u_f(t)) \phi_f(a, \gamma, t)) \\ \quad + \partial_\gamma(h(a, \gamma, u_f(t)) \phi_f(a, \gamma, t)) = 0 \end{cases} \quad (\text{convective part}),$$

combined with a source equation

$$\partial_t \phi_f(a, \gamma, t) = -\lambda(a, \gamma, U(t)) \phi_f(a, \gamma, t) \quad (\text{source part}),$$

for each follicle  $f = 1, \dots, N_f$ .

#### 2.2.3.1 Convective part

The convection part (2.2.3) is treated with a classical finite volume method. The approximate mean values of the solution at  $t = 0$  are initialized using a midpoint formula, accurate at the order 2 in space

$$\Phi_{k,l}^0 = \Phi_0(a_{k+1/2}, \gamma_{l+1/2}).$$

Using the integral form of the conservation law

$$\left\{ \int_{t^n}^{t^{n+1}} \int_{a_k}^{a_{k+1}} \int_{\gamma_l}^{\gamma_{l+1}} (\partial_t \phi_f(a, \gamma, t) + \partial_a(g(a, \gamma, u_f(t)) \phi_f(a, \gamma, t)) \right. \\ \left. + \partial_\gamma(h(a, \gamma, u_f(t)) \phi_f(a, \gamma, t))) d\gamma da dt = 0, \right.$$

## Chapitre 2. Numerical simulation of the selection process of the ovarian follicles

---

we obtain a recursion on the approximate density of each follicle, where we drop the index  $f$  for clarity sake

$$\phi_{k,l}^{n+1} = \phi_{k,l}^n - \frac{\Delta t}{\Delta a} \left( G_{k+1,l+\frac{1}{2}}(\Phi^n) - G_{k,l+\frac{1}{2}}(\Phi^n) \right) - \frac{\Delta t}{\Delta \gamma} \left( H_{k+\frac{1}{2},l+1}(\Phi^n) - H_{k+\frac{1}{2},l}(\Phi^n) \right) \quad (2.22)$$

where  $G_{k,l+\frac{1}{2}}$  (respectively  $H_{k+\frac{1}{2},l}$ ) is the numerical flux across the vertical edge  $[(a_k, \gamma_l), (a_k, \gamma_{l+1})]$  (respectively the horizontal edge  $[(a_k, \gamma_l), (a_{k+1}, \gamma_l)]$ ) defined by

$$\begin{cases} G_{k,l+\frac{1}{2}}(\Phi^n) \approx \frac{1}{\Delta t^n \Delta \gamma} \int_{t^n}^{t^{n+1}} \int_{\gamma_l}^{\gamma_{l+1}} g(a_k, \gamma, u_f(t)) \phi(a_k, \gamma, t) d\gamma dt, \\ H_{k+\frac{1}{2},l}(\Phi^n) \approx \frac{1}{\Delta t^n \Delta a} \int_{t^n}^{t^{n+1}} \int_{a_k}^{a_{k+1}} h(a, \gamma_l, u_f(t)) \phi(a, \gamma_l, t) da dt. \end{cases} \quad (2.23)$$

The dependence of the flux functions on all the follicle densities through the control term  $u_f$  is emphasized by the  $\Phi^n$  argument. It appears in the explicit in time approximations of the speeds (5.2), and (4.40) at the center of the mesh

$$\begin{cases} g_{k,l}^n = g(a_{k+\frac{1}{2}}, \gamma_{l+\frac{1}{2}}, u_f^n), \\ h_{k,l}^n = h(a_{k+\frac{1}{2}}, \gamma_{l+\frac{1}{2}}, u_f^n). \end{cases}$$

Note that the transmission conditions (4.45) and (4.46) on the interfaces where these speed coefficients (5.2) and (4.40) are discontinuous are exactly treated with the method described in the article of Godlewski and Raviart ([43]).

The numerical fluxes (2.23) are designed using a limiter strategy. Indeed, it is well known that first order schemes, like the Godunov scheme, are diffusive, and that second order schemes, like Lax Wendroff scheme, generate oscillations in the neighborhood of discontinuities. In order to get a stable as well as precise scheme, we take a weighting of a low order scheme and a high order scheme, and we define limited numerical fluxes

$$\begin{aligned} G &= G^{Low} + \ell(r_G)(G^{High} - G^{Low}), \\ H &= H^{Low} + \ell(r_H)(H^{High} - H^{Low}), \end{aligned} \quad (2.24)$$

where  $\ell$  is a limiter function, for example van Leer function

$$\ell(r) = \frac{r + |r|}{1 + r}$$

and  $r_G, r_H$  are

$$r_G = \begin{cases} \frac{g_{k-1,l} \phi_{k-1,l} - g_{k-2,l} \phi_{k-2,l}}{g_{k,l} \phi_{k,l} - g_{k-1,l} \phi_{k-1,l}} & \text{if } g_{k-2,l} \geq 0 \text{ and } g_{k-1,l} \geq 0 \text{ and } g_{k,l} \geq 0, \\ \frac{g_{k+1,l} \phi_{k+1,l} - g_{k,l} \phi_{k,l}}{g_{k,l} \phi_{k,l} - g_{k-1,l} \phi_{k-1,l}} & \text{if } g_{k-1,l} \leq 0, \text{ and } g_{k,l} \leq 0 \text{ and } g_{k+1,l} \leq 0, \\ 0 & \text{otherwise,} \end{cases}$$

## Chapitre 2. Numerical simulation of the selection process of the ovarian follicles

---

$$r_H = \begin{cases} \frac{h_{k,l-1}\phi_{k,l-1} - h_{k,l-2}\phi_{k,l-2}}{h_{k,l}\phi_{k,l} - h_{k,l-1}\phi_{k,l-1}} & \text{if } h_{k,l-2} \geq 0 \text{ and } h_{k,l-1} \geq 0 \text{ and } h_{k,l} \geq 0, \\ \frac{h_{k,l+1}\phi_{k,l+1} - h_{k,l}\phi_{k,l}}{h_{k,l}\phi_{k,l} - h_{k,l-1}\phi_{k,l-1}} & \text{if } h_{k,l-1} \leq 0 \text{ and } h_{k,l} \leq 0 \text{ and } h_{k,l+1} \leq 0, \\ 0 & \text{otherwise.} \end{cases}$$

These ratios are good indicators of the regularity of the function in each direction (see [72]). In fact, a steep gradient or a discontinuity gives a ratio far from 1, whereas a smooth function gives a ratio close to 1.

The first order fluxes entering equation (3.13) are the Godunov fluxes

$$\begin{cases} G_{k,l+\frac{1}{2}}^{Low}(\Phi^n) = (g_{k-1,l}^n)^+ \phi_{k-1,l}^n + (g_{k,l}^n)^- \phi_{k,l}^n, \\ H_{k+\frac{1}{2},l}^{Low}(\Phi^n) = (h_{k,l-1}^n)^+ \phi_{k,l-1}^n + (h_{k,l}^n)^- \phi_{k,l}^n, \end{cases}$$

and the high order fluxes are the Lax Wendroff ones

$$\begin{cases} G_{k,l+\frac{1}{2}}^{High}(\Phi^n) = \frac{1}{2}(g_{k-1,l}^n \phi_{k-1,l}^n + g_{k,l}^n \phi_{k,l}^n), \\ H_{k+\frac{1}{2},l}^{High}(\Phi^n) = \frac{1}{2}(h_{k,l-1}^n \phi_{k,l-1}^n + h_{k,l}^n \phi_{k,l}^n). \end{cases}$$

Consequently, since  $\Delta a = \Delta \gamma$ , the CFL condition which guarantees the stability is

$$\Delta t_{f,FVS}^n \leq CFL \frac{\Delta \gamma}{\max_{k,l} (|g_{k,l}|, |h_{k,l}|)} \quad (2.25)$$

with  $CFL \leq \frac{1}{2}$ .

### 2.2.3.2 Source part

The source part (2.2.3) of the PDE system is explicitly dealt with

$$\phi_{k,l}^{n+1} = \phi_{k,l}^n - \Delta t \lambda(a_{k+\frac{1}{2}}, \gamma_{l+\frac{1}{2}}, U^n) \phi_{k,l}^n, \quad (2.26)$$

which implies a stability condition on the time step, which we strengthen to enforce positivity

$$\Delta t_{f,source}^n \leq \frac{1}{\max_{k,l} |\lambda(a_{k+\frac{1}{2}}, \gamma_{l+\frac{1}{2}}, U^n)|}. \quad (2.27)$$

### 2.2.4 Macro/Meso scale Parallelization

In order to speed up the numerical simulations, the code is implemented on a parallel architecture. Every follicle follows its own dynamic independently of the others, except that it shares with them the FSH resource, and that it contributes to the tuning of plasma FSH level. Hence it is natural to use a SIMD (single instruction, multiple data) strategy, with one follicle by process. The only communications that have to be done at each time step concern the ovarian maturity, through a reduction

## Chapitre 2. Numerical simulation of the selection process of the ovarian follicles

---

operation (sum) and the current time step  $\Delta t^n$ , through a reduction operation (min). Each process computes the maximum time step satisfying both conditions (3.17) and (2.27)

$$\Delta t_f^n = \min\{\Delta t_{f,FVS}^n, \Delta t_{f,source}^n\} \quad (2.28)$$

relevant for its own follicle, and the time step used by all processes is

$$\Delta t^n = \min_f\{\Delta t_f^n\}. \quad (2.29)$$

### 2.2.5 Parallel algorithm and improvement to order 2 in time

Finally, we get the following parallel algorithm for each time step :

1. Compute time step
  - Compute local time step  $\Delta t_f^n$  (2.28)
  - **Communication** : Compute the minimum of local time steps  $\Delta t^n$  (2.29)
2. Update solution
  - Flux computation (3.13),
  - Convection computation (3.12),
  - Source computation (2.26),
3. Update control terms
  - Compute local maturity (2.18),
  - **Communication** : compute the sum of local maturities (2.19),
  - Compute FSH plasma level (2.20),
  - Compute locally bioavailable FSH rate (2.21).

Denoting by  $E$  the evolution operator, that consists in steps (2) and (3), the second order in time is achieved by a second order Runge-Kutta method (Heun)

$$\begin{aligned} \phi^* &= E(\phi^n), \\ \phi^{**} &= E(\phi^*), \\ \phi^{n+1} &= \frac{1}{2}(\phi^n + \phi^{**}). \end{aligned}$$

## 2.3 Numerical simulations

The code was developed in C++ using the parallel library MPI. The tests were performed on the Jacques-Louis Lions Laboratory super calculator SGI Altix UV 100. The current configuration is 64-core 2 GHz. The visualizations were made with OpenGL and gnuplot.

### 2.3.1 Initial condition

In the first test case, we used, for sake of comparison with anterior results, a piecewise constant function similar to the initial condition used in [39, 40]. Its support in age

## Chapitre 2. Numerical simulation of the selection process of the ovarian follicles

---

covers the first cycle while its amplitude is a step-decreasing function of the maturity.

$$\phi_0(a, \gamma) = \begin{cases} 8 & \text{if } 0.05 \leq \gamma < 0.1 \\ 7 & \text{if } 0.1 \leq \gamma < 0.15 \\ 5 & \text{if } 0.15 \leq \gamma < 0.2 \\ 0 & \text{otherwise.} \end{cases} \text{ if } 0 \leq a \leq 1, \quad (2.30)$$

$$0 \text{ otherwise.}$$

For the other test cases we used as initial condition a gaussian function centered in  $C = (C_a, C_\gamma)$

$$\phi_0(a, \gamma) = \frac{1}{2\pi\sigma} \exp\left(\frac{1}{2} \left( \frac{(a - C_a)^2}{\sigma^2} + \frac{(\gamma - C_\gamma)^2}{\sigma^2} \right)\right). \quad (2.31)$$

The variance  $\sigma^2$  can be chosen so that  $\phi_0$  be smooth enough to be used for testing the convergence rate, which is checked in the paragraph 3.5.

### 2.3.2 Sets of default parameters

Since one of our goals was to improve the computational method, we used the same set of parameters as in [40] to be able to compare at least qualitatively our results to previous simulation outputs. We distinguished two sets of parameters, one, in Table 2.1, for the *global* parameters, that are identical for all follicles (space discretization, CFL condition, . . . ), another, in Table 2.2, for the *local* parameters (initial condition, velocity parameters, . . . ), which can depend on the follicle.

Parameter	Description	Value
$N_m$	number of grid cells by half granulosa cell cycle	30
$N_c$	number of cycles	8
CFL	CFL condition	0.4
$M_f$	follicular maturity threshold	0.7
$M_o$	ovarian maturity threshold	5.
$U_{min}$	FSH plasma level (eq. (5.1)) minimum level	0.075
$c$	slope parameter	2.0
$\bar{M}$	abscissa of the inflection point	4.5
$K$	Apoptosis source term (eq. (4.41)) intensity factor	6.0
$\bar{\gamma}$	scaling factor	0.02
$t_{max}$	maximum time (to avoid excessively long computations)	7

TABLE 2.1 – Values of the global parameters used in Figures 2.4 to 2.9

The number of follicles,  $N_f$ , and the number of grid cells by half granulosa cell cycle,  $N_m$ , may depend on the simulation.



## Chapitre 2. Numerical simulation of the selection process of the ovarian follicles

---

Parameter	Description	Value	
$\gamma_s$	cellular maturity threshold	0.3	
$b_1$	intrafollicular FSH level (eq. (4.44)) basal level	0.08	
$b_2$	exponential rate	2.25	
$b_3$	scaling factor	1450.	
$g_1$	ageing function (eq. (5.2)) rate	4.0	
$g_2$	origin	1.2	
$\tau_h$	Maturation function (eq. (4.40))	1	
$c_1$		0.68	
$c_2$		0.08	
$\bar{u}$		0.02	
$C_a$	Initial condition (eq. (2.31)) abscissa of the center	0.25	
$C_\gamma$		ordinate of the center	0.15
$\sigma^2$		variance	0.0025

TABLE 2.2 – Values of the local (follicle-dependent) parameters used in Figures 2.4 to 2.9

### 2.3.3 Time evolution of the cell density for one follicle

We have first studied the behavior of the cell density of one single follicle as a function of time, in order to check some qualitative properties.

The full movie can be downloaded from

<https://www.ljll.math.upmc.fr/aymard/movie/ovulatory.mpg>

and four snapshots are displayed on Figure 2.4. Note that on the snapshots the color code is time dependent while in the movie it is set once and for all at initial time.

The parameters values are gathered in Table 1 and 2, except for  $g_2 = 1.2$  and  $\tau_h = 1.2$ . Also, in this mono-ovulatory situation we set the follicular and ovarian maturity thresholds equal to  $M_o = M_f = 16.5$ . The spatial discretization for this test uses  $N_m = 30$  cells per half granulosa cell cycle, and the domain  $\Omega$  contains eight cycles. The grid size is therefore  $8 \times (2 \times 30)^2 = 28800$ .

On the first snapshot 2.4 a), we can observe the piecewise constant initial condition (2.30). The cell mass is equal to 1.

On the second snapshot 2.4 b), at time 0.91, we can see that the density has moved to the second cycle. The cell mass is equal to 2.17. At the interface  $a = 2.5$  between zone  $G1$  and  $SM$  the ageing function decreases, which results in a local increase in the density, whose profile becomes narrower. The cell density is starting to double at the end of the second cell cycle, on the  $SM-G1$  interface at  $a = 3$ .

On the third snapshot 2.4 c), at time 1.57, the cell density has reached the third

## Chapitre 2. Numerical simulation of the selection process of the ovarian follicles

---

cycle. It is splitting into a fully differentiated subpopulation and a still proliferating one. It is worth noticing that, consistently with the model, there is no crossing of the  $SM-D$  interface. The cell mass is equal to 3.7.

On the last snapshot 2.4 d), around time  $t = 5$ , towards the end of the simulation, the density is concentrated in the  $D$  phase above the seventh cell cycle. Even if all cells have exited the cell cycle, we can distinguish three different density clouds, each of which being issued from one of the previous cycles. The cell mass is equal to 14.7.

In Figure 2.5 the time evolution of the cell mass (panel A) and follicular maturity (panel B) are displayed as green curves. While the cell mass reaches a constant value as soon as all cells have entered the differentiating phase, the maturity goes on increasing. The blue curves correspond to a similar simulation with a slower maturation function, where  $\tau_h = 1$  instead of 1.2. The cells spend more time in the proliferating phase and they enter the differentiating phase at a later time. Since more mitosis have happened, the cell mass reaches a higher value, around 25 instead of 14.7.

In the first case, with the faster maturity function, the final time condition defined by (2.13) is not reached and the simulation is stopped at  $t_{max} = 7$  after 1967 time steps. This is more than the time required to cover the eight cell cycles. Yet at that time, the follicular maturity, displayed on panel B of Figure 2.5, is around 11. far below the follicular maturity threshold of  $M_f = 16.5$ . In the second case the follicular maturity overcomes the follicular maturity threshold at time  $t = 6.85$  and is around 17. at the end of the simulation.

This behavior of the numerical solution is in accordance with the previous simulations presented in [40] and [39].

### 2.3.4 Competition between ten follicles

The second test consists in a simulation of a competition process between ten follicles. The parameters defining the plasma FSH level are set to  $U_{min} = 0.9$  and  $c = 10$  and the intensity of the source term is set to  $K = 1$ . The follicles are distinguished by the parameter defining their ageing rate (5.2) at the origin. We consider a range of  $g_2$  values running from 0.5 to 0.95, with a 0.05 increment from one follicle to the other

$$g_2 = 0.5, 0.55, \dots, 0.95.$$

The maximum time is set to  $t_{max} = 1.5$ . For all other parameters we use the values in Tables 2.1 and 2.2. Figure 2.6 represents a) the plasma FSH level  $U(t)$  defined by (5.1) b) the locally bioavailable FSH level  $u_f(t)$  defined by (4.44) c) the cell mass  $m_0(f, t)$  of each follicle defined by (2.5), d) the ovarian maturity  $M(t)$  defined by (2.7) and e) the follicular maturity  $m(f, t)$  defined by (4.42). Figure 2.6.d) shows that the ovarian maturity reaches the threshold defined by (2.13) with  $M_o = 5$  around  $t = 1.2$ .

Figure 2.6.e) shows that, at the final time, only three follicles have reached the ovulatory stage, where the follicle maturity defined by (4.42) is higher than  $M_f =$

## Chapitre 2. Numerical simulation of the selection process of the ovarian follicles

---

0.7. They correspond to the values of the parameter  $g_2 = 0.5, 0.55$  and  $0.6$ .

This setup will be used in future work to mimic realistic configurations where ten to twenty follicles can interact together. However the calibration of the model is still in its preliminary stage and the parameters used in this simulation do not meet exactly the biological specifications such as the number of cell cycles performed before ovulation or the final cell number in ovulatory follicles.

### 2.3.5 Convergence rate test

We now turn to the validation of the code, which consists in numerically verifying the asymptotic order of convergence when the time step  $\Delta t$  and space discretization step  $\Delta \gamma$  go to zero. We use the parameters in Tables 2.1 and 2.2, except for the gaussian function, whose variance is set to  $\sigma^2 = 0.002$ , so that it is very smooth. The simulation is stopped at  $t_{max} = 0.05$ , which allows us to reduce the number of cell cycles to  $N_c = 1$  and to discretize the solution on a square grid  $N_\gamma \times N_\gamma$ . We compute the solution for six different levels discretizations

$$N_\gamma = 80, 160, 320, 640, 1280, 2560$$

with the time discretization provided by the stability condition (2.28).

#### 2.3.5.1 Convergence of the numerical scheme for the linear transport

We first study the convergence of the numerical scheme when the initial condition is centred in the *SM* phase  $C = (0.7, 0.15)$  and the final time is small enough for the density to remain in this phase. The maturity function (4.40) and the source term (4.41) are both null, and the ageing function (5.2) is constant and equal to one. For such a constant linear transport, we can compute the exact solution at the final time  $t^{N_t} = t_{max}$  using the characteristic method as well as the error in  $L_1$ -norm of the numerical solution

$$E(\Delta \gamma) = \Delta \gamma^2 \sum_{k=0}^{N_\gamma-1} \sum_{l=0}^{N_\gamma-1} \left| \phi_{k,l}^{N_t} - \phi_0(a_{k+\frac{1}{2}} - t^{N_t}, \gamma_{l+\frac{1}{2}}) \right|.$$

The error curve in log scale is superposed with the theoretical order  $O(\Delta \gamma^2)$  in Figure 2.7.

#### 2.3.5.2 Full model convergence test

We now set the initial condition near the center  $C = (0.2, 0.15)$  of the *G1* phase, still with a maximum time  $t_{max} = 0.05$  ensuring that the cell density remains in this zone. In that case the source term (4.41) and the age and maturity speeds are no more constant, and we no longer know the exact solution of the PDE. The error is computed using the solution with  $N_\gamma = 5120$  as reference solution, that is compared to the solution on the other meshes. Since we have used a dyadic refinement, the size  $N_{ref}$  of the reference mesh in one direction is always a power of two times the

size  $N_{\Delta\gamma}$  of the current mesh. Denoting by  $P = \frac{N_{ref}}{N_{\Delta\gamma}}$  the ratio between the current discretization and the reference finest one, we estimate the discretization error by

$$E(\Delta\gamma) = \Delta\gamma^2 \sum_{k=0}^{N_{\Delta\gamma}-1} \sum_{l=0}^{N_{\Delta\gamma}-1} \left| \phi_{k,l}^{N_t} - \frac{1}{P^2} \sum_{p=0}^P \sum_{m=0}^P \phi_{kP+p,lP+m}^{ref} \right|$$

The asymptotic order  $O(\Delta\gamma^{1.95})$  which best fits the behavior of  $E(\Delta\gamma)$  is displayed in Figure 2.8.

### 2.3.6 HPC test

The improvement in terms of computing time provided by the parallelization is tested through a set of simulations involving an increasing number of identical follicles. The grid for these simulations consists in one cycle of  $200 \times 200$  cells, and the simulation goes on for 900 time steps. In this experiment, we disposed of enough processors to do the computing with one follicle by processor and since the `time` command is used to monitor the computing time, the output of the program has been commented out. Figure 2.9 a) shows the real computing time, which is basically the time elapsed from the beginning of the computation. Figure 2.9 b) shows the user computing time, which cumulates all the processors computing time. The computing time cannot be shorter than that required for a computation involving only one, this constitutes the so-called theoretical limit. The fact that the real computing time remains close to the computing time for one follicle and that the user computing time grows linearly with the number of follicles is very encouraging. The communications between processors appear not to affect significantly the gain in computing time due to the parallel computing. A small increase in real computing time can be noticed beyond eight follicles. This is due to the super calculator architecture, where the processors are pooled in eight processor nodes. The communications within one node are faster than across different nodes. This leads to a threshold effect observed as soon as more than eight processors are needed, therefore involving more than one single node.

## Conclusion

This paper summarizes preliminary works in the development of a dedicated software to illustrate numerically the development of follicles. The uniform grid numerical method have been successfully tested in terms of robustness, accuracy and scalability on parallel architecture. As part of a challenging project involving biologists mathematicians and computer scientists, the different situations engendered by the model (mono-ovulation, poly-ovulation or anovulation) from given combinations of parameters will now be systematically and intensively tested. In order to achieve this goal within realistic delay the HPC aspect of the method must be enriched using adaptive mesh refinement. The multiresolution method developed in [30]

and further tested and extended in [34] or [48] is currently implemented in this new configuration, where the presence of source terms and integral terms will require special considerations as exemplified for instance in [20].

### Acknowledgment

The authors would like to thanks

- Pascal Joly, Philippe Parnaudeau and Kyrill Pichon Gostaff<sup>3</sup> for their help concerning parallel computing,
- the CIRM<sup>4</sup> for the excellent working conditions,
- the INRIA Large Scale Initiative Action REGATE<sup>5</sup> for the funding of the project.

---

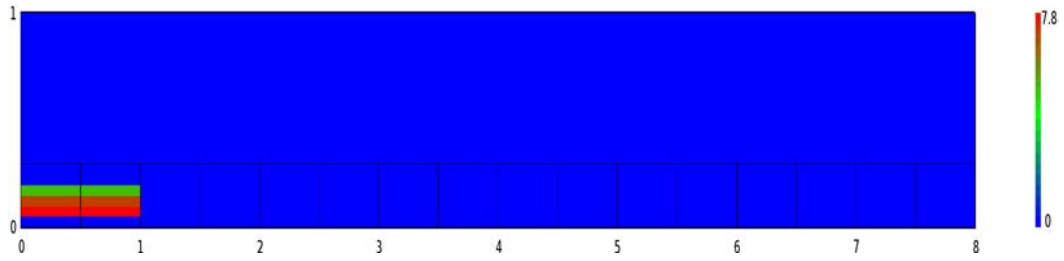
3. Laboratoire Jacques-Louis Lions, UPMC-Paris 6

4. <http://www.cirm.univ-mrs.fr>

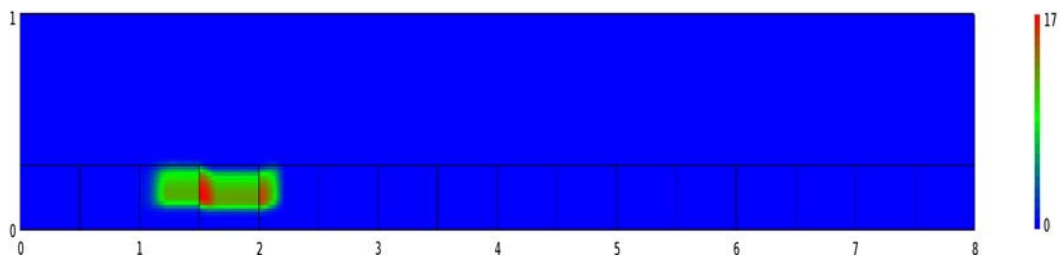
5. <https://www.rocq.inria.fr/sisyphe/reglo/regate.html>

## Chapitre 2. Numerical simulation of the selection process of the ovarian follicles

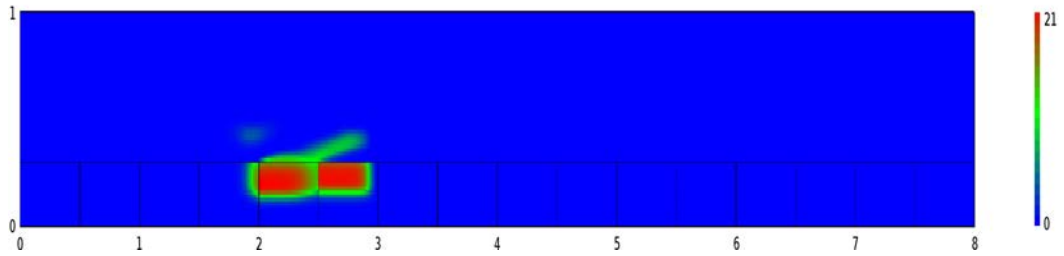
---



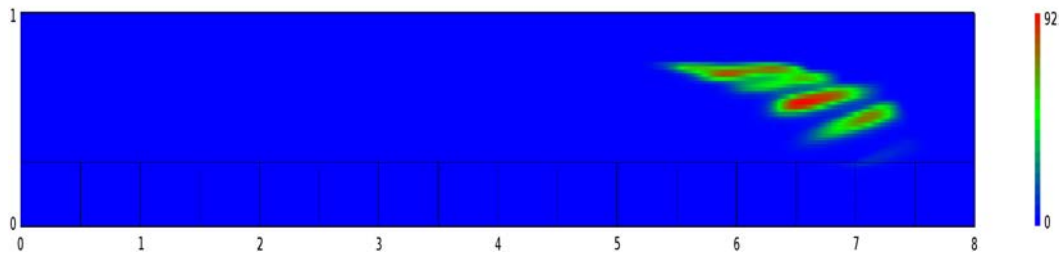
a) Initial condition. Piecewise constant in  $\gamma$  and constant in  $a$  in the first cell cycle.



b) Local increase in the density due to the decrease in the ageing function at the interface between the  $G1$  and  $SM$  phases.



c) Doubling of the cell density at the end of the second cycle and partial transfer into the differentiation phase.



d) Final density consisting of several clouds of fully differentiated cells.

FIGURE 2.4 – *Snapshots of the cell density at different times, starting from a piecewise constant density in the first cell cycle. The color code is time dependent. The full movie is available at <https://www.ljll.math.upmc.fr/aymard/movie/ovulatory.mpg>*

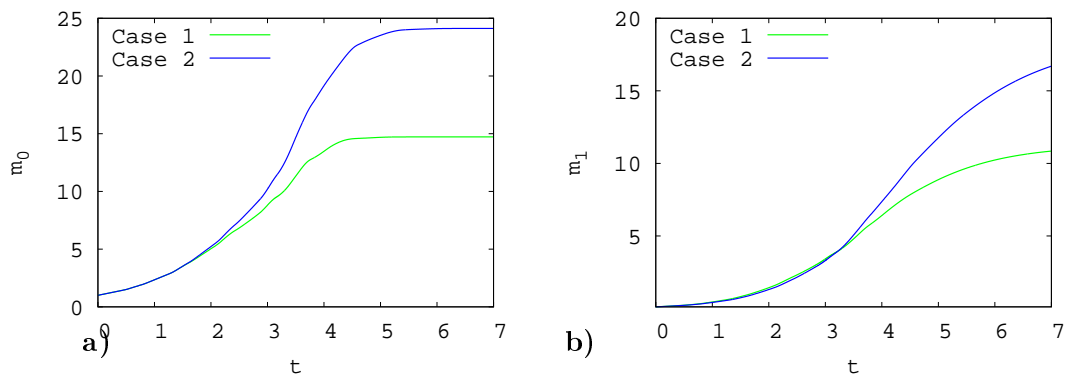


FIGURE 2.5 – a) Cell mass and b) follicular maturity time evolution for two single follicles. Case 1 corresponds to the parameters used in Figure 2.4. Case 2 differs from case 1 by the value of the time constant of the maturity function ( $\tau_h = 1$ .)

## Chapitre 2. Numerical simulation of the selection process of the ovarian follicles

---

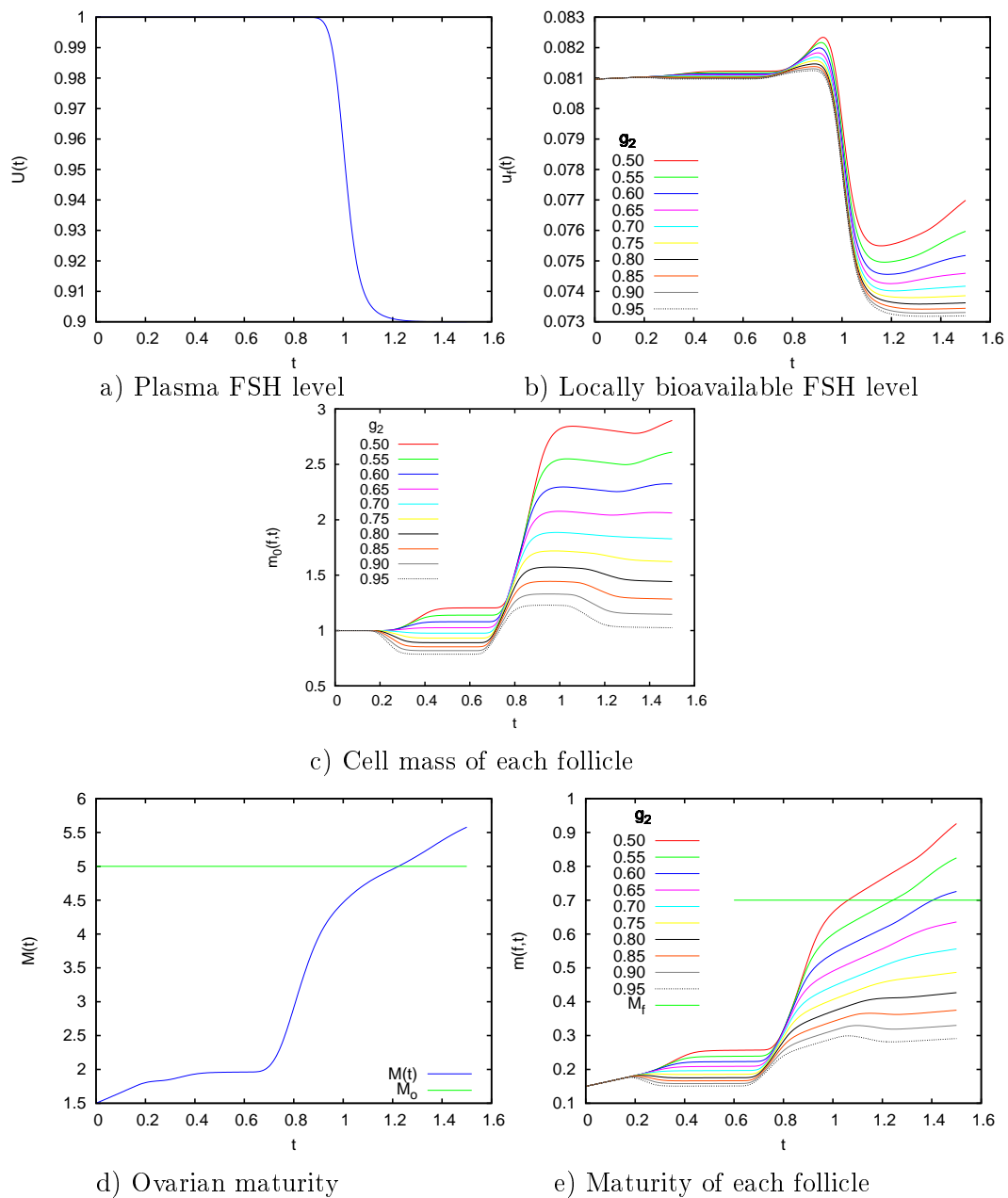


FIGURE 2.6 – Competition between ten follicles differing by their ageing function parameter  $g_2$



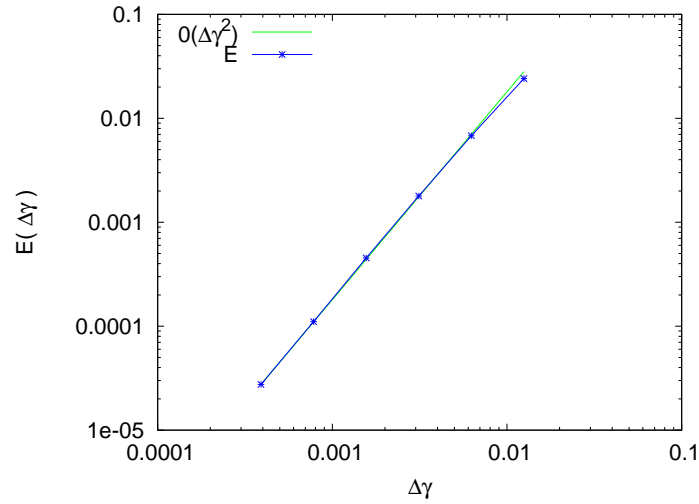


FIGURE 2.7 – Convergence rate in the SM phase. Green curve :  $\Delta\gamma \rightarrow O(\Delta\gamma^2)$ . Blue curve : L1 norm error between the exact solution and the solution with space step  $\Delta\gamma$ .

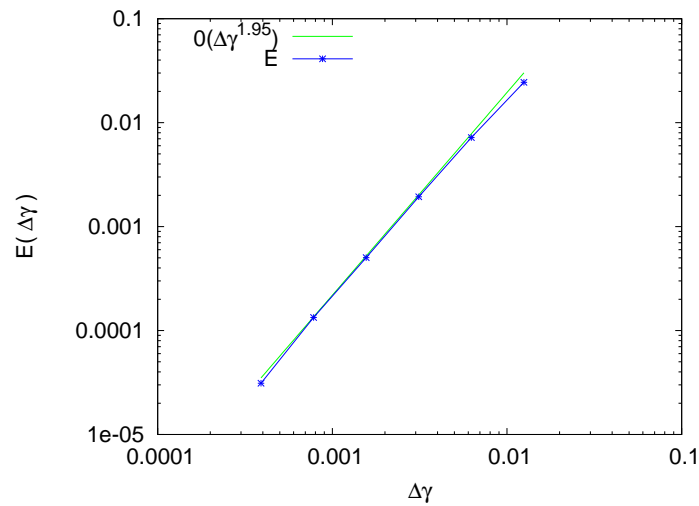


FIGURE 2.8 – CV rate on the G1 phase. Green curve :  $\Delta\gamma \rightarrow O(\Delta\gamma^{1.95})$ . Blue curve : L1 norm error between the reference solution and the solution with a space step  $\Delta\gamma$ .

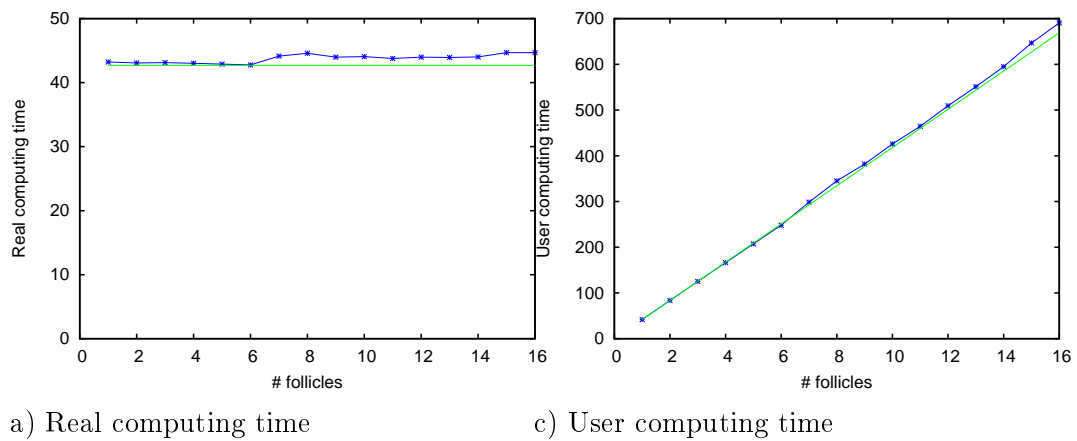


FIGURE 2.9 – *Parallelization performance test. Simulation of an increasing number of identical follicles. One follicle per processor. Green curve : theoretical limit. Blue curve : computing time (in seconds). Left : elapsed time from the start of the computation. Right : cumulative computed time on all processors*



# A numerical method for transport equations with discontinuous flux functions : application to mathematical modeling of cell dynamics

Ce chapitre a fait l'objet d'une publication [7] dans la revue SIAM Journal of Scientific Computing.

## Sommaire

---

<b>3.1</b>	<b>Introduction</b>	<b>50</b>
<b>3.2</b>	<b>Toy models</b>	<b>52</b>
<b>3.3</b>	<b>Numerical scheme</b>	<b>54</b>
3.3.1	Discretization	54
3.3.2	Treatment of discontinuous coefficients	56
<b>3.4</b>	<b>Numerical validation</b>	<b>58</b>
3.4.1	1D test cases	58
3.4.2	2D test cases	64
<b>3.5</b>	<b>Application to cell dynamics</b>	<b>69</b>
3.5.1	Numerical simulation	74
<b>3.6</b>	<b>Appendix : Exact solutions</b>	<b>75</b>
3.6.1	Exact solutions of the 1D problem with piecewise constant speeds	75
3.6.2	Exact solution of the 2D problems with piecewise constant speeds	77

---

**Abstract** In this work, we propose a numerical method to handle discontinuous fluxes arising in transport-like equations. More precisely, we study hyperbolic PDEs with flux transmission conditions at interfaces between subdomains where coefficients are discontinuous. A dedicated finite volume scheme with a limited high order enhancement is adapted to treat the discontinuities arising at interfaces. The validation of the method is done on 1D and 2D toy problems for which exact solutions are available, allowing us to do a thorough convergence study. We then apply the method to a biological model focusing on complex cell dynamics, that initially motivated this study, and illustrates the full potentialities of the scheme.

### Chapitre 3. A numerical method for transport equations with discontinuous flux functions : application to mathematical modeling of cell dynamics

---

**keywords** kinetic equations, finite volumes, discontinuous coefficients, cell dynamics

**AMS** 65M08, 35L50, 35R05, 92C37

## 3.1 Introduction

This paper is dedicated to the study of first order equations with discontinuous flux functions in the space variable. Our generic setup is the broad field of coupling problems which has received considerable attention over the past decade. Let  $\Omega \subset \mathbb{R}^n$ ,  $n \geq 1$  be a simply connected open set,  $\{\Omega_i\}_{i=1,\dots,\mathcal{N}}$  a non overlapping collection of subsets of open subsets of  $\Omega$ , with  $\cup_{i=1}^{\mathcal{N}} \overline{\Omega}_i = \overline{\Omega}$ . Then consider the phase space  $\mathcal{K} \subset \mathbb{R}^F$ , with integer  $F \geq 1$ , a convex open set. For a given  $T > 0$ , the problem reads : find  $\phi : [0, T] \times \Omega \rightarrow \mathcal{K}$  solution of

$$\begin{aligned} \partial_t \phi + \operatorname{div}(f_i(\mathbf{x}, \phi)) &= S_i(\mathbf{x}, \phi), & \text{on } (0, T) \times \Omega_i, & \text{ for all } i = 1, \dots, \mathcal{N}, \\ \phi(0, \mathbf{x}) &= \phi_0(\mathbf{x}), & \text{on } \Omega, \\ \phi(t, \mathbf{x}) &= 0, & \text{on } (0, T) \times \partial\Omega, \\ \psi_{i,j}(\phi(t, \mathbf{x}^-)) &= \phi(t, \mathbf{x}^+) & \text{on } (0, T) \times \overline{\Omega}_i \cap \overline{\Omega}_j, \end{aligned}$$

where  $f_i : \Omega_i \times \mathcal{K} \rightarrow \mathbb{R}^{n \times F}$  and  $S_i : \Omega_i \times \mathcal{K} \rightarrow \mathbb{R}^F$  are smooth functions. The last equation models the transmission condition between any pair of adjacent open sets  $\Omega_i$  and  $\Omega_j$ , hence such that  $\overline{\Omega}_i \cap \overline{\Omega}_j \neq \emptyset$ , for  $(i, j) \in \{1, \dots, \mathcal{N}\}^2$ . The transmission condition is prescribed thanks to the smooth function  $\psi_{i,j} : \mathcal{K} \rightarrow \mathcal{K}$  which allows to define the right hand trace of  $\phi(t, \mathbf{x}^+)$  at  $\overline{\Omega}_i \cap \overline{\Omega}_j$  from the left hand trace  $\phi(t, \mathbf{x}^-)$ . This setup covers a wide range of applications besides mathematical modeling of cell proliferation which is our current interest here and will be detailed in the next paragraph : multiphase flow in porous media [13], traffic flow with discontinuous road surface conditions [18], sedimentation in thickener-clarifier units [16] for instance, which have generated a lot of interest and mathematical work in recent years. In a different domain of application, hydrodynamic limit for particle systems also lead to hyperbolic conservation laws with discontinuous fluxes with  $f(x, \phi) = \lambda(x)h(\phi)$  as analyzed in [19].

In all these models, the mathematical difficulty lies in proving the well posedness of the initial boundary value problems (IBVP), as it is well summarized in the review [15]. Already in the scalar case, namely with  $F = 1$ , the  $L^1$  contraction principle cannot be inferred from the Kruzkov's analysis [51], since it would require the Lipschitz continuity of the flux in the space variable. Besides classical waves, discontinuous solutions involve additional (standing) discontinuities. On the one hand, those additional waves may prevent the existence of an *a priori* BV bound for the solutions [1]. On the other hand, they also prevent classical entropy requirements from restoring uniqueness of the solution, even for the Riemann problem. Several distinct strengthened entropy conditions have been introduced in the literature and

### Chapitre 3. A numerical method for transport equations with discontinuous flux functions : application to mathematical modeling of cell dynamics

---

we refer for instance the reader to [1], [5], [19] and [15] for a review and exhaustive references. Distinct compactness frameworks based on these entropy conditions have then been proposed to prove well posedness of the problem (see again [1, 5, 19, 15]). Here we deal with a non conservative coupling setting where the velocities are non-local. As highlighted below, these two new mathematical features arise naturally in a biological context.

We address the model developed in [41] in the general context of cell dynamics, describing the time evolution of a density function depending on age and maturity variables ( $n = 2$ ). This unknown is governed by a kinetic like equation involving velocities that are function of integro-moments of the unknown and the age and maturity variables. Closure equations for these velocities are naturally discontinuous in the age and maturity variables, precisely at biological checkpoints which correspond to the interfaces between the biological phases [77]. These discontinuities require additional information which are handled as local double IBVP, where inner boundary conditions are formulated to express the biological switch.

The conservation hypothesis must clearly be abandoned : cell proliferation (through mitosis<sup>1</sup>) is modeled by a non conservative coupling and cell death (through apoptosis) also alters the global mass of the system. An additional specificity of the equation lies in its quasi linear integro differential nature. The flux functions  $f_i$  are of the form

$$f_i(\mathbf{x}, \phi) = u_i(\mathbf{x}, \phi)\phi, \quad \text{for } i = 1, \dots, \mathcal{N}.$$

They arise from the weak dependence of the velocity field  $u$  on the solution  $\phi$  through a single moment  $m_f(t)$

$$u_i(\mathbf{x}, \phi) = v_i(\mathbf{x}, \{m_f\}_{f=1, \dots, F}), \quad \text{with } m_f(t) = \int_{\Omega} y\phi_f(t, \mathbf{x})d\mathbf{x}, \quad \text{where } \mathbf{x} = (x, y).$$

More details on the biological background of the model will be provided in Section 4.5 which is devoted to its numerical simulation.

The theoretical study of this model has been done in [71]. The main difficulties come from the nonlocal velocity, the coupling between boundary conditions and the vector nature of the coupling problem ( $F > 1$ ). Roughly speaking, the specific moment  $m_f(t)$  of the solution entering the model is shown to exist as the fix point of a map from the space of continuous functions. This leads to the existence and uniqueness of first a local weak solution, from which a global weak solution is then built. Various tools are at hand to compute the numerical approximation of this solution. Let us first mention that the general case of transport equations with discontinuous equations was first studied in a series of work by Bouchut and James [12] using duality arguments. Another standpoint is to address those discontinuities in terms of a coupling problem, as pioneered by Godlewski and Raviart [43]. We will adopt this later setting and show that it is well suited to our purpose. We will use techniques proposed in [43] and subsequent studies [3, 13]. Here we pay special

---

1. Process in cell proliferation through which a mother cell gives birth to two daughter cells

### Chapitre 3. A numerical method for transport equations with discontinuous flux functions : application to mathematical modeling of cell dynamics

---

attention to the extension of these algorithms to higher order accuracy, which up to our knowledge, have not yet been addressed in 2D simulations for non conservative coupling and also barely in 2D in general. Although the well posedness issue is not handled with the same mathematical tools as in [71], the numerical methods that we have designed in this paper can be used in the application contexts cited above, like for instance traffic models [70].

#### Outline

From the numerical point of view we are focusing in this work on the discontinuous non conservative fluxes. The remaining of the paper is therefore organized in four sections. In Section 2, we describe toy models consisting of linear hyperbolic PDE with piecewise constant speeds and linear source term, in 1D and 2D. In Section 3, we present the numerical scheme, first recalling the Finite Volume scheme and then specifically dealing with the transmission conditions. For piecewise constant speed linear transport with or without linear source term, we design a third order scheme by combining a limited reconstruction in space and a Runge Kutta time quadrature. We describe in details the numerical scheme at the interfaces where the speed and source coefficients are discontinuous, with or without discontinuity of the flux, in order to maintain the precision of the overall computation.

The numerical validation is performed in Section 4 on several 1D and 2D test cases and based on a thorough convergence study. Whenever it is possible, the exact solutions are used as references (the derivation of the analytical solutions is postponed to the Appendix). Finally, we present in Section 5 the biological model describing the cell dynamics in developing ovarian follicles [41] which has motivated our work and perform simulations that validate the numerical method for this application. If this model has already been subject to numerical simulations based on finite volumes (see [39, 41] for instance), the first rigorous numerical test was performed recently in a preliminary study [6], that only dealt with low-order treatment of discontinuities. The novelty here is the high order enhancement at discontinuous interfaces, which is a particularly challenging question.

## 3.2 Toy models

In this section we present generic toy models with piecewise constant coefficients for which exact solutions are available, and which we will use to validate the numerical method. The weak nonlinearity arising in biological models from the dependence of the speeds and source term on the moments of the solution is left out. It is therefore

### Chapitre 3. A numerical method for transport equations with discontinuous flux functions : application to mathematical modeling of cell dynamics

---

sufficient to consider a simplified scalar model defined for  $(x, y) \in \Omega = \mathbb{R}^2, t > 0$ .

$$\begin{cases} \frac{\partial \phi(x, y, t)}{\partial t} + \frac{\partial (g(x, y)\phi(x, y, t))}{\partial x} + \frac{\partial (h(x, y)\phi(x, y, t))}{\partial y} = -\Lambda(x, y)\phi(x, y, t) \\ \phi(x, y, 0) = \phi_0(x, y) & \text{(initial condition)} \\ \psi_L(g(x_s^-, y)\phi(x_s^-, y, t)) = g(x_s^+, y)\phi(x_s^+, y, t) & \text{(flux cond. on } \overline{\Omega_1} \cap \overline{\Omega_2}) \\ \psi_B(h(x, y_s^-)\phi(x, y_s^-, t)) = h(x, y_s^+)\phi(x, y_s^+, t) & \text{(flux cond. on } \overline{\Omega_i} \cap \overline{\Omega_3}, \text{ for } i = 1, 2) \end{cases} \quad (3.1)$$

with the piecewise constant velocities  $g(x, y)$  and  $h(x, y)$  and source term  $\Lambda(x, y)$  defined in the subregions depicted on Figure 3.1, namely  $\Omega_1 = \{(x, y) \in \mathbb{R}^2, x < x_s, y < y_s\}$ ,  $\Omega_2 = \{(x, y) \in \mathbb{R}^2, x > x_s, y < y_s\}$  and  $\Omega_3 = \{(x, y) \in \mathbb{R}^2, y > y_s\}$ . To fix ideas, we can already assume that  $\phi(x, y)$  represents a density of cells, and that its integral over  $\Omega$  is the total mass of the system. Furthermore, thanks to the simplicity of the geometry, we can use a cartesian grid discretization and a numerical scheme deduced from the 1D problem by tensorization. We will therefore describe the numerical method on the following 1D toy problem

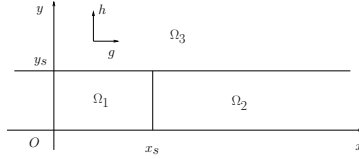


FIGURE 3.1 – Spatial set-up for the 2D toy model.  $g$  and  $h$  respectively denote the velocities in the  $x$  and  $y$  directions.

$$\begin{cases} \partial_t \phi(x, t) + \partial_x g(x)\phi(x, t) = -\Lambda(x)\phi(x, t) & \text{for } x \in \mathbb{R}, t > 0 \\ \phi(x, 0) = \phi_0(x) & \text{(initial condition)} \\ \psi_L(g(x_s^-)\phi(x_s^-, t)) = g(x_s^+)\phi(x_s^+, t) & \text{(flux condition)} \end{cases} \quad (3.2)$$

with

$$g(x) = \begin{cases} g_L, & \text{for } x < x_s, \\ g_R & \text{elsewhere,} \end{cases} \quad \Lambda(x) = \begin{cases} \Lambda_L, & \text{for } x < x_s, \\ \Lambda_R & \text{elsewhere.} \end{cases} \quad (3.3)$$

In order to deal with all the situations encountered in the real-life problem to be addressed in Section 4.5, the transmission conditions in (3.2) will be in turn

$$\psi_L(z) = z \quad \text{(flux continuity between the left and right zones)} \quad (3.4)$$

$$\psi_L(z) = 2z \quad \text{(doubling of flux between the left and right zones)} \quad (3.5)$$

$$\psi_L(z) = 0 \quad \text{(waterproof interface between the left and right zones)} \quad (3.6)$$

and the source term will be in turn defined by

$$\Lambda_L = \Lambda_R = 0 \quad \text{(no source)} \quad (3.7)$$

$$\Lambda_L = 1, \Lambda_R = 0 \quad \text{(source in the left zone only)} \quad (3.8)$$

$$\Lambda_L = 0, \Lambda_R = 1 \quad \text{(source in the right zone only)} \quad (3.9)$$



### 3.3 Numerical scheme

We now turn to the numerical method. We first recall the finite volume scheme in the continuous regions, and then present the method that we have designed to deal with the transmission conditions.

#### 3.3.1 Discretization

In this paragraph we describe the numerical scheme for the simplified problems (3.1) and (3.2). To solve (3.1) we restrict the computational domain to  $[0, L_x] \times [0, L_y]$ , enforcing periodical boundary conditions on the outer boundaries. We set the domain dimensions as  $L_x = L_y = 1$  and we select the initial condition such that, at all times, the outer boundaries are far enough from the region where the solution is not zero. We can therefore compare the numerical solution obtained when applying periodic boundary conditions with the exact solutions described in the Appendix. We denote by  $N_x$  the number of grid meshes in each direction and by  $\Delta x = L_x/N_x$  the space step. The step needs to be chosen so that the locations of the interfaces  $x_s$  and  $y_s$ , where the speed coefficients  $g(x, y)$  and  $h(x, y)$  are discontinuous, fall on grid points. From now on we focus on the 1D finite volume scheme to solve (3.2), since the 2D scheme can be obtained by tensorization. Along the spatial grid

$$x_k = k\Delta x, \quad x_{k+1/2} = (k + 1/2)\Delta x, \quad \text{for } k = 0, \dots, N_x, \quad (3.10)$$

the time discretization is defined by

$$t^0 = 0, \quad t^{n+1} = t^n + \Delta t^n, \quad \text{for } n = 0, \dots, N \quad (3.11)$$

with  $N$  such that  $t^N = t_{\text{final}}$ , and time steps  $\Delta t^n$  that may change at each iteration, in order to preserve stability. The unknowns are the approximate mean values of the solution in each grid mesh  $[x_k, x_{k+1}]$

$$\phi_k^n \approx \frac{1}{\Delta x} \int_{x_k}^{x_{k+1}} \phi(x, t^n) dx. \quad \text{for } k = 0, \dots, N_x - 1.$$

Equation (3.2) is then explicitly discretized to obtain a recursion formula

$$\phi_k^{n+1} = \phi_k^n - \frac{\Delta t^n}{\Delta x} (F_{k+1}(\phi^n) - F_k(\phi^n)) - \Delta t^n \Lambda(x_{k+\frac{1}{2}}) \phi_k^n \quad (3.12)$$

where  $F_k$  is the numerical flux across  $x_k$ , designed using a limiter strategy. Indeed, it is well known that first order schemes, like the Godunov scheme, are diffusive, and that second order schemes, like the Lax-Wendroff scheme, generate oscillations in the neighborhood of discontinuities. In order to get a stable as well as precise scheme, we take a weighting of a low order scheme and high order scheme, and we define the limited numerical flux

$$F_k = F_k^{Low} + \ell(r_k)(F_k^{High} - F_k^{Low}), \quad (3.13)$$

### Chapitre 3. A numerical method for transport equations with discontinuous flux functions : application to mathematical modeling of cell dynamics

---

where  $\ell$  is the limiter function designed by Koren [50]

$$\ell_{Koren}(r) = \max(0, \min(2r, \frac{2+r}{3}, 2)),$$

and  $r_k$  is defined by

$$\left\{ \begin{array}{l} r_k = \mathcal{R}((g_{k+i}\phi_{k+i})_{i=-2,\dots,1}; (g_{k+i})_{i=-2,\dots,1}) \\ \mathcal{R}((z_{k+i})_{i=-2,\dots,1}; (g_{k+i})_{i=-2,\dots,1}) = \begin{cases} \frac{z_{k-1} - z_{k-2}}{z_k - z_{k-1}} & \text{if } g_{k+i} \geq 0 \quad \forall i = -2, \dots, 1 \\ \frac{z_{k+1} - z_k}{z_k - z_{k-1}} & \text{if } g_{k+i} \leq 0 \quad \forall i = -2, \dots, 1 \\ 0 & \text{otherwise,} \end{cases} \end{array} \right. \quad (3.14)$$

with the notation  $g_k = g(x_k)$ . This ratio is a good indicator of the regularity of the function (see [72]). In fact, a steep gradient or a discontinuity gives a ratio far from 1, whereas a smooth function gives a ratio close to 1.

The first order flux entering equation (3.13) is the Godunov flux

$$F_k^{Low}(\phi^n) = (g_{k-1}^n)^+ \phi_{k-1}^n + (g_k^n)^- \phi_k^n, \quad (3.15)$$

and the high order flux is the Lax-Wendroff one, which is of second order in space wherever the function  $g(x)$  is continuous.

$$F_k^{High}(\phi^n) = \frac{1}{2}(g_{k-1}^n \phi_{k-1}^n + g_k^n \phi_k^n), \quad (3.16)$$

The CFL stability condition is

$$\Delta t^n \leq \min \left( CFL \frac{\Delta x}{\max_k |g_k|}, \frac{1}{\max_k |\Lambda_{k+\frac{1}{2}}|} \right) \quad (3.17)$$

with  $CFL \leq \frac{1}{2}$  and the notation  $\Lambda_{k+1/2} = \Lambda(x_{k+1/2})$ .

Choosing Koren limiter in (3.13) provides third order in space for the convective part of the equation in each domain where it is continuously defined. The source term being discretized by a center point quadrature is at most second order in each domain. Second order in time (third order when  $\Lambda = 0$ ) is achieved by a third order Runge-Kutta method ([44])

$$\begin{aligned} \mathcal{B}_k^1 &= F_{k+1}(\phi^n) - F_k(\phi^n) + \Delta x \Lambda_{k+1/2} \phi_k^n, \\ \phi_k^* &= \phi_k^n - \frac{\Delta t^n}{\Delta x} \mathcal{B}_k^1, \\ \mathcal{B}_k^2 &= F_{k+1}(\phi^*) - F_k(\phi^*) + \Delta x \Lambda_{k+1/2} \phi_k^*, \\ \phi_k^{**} &= \phi_k^n - \frac{1}{4} \frac{\Delta t^n}{\Delta x} \mathcal{B}_k^1 - \frac{1}{4} \frac{\Delta t^n}{\Delta x} \mathcal{B}_k^2, \\ \mathcal{B}_k^3 &= F_{k+1}(\phi^{**}) - F_k(\phi^{**}) + \Delta x \Lambda_{k+1/2} \phi_k^{**}, \\ \phi_k^{n+1} &= \phi_k^n - \frac{1}{6} \frac{\Delta t^n}{\Delta x} \mathcal{B}_k^1 - \frac{1}{6} \frac{\Delta t^n}{\Delta x} \mathcal{B}_k^2 - \frac{2}{3} \frac{\Delta t^n}{\Delta x} \mathcal{B}_k^3, \end{aligned}$$

We will now focus on the interface between two subdomains where function  $g$  can be discontinuous, neglecting the source term for the time being.

### 3.3.2 Treatment of discontinuous coefficients

The domain is discretized in such a manner that the interface between two subregions where  $g$  is continuous is placed on an edge between two grid meshes. Let us consider that the interface position  $x_s = x_K$  is at the interface between meshes  $[x_{K-1}, x_K]$  and  $[x_K, x_{K+1}]$  as depicted on Figure 3.2. To adapt the finite volume scheme designed in

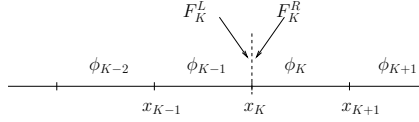


FIGURE 3.2 – Left and right flux surrounding the interface between meshes  $[x_{K-1}, x_K]$  and  $[x_K, x_{K+1}]$ .

the previous paragraph so that it can handle the transmission condition, we define for each interface at grid point  $x_k$  two fluxes : the left flux  $F_k^L$  and the right flux  $F_k^R$  (Figure 3.2). Then we rewrite equation (3.12) as

$$\phi_k^{n+1} = \phi_k^n - \frac{\Delta t^n}{\Delta x} (F_{k+1}^L - F_k^R)$$

If there is no transmission condition on the flux, we have

$$F_k^L = F_k^R = F_k$$

defined by (3.13). In contrast, if there is a transmission condition such as (3.5), we set

$$F_k^R = \psi_L(F_k^L). \quad (3.18)$$

The value of the numerical fluxes defined by (3.15) and (3.16) are derived for smooth coefficients  $g(x)$  and continuous fluxes  $g(x)\phi(x)$ . They have therefore to account for the transmission condition  $\psi_L$ , seen from the left side of  $x_s$ . We define, at the interface  $K$ , the first order flux

$$F_K^{Low,L}(\phi^n) = (g_{K-1}^n)^+ \phi_{K-1}^n + \frac{1}{2} (1 - \text{sign}(g_K^n)) \psi_L^{-1}(g_K^n \phi_K^n), \quad (3.19)$$

and the second order flux

$$F_K^{High,L}(\phi^n) = \frac{1}{2} (g_{K-1}^n \phi_{K-1}^n + \psi_L^{-1}(g_K^n \phi_K^n)). \quad (3.20)$$

Furthermore, the high order enhancement using a limited combination such as (3.13) is affected by the presence of an interface not only at  $x_K$  but also in its vicinity. Indeed, if the velocity  $g(x)$  is positive, the limiter “sees” interface  $K$  at neighboring interfaces  $k = K - 1, \dots, K + 1$  through the ratio  $r_k$ . The transmission condition (3.18) causes the ratio  $r_k$  (3.14) to depart from 1 on these interfaces and consequently induces a loss of accuracy in the numerical scheme. This can be avoided

**Chapitre 3. A numerical method for transport equations with discontinuous flux functions : application to mathematical modeling of cell dynamics**

---

by computing the ratio  $r_k$  from the continuous quantities seen from  $x_k$ . Namely, at interfaces  $K-1, K, K+1$ , the limiter is computed using

$$\begin{cases} r_{K-1} = \mathcal{R}(g_{K-3}\phi_{K-3}, g_{K-2}\phi_{K-2}, g_{K-1}\phi_{K-1}, \psi_L^{-1}(g_K\phi_K); (g_{K+i})_{i=-3,\dots,0}) \\ r_K = \mathcal{R}(\psi_L(g_{K-2}\phi_{K-2}), \psi_L(g_{K-1}\phi_{K-1}), g_K\phi_K, g_{K+1}\phi_{K+1}; (g_{K+i})_{i=-2,\dots,1}) \\ r_{K+1} = \mathcal{R}(\psi_L(g_{K-1}\phi_{K-1}), g_K\phi_K, g_{K+1}\phi_{K+1}, g_{K+2}\phi_{K+2}; (g_{K+i})_{i=-1,\dots,2}) \end{cases} \quad (3.21)$$

instead of (3.14).

To fix ideas, we apply this method in cases (3.4) and (3.5) encountered in our application model.

**3.3.2.1 First application : Flux continuity with discontinuous speed**

In this case the transmission condition is

$$\psi_L = Id.$$

At interface  $K$ , the condition is

$$(g\phi)_K^R = (g\phi)_K^L$$

with

$$g_{K-1} \neq g_K.$$

The transmission condition (3.18) provides the flux from the right side of the interface

$$F_K^R = F_K^L.$$

The scheme works well without modification, both at first and third order.

**3.3.2.2 Second application : Doubling flux**

In this case the transmission condition (3.18) is

$$\psi_L(\mathcal{F}) = 2\mathcal{F}.$$

At interface  $K$ , the condition is

$$(g\phi)_K^R = 2(g\phi)_K^L,$$

with

$$g_{K-1} \neq g_K.$$

The flux on the left of the interface is computed using continuous values in the definitions of (3.15) and (3.16). From the left side, the continuous values are  $g\phi$  for  $x < x_K$  and  $\psi_L(g\phi)$  for  $x > x_K$ . Therefore we write for the first order flux

$$F_K^{low,L} = (g_{K-1})^+ \phi_{K-1} + \frac{(g_K)^- \phi_K}{2},$$

### Chapitre 3. A numerical method for transport equations with discontinuous flux functions : application to mathematical modeling of cell dynamics

---

and for the second order flux

$$F_K^{high,L} = \frac{g_{K-1}\phi_{K-1} + \frac{g_K\phi_K}{2}}{2}.$$

Concerning the limiter, it has to take into account the doubling condition on three meshes in the vicinity of the interface. At interfaces  $K-1, K, K+1$  the limiter (3.21) is computed from the continuous quantities

$$\begin{cases} r_{K-1} = \mathcal{R} \left( g_{K-3}\phi_{K-3}, g_{K-2}\phi_{K-2}, g_{K-1}\phi_{K-1}, \frac{g_K\phi_K}{2}; (g_{K+i})_{i=-3,\dots,0} \right), \\ r_K = \mathcal{R} (2g_{K-2}\phi_{K-2}, 2g_{K-1}\phi_{K-1}, g_K\phi_K, g_{K+1}\phi_{K+1}; (g_{K+i})_{i=-2,\dots,1}), \\ r_{K+1} = \mathcal{R} (2g_{K-1}\phi_{K-1}, g_K\phi_K, g_{K+1}\phi_{K+1}, g_{K+2}\phi_{K+2}; (g_{K+i})_{i=-1,\dots,2}). \end{cases}$$

The transmission condition (3.18) provides the flux from the right side of the interface

$$F_K^R = 2F_K^L.$$

## 3.4 Numerical validation

In the sequel, we present several numerical simulations to substantiate the validation of our method. Two 1D situations are handled, one with a flux continuity (3.4), another with a doubling at the interface (3.5), with or without a source term (3.7), (3.8), (3.9). We then present two 2D test cases : a shear phenomenon which is encountered between zones  $\Omega_1$  and  $\Omega_3$ , and a waterproof condition (3.6), between zones  $\Omega_2$  and  $\Omega_3$ .

### 3.4.1 1D test cases

For this set of test cases we can compare the numerical solution at final time  $t^N$  with the exact solution described in Appendix 3.6.1 and compute the  $L1$ -norm relative error

$$E_{\Delta x} = \frac{\sum_{k=0}^{N_x} |\phi_k^N - \bar{\phi}_k^N|}{\sum_{k=0}^{N_x} |\bar{\phi}_k^N|}$$

where  $\bar{\phi}_k^N$  is the mean value of the exact solution at final time on mesh  $[x_k, x_{k+1}]$ . This mean value is itself estimated with a quadrature formula – the second order central point formula. This is actually justified for the 1D test cases where the discontinuity of the solution coincides with a grid point. In order to recover the expected behavior the initial condition is a smooth gaussian function of total mass equal to 1 and centered on  $c_x=0.3$ .

$$\phi_0(x) = \frac{1}{2\pi\sigma} \exp\left(\frac{1}{2} \left(\frac{(x - c_x)^2}{\sigma^2}\right)\right). \quad (3.22)$$

**Chapitre 3. A numerical method for transport equations with discontinuous flux functions : application to mathematical modeling of cell dynamics**

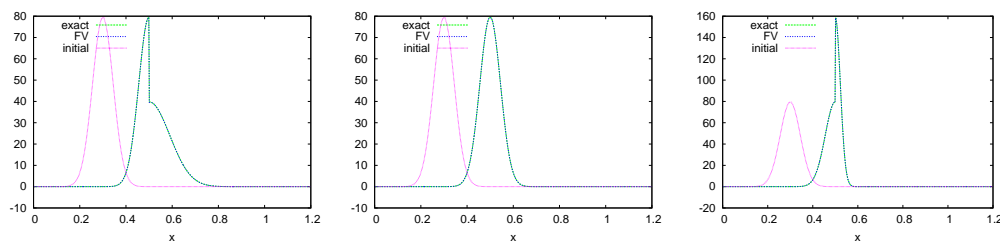
---

test	description	$\psi_L(\mathcal{F})$	$g_L$	$g_R$	$\Lambda_L$	$\Lambda_R$	p	$E.10^4$
1	Continuous flux, speed discontinuity on $\Omega_1$ - $\Omega_2$ interface, no source	$\mathcal{F}$	0.5	1	0	0	2.35	9.23
			1				2.39	9.39
			2				2.40	9.74
			3				2.29	9.71
2	Doubling flux, speed discontinuity on $\Omega_1$ - $\Omega_2$ interface, no source	$2\mathcal{F}$	1	0.5	0	0	2.40	10.25
				1			2.38	9.81
				2			2.35	9.64
				3			2.34	9.61
3	Continuous flux, speed discontinuity on $\Omega_1$ - $\Omega_2$ interface, source in $\Omega_1$	$\mathcal{F}$	0.5	1	1	0	2.35	8.89
			1				2.39	9.22
			2				2.37	9.47
			3				2.28	9.63
4	Doubling flux, speed discontinuity on $\Omega_1$ - $\Omega_2$ interface, source in $\Omega_1$	$2\mathcal{F}$	1	0.5	0	1	2.42	10.34
				1			2.36	9.68
				2			2.34	9.60
				3			2.33	9.58

TABLE 3.1 – 1D test cases description, asymptotic order  $p$  of the L1-norm error  $O(\Delta x^p)$  and relative error value for  $\Delta x = 10^{-3}$ . Each test case is run for four different speed contrasts, leading to slightly different asymptotic behavior.

### Chapitre 3. A numerical method for transport equations with discontinuous flux functions : application to mathematical modeling of cell dynamics

---



a)  $g_L = 0.5, t^N = 0.4$       b)  $g_L = 1, t^N = 0.2$       c)  $g_L = 2, t^N = 0.1$

FIGURE 3.3 – 1D test case 1 : flux continuity condition and velocity jump at interface  $x_s = 0.5$ . Snapshot of the density  $\phi(x, t)$  at initial time and when it crosses the interface between zones  $\Omega_1$  and  $\Omega_2$  ( $t = t^N$ ).

#### Test case 1. Continuous flux, speed discontinuity on $\Omega_1$ - $\Omega_2$ interface, no source

This tests mimics what happens at the transition (3.4) between zones  $\Omega_1$  and  $\Omega_2$ . The speed is equal to 1 in the second half of the domain, and takes values 0.5, 1 (for reference), 2 or 3 in the first half of the domain. As described in Appendix 3.6.1 the exact solution in that case is

$$\phi^{exact}(x, t) = \begin{cases} \phi_0(x - g_L t) & \text{for } x < x_s, \\ g_L \phi_0(x_s - g_L(t - x - x_s)) & \text{for } x_s - t \leq x - t < x_s, \\ \phi_0(x - t) & \text{for } x - t \geq x_s, \end{cases}$$

In Figure 3.3, we plot the density  $\phi$  at a time  $t^N = 0.2/g_L$ , chosen so that half the mass has already crossed the interface between zones  $\Omega_1$  and  $\Omega_2$ . Since the flux is continuous at  $x_s = 0.5$  and the speed is discontinuous, the solution presents a discontinuity (except for the special case  $g_L = 1$ ). We can notice the good quality of the approximation, compared to the exact solution. There are no oscillations in the neighborhood of the discontinuity, neither numerical diffusion. Remark that if the speed decreases much on the right side of the interface, the density shape sharpens. This is qualitatively similar to what happens for a traffic flow when there is a sudden speed limitation. This phenomenon penalizes the accuracy. In panel b) of Figure 3.4, the  $L1$ -norm error with respect to the space step  $\Delta x$  is displayed in logarithmic scale, which allows us to compute the convergence rate. Depending on the precise value of  $g_L$ , the convergence rate varies between 2.26 and 2.4. More precisely, it gets worse as the value of  $g_L$  increases. This can be explained by looking at the contrast between  $g_L$  and  $g_R = 1$ . By analogy with the traffic flow, the interface between a high speed on the left and a lower one on the right hand side corresponds to a sudden speed decrease which generates an accumulation of traffic. The density profile becomes narrower and it becomes harder to make accurate simulations. In contrast, when the speed is higher on the right, the density profile becomes smoother

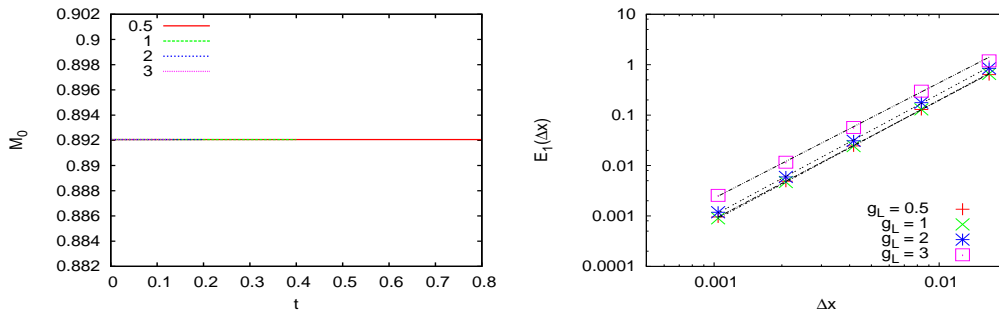
### Chapitre 3. A numerical method for transport equations with discontinuous flux functions : application to mathematical modeling of cell dynamics

---

and it becomes easier to approximate the solution. In any case, the best order of convergence 2.4 is achieved when  $g_L = g_R = 1$ . In panel a) of Figure 3.4, the total mass

$$m_0(t) = \int_0^{L_x} \phi(x, t) dx$$

is displayed with respect to the time to check whether the scheme is conservative. We also perform a 2D generalization of this test in the case  $g_L = 2$ , with a smooth



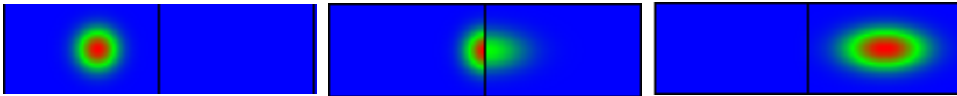
a) Mass with respect to time                      b) Error convergence with respect to  $\Delta x$

FIGURE 3.4 – 1D test case 1 : flux continuity condition and velocity jump at interface  $x_s = 0.5$ . In panel a) the total mass remains constant with time. In panel b) the  $L_1$ -norm error goes to 0 with  $\Delta x$ . The convergence rates (around 2.4) are gathered in Table 3.4.1.

initial condition

$$\phi_0(x, y) = \frac{1}{2\pi\sigma^2} \exp\left(\frac{1}{2} \left( \frac{(x - c_x)^2}{\sigma^2} + \frac{(y - c_y)^2}{\sigma^2} \right)\right). \quad (3.23)$$

centered on  $(c_x, c_y) = (0.3, 0.15)$  and the same variance  $\sigma^2 = 0.002$  as before. The corresponding results are displayed on Figure 3.5.



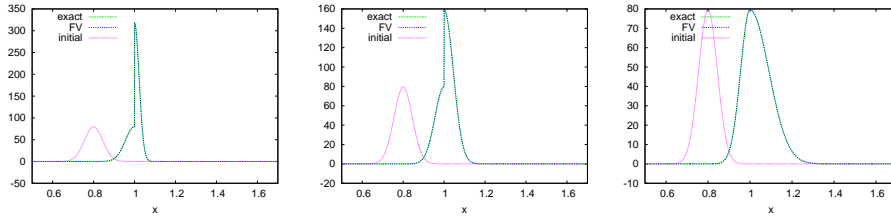
a) Initial time.                      b) Intermediate time.                      c) Final time  $t = 0.25$ .  
The density bump lies in                      Density crossing                      The density bump lies in  
zone  $\Omega_1$ .                                      interface at  $t = 0.15$ .                      zone  $\Omega_2$ .

FIGURE 3.5 – 2D visualization of test case 1 : flux continuity condition and velocity jump at interface  $x_s = 0.5$ ,  $u_L = 0.5$ ,  $u_R = 1$ . Snapshot of the density (initial time, passing interface, final time). ( $CFL = 0.4$ ,  $\Delta x = 0.001$ ).



**Chapitre 3. A numerical method for transport equations with discontinuous flux functions : application to mathematical modeling of cell dynamics**

---



a)  $g_R = 0.5$

b)  $g_R = 1$

c)  $g_R = 2$

FIGURE 3.6 – 1D test case 2 : doubling flux condition and velocity jump at interface  $x_s = 0.5$ . Snapshot of the density  $\phi(x,t)$  at initial time and when it crosses the interface between zones  $\Omega_1$  and  $\Omega_2$  ( $t = 0.2$ ).

**Test case 2. Doubling flux, speed discontinuity on  $\Omega_1$ - $\Omega_2$  interface, no source**

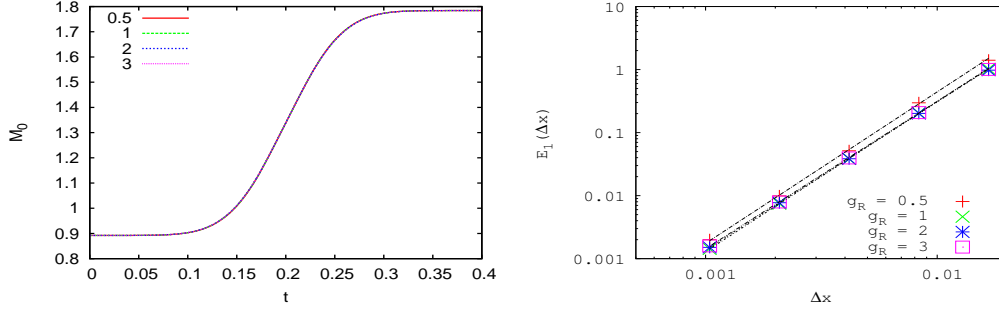
The second test case addresses flux discontinuity described in paragraph 3.3.2.2. The speed is equal to  $g_L = 1$  for  $x < x_s$  and takes successively the values  $g_R = 0.5, 1, 2$  or 3 for  $x > x_s$ . This time the exact solution is

$$\phi^{exact}(x,t) = \begin{cases} \phi_0(x-t) & \text{for } x < x_s, \\ \frac{2}{g_R}(\phi_0(x_s - t - \frac{(x-x_s)}{g_R})) & \text{for } (x_s - g_R t) \leq (x - g_R t) < x_s, \\ \phi_0(x - g_R t) & \text{for } (x - g_R t) \geq x_s, \end{cases}$$

In Figure 3.6, we plot the density  $\phi$  at time  $t = 0.2$ . This corresponds to the time at which half the mass has already crossed the interface between zones  $\Omega_2$  and  $\Omega_1$ . The solution is continuous when  $g_R = 2$ , but exhibits a discontinuity in slope. We can draw the same conclusions as for the first test case. In particular, there are no oscillations in the neighborhood of the discontinuity, even if there is a discontinuity in the solution.

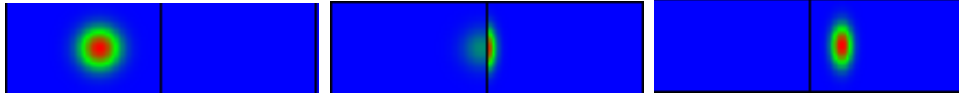
In panel b) of Figure 3.7 we plot in a logarithmic scale the  $L1$ -norm error with respect to the space step  $\Delta x$ . The convergence rates, gathered in Table 3.4.1, are similar to the first test case, which means that the transition condition has not reduced the precision of the scheme. We can notice that the lower the  $g_R$ , the worse the error. This is a similar situation to the first test case which can be explained by the same analogy with the traffic flow. In panel a) of Figure 3.7 we plot the mass with respect to time. Once all the cells have been transported in the right subdomain, the mass is doubled. As expected, since the speed is always  $g_L = 1$  in the left subdomain, in all cases the cells reach the interface at the same time and the mass profiles are identical. Figure 3.8 displays the 2D generalization of the second test case.

### Chapitre 3. A numerical method for transport equations with discontinuous flux functions : application to mathematical modeling of cell dynamics



a) mass with respect to time      b)  $L_1$ -norm error with respect to  $\Delta x$

FIGURE 3.7 – 1D test case 2. Doubling flux condition and velocity jump at interface  $x = x_s$ . The mass in panel a) has doubled when all the density has reached the second subdomain, as expected. In panel b) the  $L_1$ -norm error goes to zero with  $\Delta x$ . The convergence rates (around 2.4) are displayed in Table 3.4.1.



a) Initial time. The density bump is in zone  $\Omega_1$ .      b) Intermediate time. Density crossing interface at  $t = 0.16$ .      c) Final time  $t = 0.4$ . The density bump is in zone  $\Omega_2$ .

FIGURE 3.8 – 2D visualization of test case 2. Doubling flux condition and velocity jump at interface  $x = x_s$ ,  $u_L = 1$ ,  $u_R = 0.5$ . Snapshots of the density (initial time, passing interface, final time). The mass is doubling. ( $CFL = 0.4$ ,  $\Delta x = 0.001$ ).

#### Test case 3. Continuous flux, speed discontinuity on $\Omega_1$ - $\Omega_2$ interface, linear source on the left

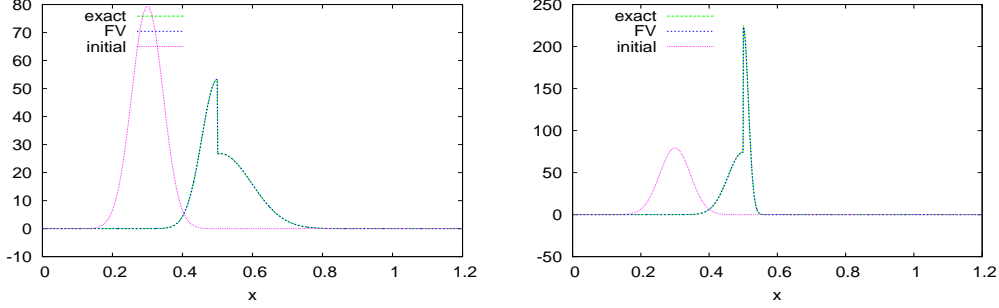
The third test case is the same as the first one except that we add a linear source term  $\Lambda_L = 1$  in the left part of the domain. The exact solution, detailed in Appendix, is

$$\phi^{exact}(x, t) = \begin{cases} \phi_0(x - g_L t) \exp(-t) & \text{for } x < x_s, \\ g_L \phi_0(x_s - g_L(t - x - x_s)) \exp(-t + x - x_s) & \text{for } x_s - t \leq x - t < x_s, \\ \phi_0(x - t) & \text{for } x - t \geq x_s, \end{cases}$$

As in the first test case, we plot the density at time  $t^N$ , when half the mass has already crossed the interface between zones  $\Omega_1$  and  $\Omega_2$ . Two cases  $g_L = 0.5$  and  $g_L = 3$  are represented respectively on panels a) and b) of Figure 3.9. The convergence study is illustrated in panel b) of Figure 3.10. The scheme still works well with the source term, although the asymptotic order for the  $L_1$ -norm error is 2.4, which is slightly less accurate than without the source term. Since the source term is active

**Chapitre 3. A numerical method for transport equations with discontinuous flux functions : application to mathematical modeling of cell dynamics**

---



a)  $g_L = 0.5, t^N = 0.4$

b)  $g_L = 3.0, t^N = 0.0666$

FIGURE 3.9 – 1D test case 3 : flux continuity condition and velocity jump at interface  $x_s = 0.5$ , with source on the left. Snapshot of the density at initial time and when it crosses the interface between zones  $\Omega_1$  and  $\Omega_2$  ( $t = t^N$ ).

on the left subdomain, there is a loss in mass until the density bump reaches the interface (see panel a) of Figure 3.10). In the case where  $g_L = 3$ , the mass stabilizes after the density has passed through the interface. In the case where  $g_L = 0.5$  the density bump has not entirely passed through the interface at  $t = 0.25$  so that the mass is not yet stabilized.

**Test case 4. Doubling flux, speed discontinuity on  $\Omega_1$ - $\Omega_2$  interface, linear source**

The fourth test case is the same as the second one except that we add a linear source term  $\Lambda_R = 1$  in the right part of the domain. The exact solution is then

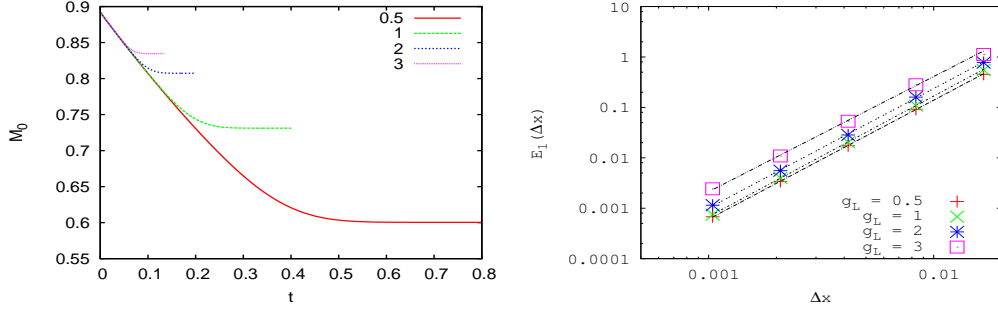
$$\phi^{exact}(x, t) = \begin{cases} \phi_0(x - t) & \text{for } x < x_s, \\ \frac{2}{g_R}(\phi_0(x_s - t - \frac{(x-x_s)}{g_R})) \exp(\frac{x_s-x}{g_R}) & \text{for } (x_s - g_R t) \leq (x - g_R t) < x_s, \\ \phi_0(x - g_R t) \exp(-t) & \text{for } (x - g_R t) \geq x_s, \end{cases}$$

As in the second test case, we plot the density at time  $t = 0.2$ , for two values  $g_R = 0.5$  and  $g_R = 3$  respectively on panels a) and b) of Figure 3.11. Here also the asymptotic order of convergence for the error is roughly 2.4 (see panel b) of Figure 3.12). The source term is active in the right subdomain. The mass, observed in panel a) of Figure 3.12, first doubles as the density bump goes through the interface, then diminishes as soon as the loss term becomes active. The drop in mass begins at the same time in both cases  $g_R = 0.5$  and  $g_R = 3$  since the interface is reached with the same left speed.

**3.4.2 2D test cases**

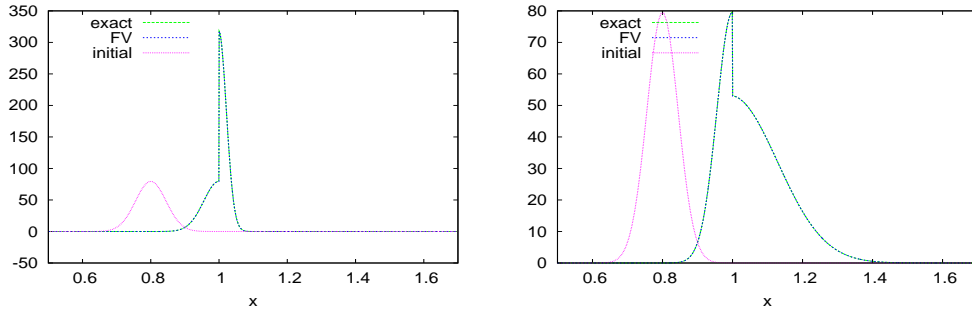
We now turn to the validation of the method in 2D. As shown on Figure 3.1, the computing domain includes three internal boundaries : one vertical between zones  $\Omega_1$  and  $\Omega_2$ , and two horizontal ones, between zones  $\Omega_1$  and  $\Omega_3$ , and  $\Omega_2$  and  $\Omega_3$ . The initial condition is centered on  $(c_x, c_y) = (0.3, 0.15)$ .

### Chapitre 3. A numerical method for transport equations with discontinuous flux functions : application to mathematical modeling of cell dynamics



a) mass with respect to time      b)  $L_1$  norm error

FIGURE 3.10 – 1D test case 3 : flux continuity condition and velocity jump at interface  $x_s = 0.5$ , linear source on the left. In panel a) the mass decreases with time before the density has passed the interface ( $CFL = 0.4, \Delta x = 0.001$ ). In panel b) the  $L_1$ -norm error goes to zero with  $\Delta x$ . The convergence rates (around 2.4) are displayed in Table 3.4.1.



a)  $g_R = 0.5$       b)  $g_R = 3$

FIGURE 3.11 – 1D test case 4 : doubling flux condition and velocity jump at interface  $x_s = 0.5$ , linear source on the right. Snapshot of the density  $\phi(x,t)$  at  $t = 0$  and  $t = 0.2$ , when it crosses the interface between zones  $\Omega_1$  and  $\Omega_2$ .

#### Error Computing

In 2D the relative error due to discretization is computed with

$$E(\Delta x) = \frac{\sum_{k=0}^{N_x-1} \sum_{l=0}^{N_y-1} |\phi_{k,l}^N - \bar{\phi}_{k,l}^N|}{\sum_{k=0}^{N_x-1} \sum_{l=0}^{N_y-1} |\bar{\phi}_{k,l}^N|} \quad (3.24)$$

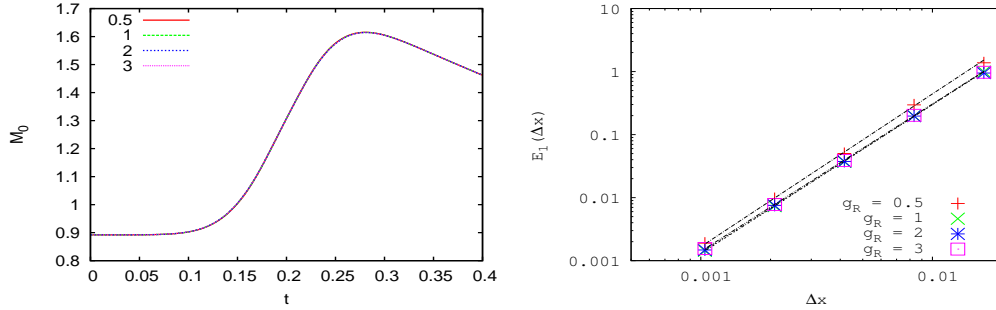
where  $\bar{\phi}_{k,l}^N$  is the mean value of the exact solution on the grid mesh  $[x_k, x_{k+1}] \times [y_l, y_{l+1}]$  at time  $t^N$ . This value is computed using the 2nd order quadrature formula.

#### 2D test case 1 : shear interface

For this test case the speed is vertical  $g(x,y) = 0$ ,  $h(x,y) = 1$  in the bottom part of the computing domain ( $\Omega_1 \cup \Omega_2$ ). In the top part ( $\Omega_3$ ), the speed is oblique

### Chapitre 3. A numerical method for transport equations with discontinuous flux functions : application to mathematical modeling of cell dynamics

---



a) mass with respect to time

b) L1 norm error

FIGURE 3.12 – 1D test case 4 : doubling flux condition and velocity jump at interface  $x_s = 0.5$ , linear source on the right. The mass increases just after the interface, then decreases when the density has passed the interface. The convergence rates (around 2.4) are displayed in Table 3.4.1.

$g(x, y) = 1$  and  $h(x, y) = 1$ . The discontinuity in speed is located on the horizontal internal boundary between  $\Omega_1$  and  $\Omega_3$  at  $y_s = 0.3$ . On the snapshots in Figure 3.13, we can observe a shear phenomenon that could not occur in 1D. We also notice that the change of direction of the speed transforms the shape of the density bump, from circular to elliptic. The total mass remains constant. Due to the shear, the asymptotic rate of convergence of the error drops to 2.1 (see panel a) of Figure 3.14)

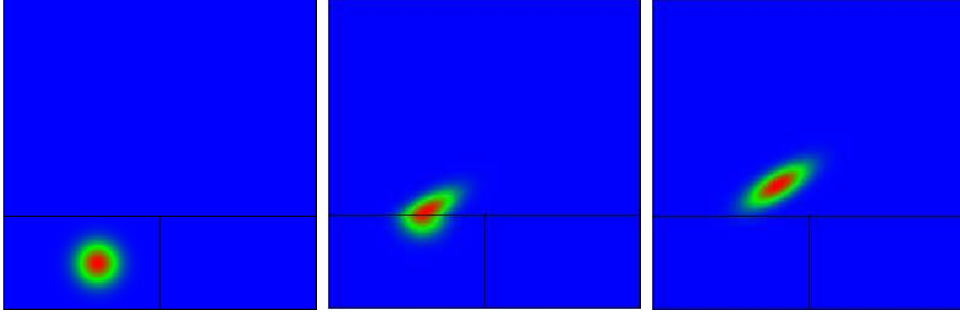
#### 2D test case 2 : waterproof interface

In this test case, the speed is diagonal  $g(x, y) = 1$ ,  $h(x, y) = 1$  in zone  $\Omega_1 \cup \Omega_3$ , and horizontal  $g(x, y) = 1$ ,  $h(x, y) = 0$  in zone  $\Omega_2$ . On the snapshots in Figure 3.15, we can observe a phenomenon of waterproof interface that could not occur in 1D. The fraction of the mass which crosses the vertical interface between zones  $\Omega_1$  and  $\Omega_2$  remains trapped in zone  $\Omega_2$  and can only move horizontally. The convergence rate drops to 1 (panel b) of Figure 3.14). In order to understand better the drop in precision for the 2D test case 2, we have studied other values for  $h(x, y) = h_B$  in zone  $\Omega_2$ , ranging from the reference case with no speed variation ( $h_B = 1$ ) to the waterproof test case ( $h_B = 0$ ). Figure 3.16 displays the density at time  $t = 0.25$  for three different  $h_B$  values. For  $h_B = 0.9$ , the solution displayed on panel a) looks very much like that obtained with the reference case. For  $h_B = 0.001$ , the solution displayed in panel c) looks very much like that obtained with the waterproof boundary in Figure 3.15.

Since the exact solution is not available except for the extreme values  $h_B = 0$  or 1, we compute the error at time  $t^N$  using in (4.35), instead of the exact solution, a reference solution  $\phi^{fine}$  computed on a very fine grid with  $N_x^{fine}$  grid meshes, and

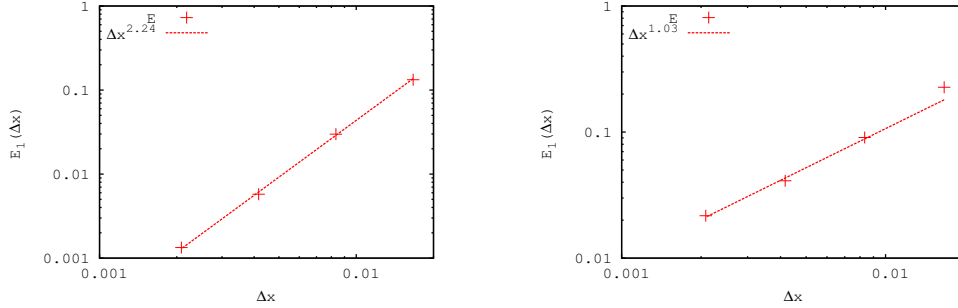
**Chapitre 3. A numerical method for transport equations with discontinuous flux functions : application to mathematical modeling of cell dynamics**

---



a) Initial time. The density bump lies in zone  $\Omega_1$ .  
 b) Intermediate time. Density crossing interface at  $t = 0.15$ .  
 c) Final time  $t = 0.25$ . The density bump lies in zone  $\Omega_3$ .

FIGURE 3.13 – *First 2D test case, shear phenomenon. Snapshot of the density (initial time, passing interface, final time). (CFL = 0.4,  $\Delta x = 0.001$ ).*



a) 2D test case 1 : shear.                      b) 2D test case 2 : waterproof.

FIGURE 3.14 – 2D test cases.  $L_1$ -norm relative error with respect to  $\Delta x$ . The convergence rate is 2.24 in the shear case (panel a) and drops to 1.03 in the waterproof case (panel b).

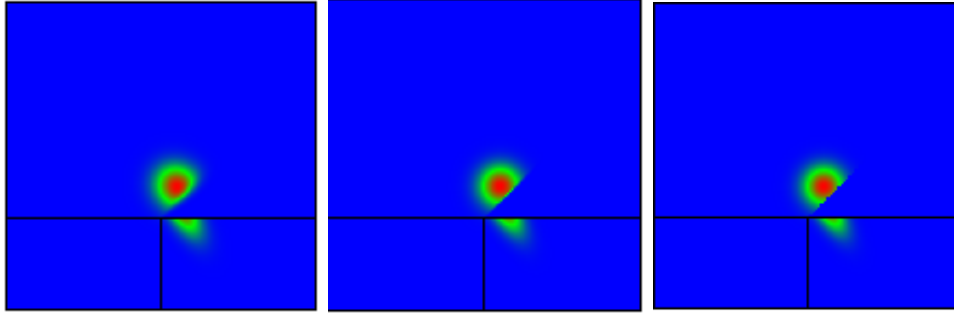
projected on the current grid with  $N_x^{coarse}$  grid meshes :

$$\bar{\phi}_{k,l}^N = \sum_{i=1}^m \sum_{j=1}^m \phi_{km+i,lm+j}^{fine,N} \quad (3.25)$$

with

$$m = \frac{N_x^{fine}}{N_x^{coarse}}.$$

Here the space step for the reference solution is  $\Delta x = 0.0005$ , corresponding to  $N_x^{fine} = 2000$ . The different error curves and the variation of the error order with respect to  $h_B$  are displayed respectively on panel a) and b) of Figure 3.17. This numerical experiment confirms that there is a regular drop in precision as the contrast between the vertical speeds on both sides of the interface between zones  $\Omega_2$  and



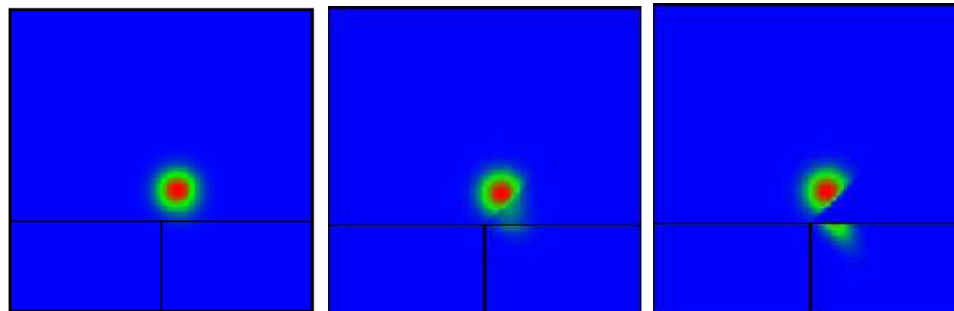
a) First order approximation.

b) Third order approximation.

c) Exact solution.

FIGURE 3.15 – 2D test case 2 : waterproof interface. Density at final time (First and third order computations and exact solution). Some numerical diffusion affects the shape of the density, due to the splitting of the initial density bump into two separate clouds.

$\Omega_3$  increases. It is worth noting that the asymptotic order for the waterproof case ( $h_B = 0$ ) computed using the converged solution is 1.25 instead of the value 1 obtained when using the exact solution in formula (4.35). On the other hand, the asymptotic order for the case with constant diagonal speed ( $h_B = 1$ ) is 2.25, which is less than the asymptotic order 2.4 obtained in the 1D test case 1 for  $g_L = g_R = 1$ . This small drop in precision is generally encountered with 2D schemes on cartesian grids obtained by tensorization of a 1D scheme.



a)  $h_B = 0.9$

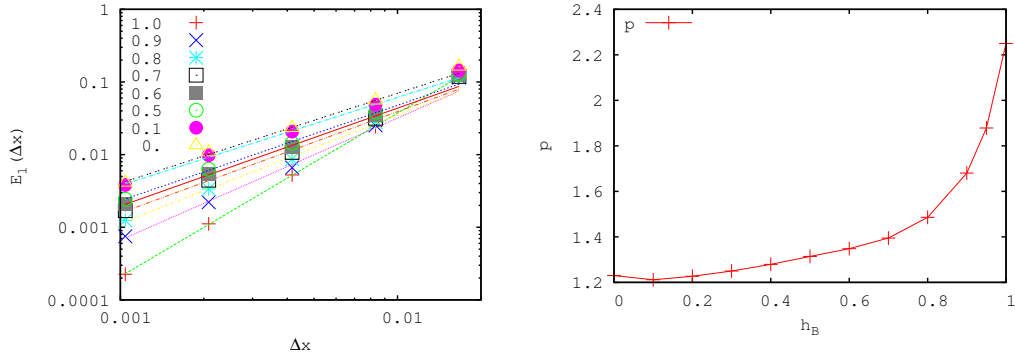
b)  $h_B = 0.5$

c)  $h_B = 0.001$

FIGURE 3.16 – From constant diagonal speed to waterproof interface. Solution at time  $t = 0.25$  for three different values of the vertical speed in the  $\Omega_2$  zone. a)  $h_B = 0.9$ , b)  $h_B = 0.5$ , c)  $h_B = 0.001$

**Chapitre 3. A numerical method for transport equations with discontinuous flux functions : application to mathematical modeling of cell dynamics**

---



a) Relative error variation.

b) Order  $p$  as function of  $h_B$ .

FIGURE 3.17 – From constant diagonal speed to waterproof interface. a) Relative error and best least-square fit  $O(\Delta x^p)$  for different constant  $h_B$  values of the vertical speed in zone  $\Omega_2$ . b) Asymptotic order  $p$  as a function of  $h_B$  value in zone  $\Omega_2$ .

### 3.5 Application to cell dynamics

In this last section we turn to the numerical simulation of a biological model dedicated to the cell dynamics in ovarian follicular development described in Echenim *et al.* [41] and references therein. This multiscale model describes the latest stages of the follicular development before ovulation. The cell population of each follicle is studied as a density function depending on time, and structured according to two functional space variables, age and maturity. The density functions of the interacting follicles are solutions of weakly coupled PDE-s, where the age and maturity velocities and the loss term depend on the integro-moments of the solution for all the follicles. The moment-based integral formulation accounts in a compact way for the feedback loop involving the ovarian hormones, secreted from follicular cells, and the pituitary hormone FSH (follicle-stimulating hormone) that targets in turn the follicular cells.

The model follows the development of the follicles starting from an initial stage where all cells are proliferating. As the granulosa cells progress through subsequent cell cycles their maturities increase up to a threshold beyond which they exit the proliferation cell cycle and enter the differentiation stage. The control exerted by FSH on the cells depends on the cell state, that is on whether the cell be within or outside the division cycle. Moreover, cells become insensitive to any external control during a part of the cycle, leading to a drift dynamics behaving as pure transport. Cell proliferation is underlain by the process of mitosis, through which a mother cell gives birth to two daughter cells. Mitosis is the endpoint of the cell cycle that consists of the 4 phases G1, S, G2 and M, and ensures proper DNA replication and repartition between the new cells. The most common way of representing mitosis in age-structured models of cell populations is to add a gain term in the right hand side, that controls the (average) doubling time in the cell population (see the renewal equations presented in Chapter 3 of [64]). This is however a rough description of the

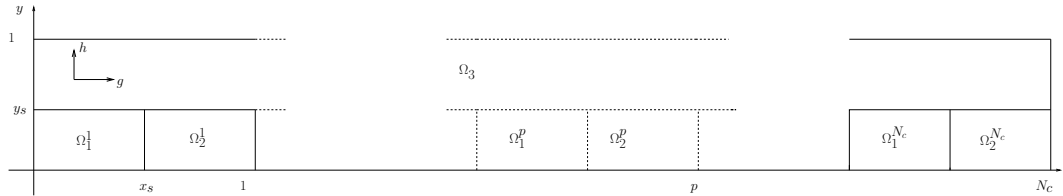


### Chapitre 3. A numerical method for transport equations with discontinuous flux functions : application to mathematical modeling of cell dynamics

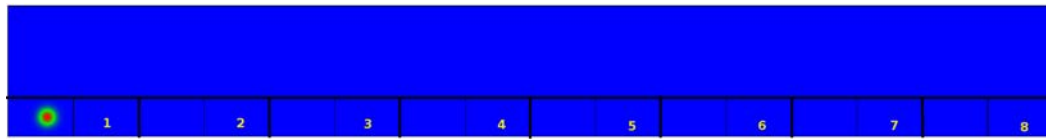
---

mitosis event since it is distributed over all cell ages. An alternative way to represent mitosis is to consider appropriate boundary conditions coupling the dynamics of the population of proliferating cells with another population of cells that have exited the cell cycles. Instances of corresponding models can be found in the context of hematopoiesis, the process by which blood cells are produced (see e.g. [2]).

Here, we consider a model where one cannot get rid of discontinuity problems, since not only the mitosis event, but also the distinction between different phases of the cell cycle are embedded in the cell population dynamics. Due to the phase-dependent sensitivity of cells to the extracellular signals that make them progress along the cell cycle, discontinuities on the velocities have to be dealt with in addition to the mitosis-induced discontinuity. More precisely, we account for both the `START` transition from phase G1 to S and the `EXIT` transition after mitosis completion [77]. Besides the mitosis process which increases the total cell mass, the follicular development model takes into account apoptosis, and therefore total mass loss, via a source term which is active only locally in a narrow zone delimited by a skewed gaussian law and centered on the boundary between the first phase G1 of the cycle and the differentiation phase. The coefficients of the cell loss term depend on the first moment of the solution.



a) Computational domain. The cell cycles  $\Omega_1^p \cup \Omega_2^p$ ,  $p = 1, \dots, N_c$  of unit width are periodically reproduced.  $\Omega_1^p$  corresponds to phase G1, while  $\Omega_2^p$  aggregates phases S, G2 and M of the  $p$ th cell cycle.



b) Initial condition, identical for both follicles. The simulation covers  $N_c = 8$  cell cycles. Each cell cycle is divided into  $\Omega_1^p \cup \Omega_2^p$ , for  $p = 1, \dots, 8$ . The cell population is initially in  $\Omega_1^1$ , in the first cell cycle. A color code is used for the cell density ; a null density is encoded in blue, while the highest value is encoded in red.  $\phi_{max} = 75$  denotes the maximum observed density over the whole domain at  $t = 0$ .

FIGURE 3.18 – *Initial condition and computational domain for the full model*

An additional difficulty compared to the toy models is the weak coupling between different follicles through the generalized moments of the cell density. The HPC issues in the case of realistic numbers of follicles -around 20- have already been addressed in [6]. Here we will first focus on the case of one follicle to study the precision of the scheme, and then illustrate the vector case with a competition between two follicles. Let us denote by  $F$  the number of follicles in the general case. The density



**Chapitre 3. A numerical method for transport equations with discontinuous flux functions : application to mathematical modeling of cell dynamics**

---

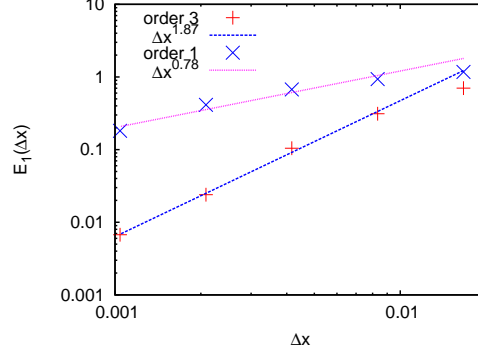


FIGURE 3.19 – Follicular development model. Symbols  $\times$  and  $+$  indicate the numerical errors respectively obtained with the order 1 and order 3 schemes. Dotted lines indicate the corresponding numerical convergence rates (0.78 and 1.87) obtained by least-square fit.

plasma FSH level and the locally bioavailable FSH level and depend on the first maturity moment of the densities

$$m_f(t) = \int_0^1 \int_0^{N_c D_c} y \phi_f(x, y, t) dx dy. \quad (3.30)$$

$$M(t) = \sum_{f=1}^F m_f(t) \quad (3.31)$$

Note that in the case of one single follicle  $m_f(t)$  and  $M(t)$  are identical. The plasma FSH level  $U(t)$  showing up in the arguments of the source term in (4.38) is defined by

$$U(t) = U_{\min} + \frac{1 - U_{\min}}{1 + \exp(c(M(t) - \bar{M}))}, \quad (3.32)$$

where  $U_{\min}$ ,  $c$  and  $\bar{M}$  are real positive constants. The locally bioavailable FSH level  $u_f(t)$  showing up in the arguments of the speeds in (4.38) is defined by

$$u_f(t) = \min \left( b_1 + \frac{\exp(b_2 m_f(t))}{b_3}, 1 \right) U(t), \quad \text{for } f = 1, \dots, F \quad (3.33)$$

where  $b_1$ ,  $b_2$  and  $b_3$  are real positive constants that may depend on the follicle  $f$ . The numerical value of the biological constants appearing in definitions 5.2, 4.40 and 4.44 are gathered in Table 3.4.1. To simplify the notation we do not explicitly indicate their possible dependence as a function of the follicle even if it is implied by the  $f$  index notation.

The precise definition of the required transmission conditions along the successive cell cycles of the domain has been formulated in [71]. For each cycle  $p = 1, \dots, N_c$ , – the flux on the  $x$ -axis is continuous on the interface between  $\Omega_1^p$  and  $\Omega_2^p$

$$\phi_f(t, x^+, y) = (\gamma_1 u_f + \gamma_2) \phi_f(t, x^-, y), \quad x = (p - 1/2)D_c, \quad 0 \leq y \leq y_s. \quad (3.34)$$

**Chapitre 3. A numerical method for transport equations with discontinuous flux functions : application to mathematical modeling of cell dynamics**

---

Parameter	Description	Value				
$U_{min}$ $c$ $\bar{M}$	FSH plasma level (eq. (5.1))					
	minimum level	0.5				
	slope parameter	2.0				
	abscissa of the inflection point	4.5				
$\bar{\Lambda}$ $\bar{\gamma}$	Apoptosis source term (eq. (4.41))					
	intensity factor	0.1				
	scaling factor	0.01				
$y_s$	cellular maturity threshold	0.3				
$b_1$ $b_2$ $b_3$	intrafollicular FSH level (eq. (4.44))					
	basal level	0.08				
	exponential rate	2.25				
	scaling factor	1450.				
$\gamma_1$ $\gamma_2$	Aging function (eq. (5.2))					
	rate	2.				
	origin	2.				
$c_1$ $c_2$ $\bar{u}$	Maturation function (eq. (4.40))	0.68				
		0.08				
		0.02				
$\tau_h$		<table border="1" style="width: 100%; border-collapse: collapse;"> <tr> <td style="width: 50%; text-align: center;">foll. 1</td> <td style="width: 50%; text-align: center;">foll. 2</td> </tr> <tr> <td style="text-align: center;">0.7</td> <td style="text-align: center;">1.0</td> </tr> </table>	foll. 1	foll. 2	0.7	1.0
	foll. 1	foll. 2				
0.7	1.0					

TABLE 3.2 – Values of the parameters for the biological model simulation. The two follicles in competition differ by the value of  $\tau_h$

- The flux is doubling on the interface between  $\Omega_2^p$  and  $\Omega_1^{p+1}$ , which accounts for the birth of two daughter cells from one mother cell at the end of each cell cycle

$$(\gamma_1 u_f + \gamma_2) \phi_f(t, x^+, y) = 2 \phi_f(t, x^-, y), \quad x = pD_c, \quad 0 \leq y \leq y_s. \quad (3.35)$$

- A homogeneous Dirichlet condition holds to the north of the interface between  $\Omega_2^p$  and  $\Omega_3$

$$\phi_f(t, x, y_s^+) = 0, \quad (p - 1/2)D_c \leq x \leq pD_c. \quad (3.36)$$

We refer the reader to [6] and the references therein for more details on the model, but nevertheless stress out that, as mentioned in the introduction, an important feature, which is inherent to cell dynamics, is the mitosis event occurring at the end of each cell cycle, materialized in our setup by the interfaces between phases  $\Omega_2^p$  and  $\Omega_1^{p+1}$ . In most transport equations dedicated to cell dynamics, it is not accounted for as such but rather by a gain term distributed over the cell population [64]. In [71], the model with discontinuous coefficients was shown to have a unique weak solution, provided that the discontinuities in the velocities were taken into account by flux continuity – at the interfaces between phases  $\Omega_1^p$  and  $\Omega_2^p$  – and flux doubling – at the end of each cycle. The  $\Omega_2$  phase being a transport in age direction only, null flux at the interface with the differentiating phase  $\Omega_3$  ensures the well posedness.

### 3.5.1 Numerical simulation

We first perform a convergence study on the full model. For this time consuming validation, we restrict ourselves to only one follicle  $F = 1$ , and we stop the simulation after a duration  $t^N = 1$ , which is short enough so that the computations on very fine grids remain tractable and would correspond to the transit time across the first cell cycle if there were no control of the ageing velocity (i.e. if the cell age evolved as time in  $\Omega_1^1$ ). Even if, for the chosen values of the biological parameters, gathered in Table 4.2, the first cell cycle duration is slightly less than one, all of the three interesting transitions :  $\Omega_1^1 - \Omega_2^1$ ,  $\Omega_2^1 - \Omega_1^2$  and  $\Omega_1^2 - \Omega_3$  do happen, as well as the waterproof phenomenon between zones  $\Omega_2^1 - \Omega_3$ . The relative error is computed with respect to a converged solution obtained using a very fine grid of 1920 grid meshes per half cycle (see equation 3.25). It is displayed as a function of the space step  $\Delta x$  on Figure 3.19. The asymptotic order of convergence computed by least square fitting is almost 2. The biological model combines all the numerical difficulties studied separately in the six previous test cases. As expected, the order of convergence is intermediate between the best and worst order achieved for these toy problems. Finally we illustrate the competition between two follicles. The initial condition, with all cells in phase  $\Omega_1^1$ , is represented in the bottom panel of Figure 4.12. To illustrate the different phenomena at interfaces, snapshots of the density at significant times of the follicular development are displayed on Figures 3.20 and 3.21. For this simulation, a space discretization  $\Delta x = 0.0125$  (80 grid meshes per half cycle) is used, leading to a varying time discretization of  $\Delta t \approx 0.0012$  which meets condition (3.17) with  $CFL = 0.4$ . The snapshots illustrate two important aspects of the model. First, the transitions between adjacent phases are well captured using the numerical method described in the paper. Second, the differences between two competing follicles whose biological properties differ only by the value of one parameter, the maturation velocity, are shown to increase with time. At initial time the two follicles have the same cell population  $m_0^1 = m_0^2 = 1$ , and in the final snapshot at  $t = 6.30$  (panel b) of Figure 3.21, which corresponds to a proliferation process covering seven cell cycles, their cell populations are very different in term of cell mass and cell repartition within the domain. Follicle 1 (top), which is 70% slower in the maturity axis ( $y$ ) than follicle 2 (bottom), is still evolving towards the asymptotic maturity level and its density is still split into distinct clouds corresponding to cells having entered the differentiation phase during different cell cycles, with a total mass  $m_0^1 = 9.3$ . At the same time, follicle 2 has almost reached its asymptotic state with a final total mass of only  $m_0^2 = 4.3$ . The difference in mass can be explained by the fact that being slower in the  $y$  direction, follicle 1 spends more time in the proliferation phase than follicle 2.

## Conclusion

The numerical method presented in Sections 2 and 3 and validated by the numerical convergence study in Section 4 is applied to a biological model for follicular

development. Thanks to the original treatment of interface conditions it shows very promising results in terms of accuracy. This numerical tool will enable us to calibrate the model with respect to quantitative biological specifications. More extensive simulations with realistic numbers of follicles in competition will be performed to better understand the phenomena under study, like for instance multiple ovulations.

### 3.6 Appendix : Exact solutions

We now give the details of the exact solution computations for the 1D and 2D test cases. In all cases, the solutions will exhibit well defined traces on the left and right hand sides of the discontinuity. Those limits will be explicit in the solution.

#### 3.6.1 Exact solutions of the 1D problem with piecewise constant speeds

##### Problem

$$\left\{ \begin{array}{l} \partial_t \phi + \partial_x g \phi = \Lambda \phi \quad \text{on } \mathbb{R}^+ \times (-\infty, x_s[ \cup ]x_s, +\infty) \\ g(x) = \begin{cases} g_L & \text{if } x < x_s \\ g_R & \text{if } x \geq x_s \end{cases} \quad (\text{discontinuous speed}) \\ \Lambda(x) = \begin{cases} \Lambda_L & \text{if } x < x_s \\ \Lambda_R & \text{if } x \geq x_s \end{cases} \quad (\text{discontinuous speed}) \\ \phi(x, 0) = \phi_0(x) \quad (\text{initial condition}) \\ \psi_L(g_L \phi(x_s^-, t)) = g_R \phi(x_s^+, t) \quad (\text{flux condition}) \end{array} \right. \quad (3.37)$$

**Case without source term** We first solve this problem when  $\Lambda_L = \Lambda_R = 0$ , using the method of characteristics. The case when  $g_R < g_L$  is displayed in Figure 3.22. If there were no transmission conditions, the characteristics would be straight lines passing through the vertical axis. The analysis is detailed in the case where  $g_L$  and  $g_R$  speeds are positive which is the most usual situation in problems arising from biology.

**Solution** For  $x < x_s$ , considering the characteristics of

$$\left\{ \begin{array}{l} \partial_t \phi + g_L \partial_x \phi = 0, \\ \phi(x, 0) = \phi_0(x), \end{array} \right.$$

the solution is  $\phi(x, t) = \phi_0(x - g_L t)$  for  $x < x_s$ . Using the same argument, considering the characteristics of

$$\left\{ \begin{array}{l} \partial_t \phi + g_R \partial_x \phi = 0, \\ \phi(x, 0) = \phi_0(x), \end{array} \right.$$

we find that  $\phi(x, t) = \phi_0(x - g_R t)$  for  $x - g_R t > x_s$ . The solution for  $x_s - g_R t < x - g_R t < x_s$  is obtained thanks to the transmission condition  $\psi_L(g_L \phi(x_s^-, t)) =$

**Chapitre 3. A numerical method for transport equations with discontinuous flux functions : application to mathematical modeling of cell dynamics**

---

$g_R\phi(x_s^+, t)$ , combined with the fact that  $\phi(x_s^-, t) = \phi_0(x_s - g_L t)$ , which leads to  $g_R\phi(x_s^+, t) = \psi_L(g_L\phi(x_s^-, t)) = \psi_L(g_L\phi_0(x_s - g_L t))$ . Defining a trace function

$$Tr(t) = \frac{1}{g_R}\psi_L(g_L\phi_0(x_s - g_L t)), \quad (3.38)$$

which acts as a boundary condition, we follow the characteristics

$$\begin{cases} \dot{x}(t) = g_R, \\ x(t_s) = x_s, \end{cases}$$

given by  $x(t) = x_s + (t - t_s)g_R$ , to reach  $x_s$  at time

$$t_s = t - \frac{(x - x_s)}{g_R}. \quad (3.39)$$

Finally, the solution is

$$\phi(x, t) = \begin{cases} \phi_0(x - g_L t) & \text{for } x < x_s, \\ Tr(t_s) & \text{for } x_s - g_R t < x - g_R t < x_s, \\ \phi_0(x - g_R t) & \text{for } x - g_R t < x_s, \end{cases}$$

where  $Tr$  is defined by condition (4.27) and  $t_s$  is defined by (4.28).

**Case with source term** It can be deduced from the homogeneous case by changing the unknowns. In the subregion where  $\Lambda$  is constant, if  $\phi(x, t)$  is solution of  $\partial_t\phi + \partial_x g\phi = -\Lambda\phi$  then  $\tilde{\phi}(x, t) = \exp(\Lambda t)\phi(x, t)$  is solution of  $\partial_t\tilde{\phi} + \partial_x g\tilde{\phi} = 0$ . We distinguish two cases :

With a **source term in the left subdomain**  $\Lambda_L > 0$  and  $\Lambda_R = 0$ , the solution is

$$\phi(x, t) = \begin{cases} \phi_0(x - g_L t) \exp(-\Lambda t) & \text{for } x < x_s, \\ Tr(t_s) \exp(-\Lambda t_s) & \text{for } x_s - g_R t < x - g_R t < x_s, \\ \phi_0(x - g_R t) & \text{for } x - g_R t > x_s. \end{cases}$$

where  $Tr$  is defined by the transmission condition (4.27) and  $t_s$  is defined by (4.28).

With a **source term in the right subdomain**  $\Lambda_L = 0$  and  $\Lambda_R > 0$ , the solution is

$$\phi(x, t) = \begin{cases} \phi_0(x - g_L t) & \text{for } x < x_s, \\ Tr(t_s) \exp(-\Lambda(t - t_s)) & \text{for } x_s - g_R t < x - g_R t < x_s, \\ \phi_0(x - g_R t) \exp(-\Lambda t) & \text{for } x - g_R t > x_s. \end{cases}$$

where  $Tr$  is defined by the transmission condition (4.27) and  $t_s$  is defined by (4.28).

### 3.6.2 Exact solution of the 2D problems with piecewise constant speeds

#### 3.6.2.1 Horizontal speed

##### Problem

$$\begin{cases} \partial_t \phi + \partial_x g \phi = 0 & \text{on } \mathbb{R}^+ \times (-\infty, x_s[ \cup ] x_s, +\infty) \times \mathbb{R} \\ g(x, y) = \begin{cases} g_L & \text{if } x < x_s \\ g_R & \text{if } x > x_s \end{cases} & \text{(discontinuous speed)} \\ \phi(x, y, 0) = \phi_0(x, y) & \text{(initial condition)} \\ \psi_L(\phi(x_s^-, y, t)) = \phi(x_s^+, y, t) & \text{(flux condition)} \end{cases} \quad (3.40)$$

**Solution** The solution is

$$\phi(x, y, t) = \begin{cases} \phi_0(x - g_L t, y) & \text{for } x < x_s, \\ Tr(t_s, y) & \text{for } x_s - g_R t < x - g_R t < x_s, \\ \phi_0(x - g_R t, y) & \text{for } x - g_R t > x_s, \end{cases}$$

where  $Tr$  is defined by the transmission condition

$$Tr(t, y) = \psi_L(\phi_0(x_s - g_L t, y)). \quad (3.41)$$

and  $t_s$  is defined by (4.28).

#### 3.6.2.2 Shear

##### Problem

$$\begin{cases} \partial_t \phi + \partial_y \phi = 0 & \text{on } \mathbb{R}^+ \times \mathbb{R} \times (-\infty, y_s[ \\ \partial_t \phi + \partial_x \phi + \partial_y \phi = 0 & \text{on } \mathbb{R}^+ \times \mathbb{R} \times ] y_s, +\infty) \\ \phi(x, y, 0) = \phi_0(x, y) & \text{(initial condition)} \end{cases} \quad (3.42)$$

**Solution** The characteristics are of the form

$$\begin{cases} x(t) = x_0 + t^D \\ y(t) = y_0 + t_s + t^D \end{cases}$$

with  $t^D = y - y_s$  the time spent in the upper zone. We can then define the solution piecewise

$$\phi(x, y, t) = \begin{cases} \phi_0(x, y - t) & \text{for } (x, y) \in \mathbb{R} \times (-\infty, y_s[ \\ \phi_0(x - t^D, y - t) & \text{for } (x, y) \in \mathbb{R} \times ] y_s, +\infty) \end{cases}$$



### 3.6.2.3 Waterproof

#### Problem

$$\begin{cases} \partial_t \phi + \partial_x \phi = 0 & \text{on } \mathbb{R}^+ \times ]x_s, +\infty) \times (-\infty, y_s[ \\ \partial_t \phi + \partial_x \phi + \partial_y \phi = 0 & \text{on } \mathbb{R}^+ \times (\infty, x_s[ \times (-\infty, y_s[ \text{ and } \mathbb{R}^+ \times \mathbb{R} \times ]y_s, +\infty) \\ \phi(x, y, 0) = \phi_0(x, y) & \text{(initial condition)} \end{cases} \quad (3.43)$$

In order to close this problem, we have to add a homogeneous Dirichlet condition on the north of the limit between the 2 zones

$$\phi(x, y_s^+, t) = 0 \quad \text{for } x \in ]x_s, +\infty)$$

**Solution** As in the precedent case the characteristics are of the form

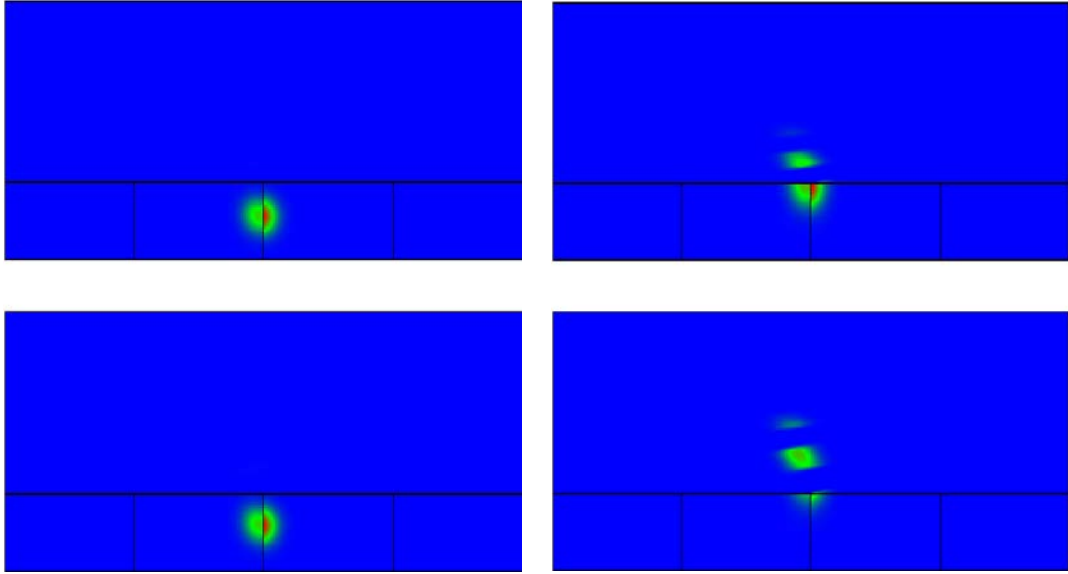
$$\begin{cases} x(t) = x_0 + t^D, \\ y(t) = y_0 + t_s + t^D. \end{cases}$$

We can then define the solution piecewise

$$\phi(x, y, t) = \begin{cases} \phi_0(x - t, y - t) & \text{for } (x, y) \in \mathbb{R} \times ]y_s, +\infty) \text{ and } x - t^D < x_s \\ 0 & \text{for } (x, y) \in \mathbb{R} \times ]y_s, +\infty) \text{ and } x - t^D > x_s \\ \phi_0(x - t, y - t) & \text{for } (x, y) \in \Omega_1 \\ \phi_0(x - t, y - t_s) & \text{for } (x, y) \in \Omega_2 \end{cases}$$

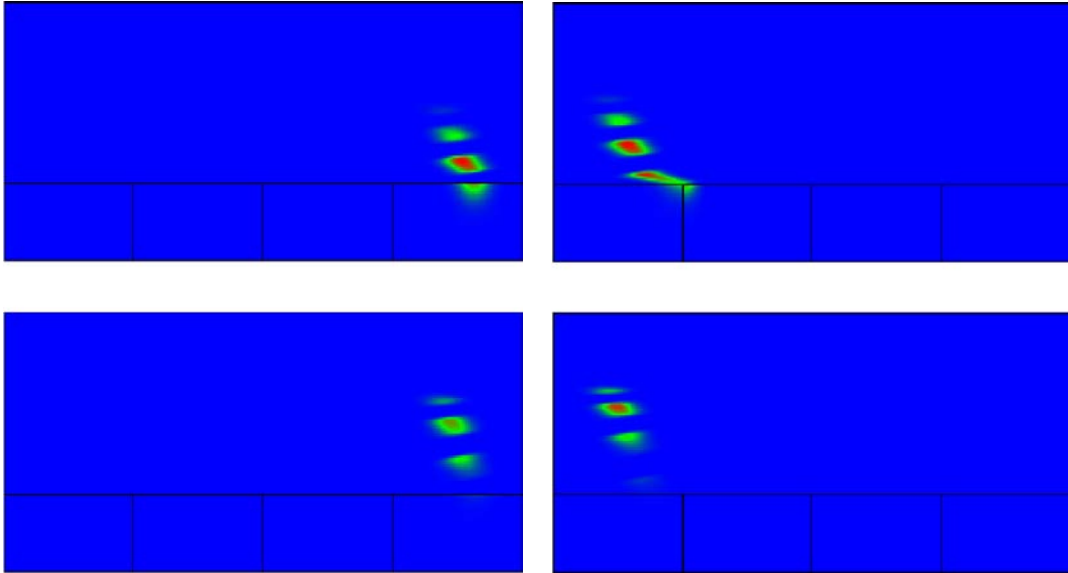
Chapitre 3. A numerical method for transport equations with discontinuous flux functions : application to mathematical modeling of cell dynamics

---



a)  $t = 0.66$ , cell cycles 1 – 2  
 transition  $\Omega_2^1 \rightarrow \Omega_1^2$ .  
 $\phi_{max} = 140$

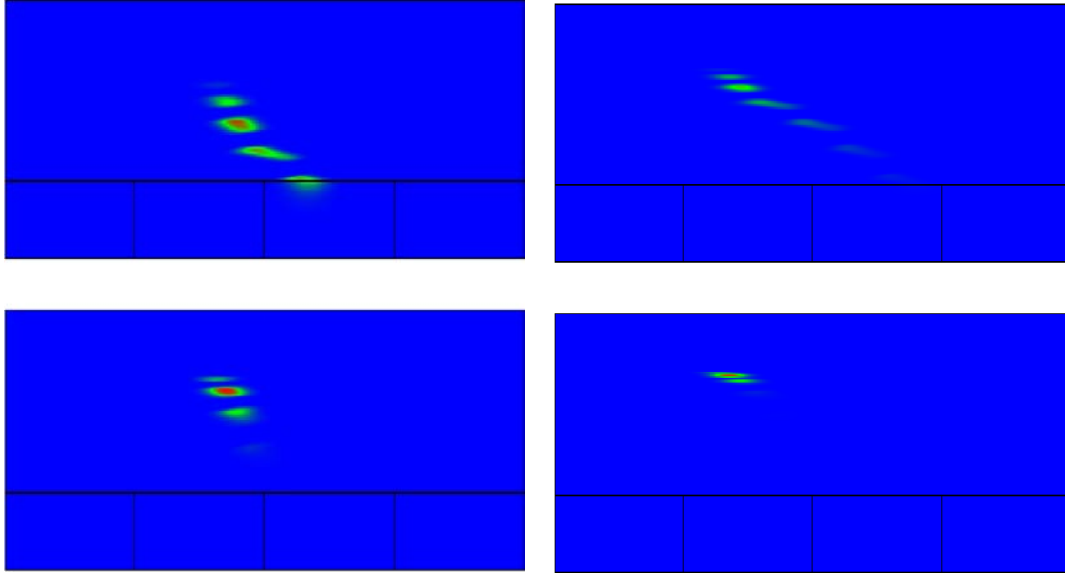
b)  $t = 2.58$ , cell cycles 3 – 4  
 transition  $\Omega_2^3 \rightarrow \Omega_1^4$  and  $\Omega_1^4 \nearrow \Omega_3$ ,  
 $\phi_{max} = 430$



c)  $t = 3.35$ , cell cycles 3 – 4,  
 waterproof boundary between  $\Omega_2^4$   
 and  $\Omega_3$  in the 4th cell cycle for  
 follicle 1,  $\phi_{max} = 495$

d)  $t = 3.8$ , cell cycles 5 – 6,  
 differentiation in  $\Omega_3$ , residual mass in  
 cycle  $\Omega_2^5$  for follicle 1,  
 $\phi_{max} = 510$

FIGURE 3.20 – Full model : snapshots of the density at different times of the follicular development,  $\Delta x = 0.0125$ . The color code is time related. Blue and red colors indicate respectively null and maximum observed density  $\phi_{max}$  over the whole domain at the snapshot times. Follicle 1 (top) and follicle 2 (bottom) differ by  $\tau_h = 0.7$  and 1.0 respectively



a)  $t = 4.48$ , cell cycles 5 – 6, differentiation in  $\Omega_3$ , residual mass in the 6th cycle  $\Omega_1^6$  for follicle 1,  $\phi_{max} = 660$

b)  $t = 6.30$ , cell cycles 7 – 8, differentiation in  $\Omega_3$ ,  $\phi_{max} = 1600$

FIGURE 3.21 – Full model : snapshots of the density at different times of the follicular development,  $\Delta x = 0.0125$ . The color code is time related. Blue and red colors indicate respectively null and maximum observed density  $\phi_{max}$  over the whole domain at the snapshot times. Follicle 1 (top) and follicle 2 (bottom) differ by  $\tau_h = 0.7$  and 1.0 respectively

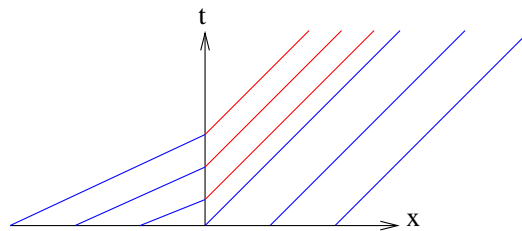


FIGURE 3.22 – Characteristics of a transmission problem with a transmission condition. The solution is a stationary shock, corresponding to  $g_R < g_L \psi'_L$ .

# Adaptive mesh refinement strategy for a non conservative transport problem

---

Ce chapitre a fait l'objet d'une publication dans la revue M2AN [8].

## Sommaire

---

<b>4.1 Introduction . . . . .</b>	<b>82</b>
<b>4.2 Investigation of numerical noise appearance . . . . .</b>	<b>84</b>
4.2.1 Description of the simplified set-up . . . . .	84
4.2.2 Finite volume discretization on a two-level MR hierarchy . . .	85
4.2.3 Explication of the numerical noise appearance . . . . .	88
4.2.4 Adaptive doubling strategy . . . . .	91
4.2.5 Formalization and generalization of the doubling strategy . .	92
<b>4.3 Numerical validation of the adaptive thresholds strategies .</b>	<b>94</b>
4.3.1 Effect of the adaptive doubling strategy on the solution . . .	94
4.3.2 Time adaptation of the MR threshold $\varepsilon_t$ . . . . .	99
4.3.3 Link between the MR and the doubling thresholds . . . . .	100
<b>4.4 Application to a biological model . . . . .</b>	<b>102</b>
4.4.1 Model and numerical method . . . . .	102
4.4.2 Numerical performances . . . . .	106
4.4.3 Comparison of the numerical performances for both models of mitosis . . . . .	110
<b>4.5 Appendix : Description of the model for follicular develop- ment . . . . .</b>	<b>112</b>
<b>4.6 Appendix : Example of spurious numerical noise appearance</b>	<b>114</b>
<b>4.7 Numerical noise appearance . . . . .</b>	<b>114</b>

---

**abstract** Long time simulations of transport equations raise computational challenges since they require both a large domain of calculation and sufficient accuracy. It is therefore advantageous, in terms of computational costs, to use a time varying adaptive mesh, with small cells in the region of interest and coarser cells where the solution is smooth. Biological models involving cell dynamics fall for instance within this framework and are often non conservative to account for cell division. In that

case the threshold controlling the spatial adaptivity may have to be time-dependent in order to keep up with the progression of the solution. In this article we tackle the difficulties arising when applying a Multiresolution method to a transport equation with discontinuous fluxes modeling localized mitosis. The analysis of the numerical method is performed on a simplified model and numerical scheme. An original threshold strategy is proposed and validated thanks to extensive numerical tests. It is then applied to a biological model in both cases of distributed and localized mitosis.

### 4.1 Introduction

When studying phenomena over long periods in the context of transport equations, one has to make simulations on large computational domains so as to follow the interesting part of the solution throughout the duration of the experiment. On the other hand it can happen that the solutions of the problem occupy only a bounded and time varying portion of the domain. Such set-ups can be very costly to simulate numerically with uniform grids. Time-adaptive mesh refinement methods enable one to minimize the size of the mesh, hence both the memory requirement and CPU time, by restricting the use of fine discretization to the areas of interest, or more generally to the regions where the solution varies rapidly.

Among these methods, the Multiresolution (MR) method is specially interesting for hyperbolic systems of PDEs. Firstly, the underlying wavelet theory enables one to control the numerical error induced by the adaptive scheme with a small parameter  $\varepsilon$  that is explicitly used as a threshold in the algorithm. Secondly, MR provides one with a strategy to monitor the evolution in time of the adaptive grid, relying on the hyperbolicity of the equations to predict the displacement of the singularities. We refer the reader to [30, 62] for a thorough analysis of the MR method applied to a scalar hyperbolic equation and to [31, 34] and references therein for applications to more realistic models and set-ups. Although originally designed for and theoretically studied in the case of hyperbolic equations, Multiresolution is currently used to simulate reaction-diffusion and parabolic equations with interesting computational performances, as illustrated in [33, 56].

In the current work we deal with non conservative transport equations frequently arising in biological models involving cell dynamics. In these models, which are most of the time 1D, the space variable plays the role of age and marks the progression along the cell cycle. Mitosis is the endpoint of the cell cycle, that gives birth to two daughter cells from one mother cell. From the modeling viewpoint, mitosis can be distributed over the cell cycle thanks to a positive linear source term, amounting to doubling the cell density in a time equivalent to the duration of the cell cycle (see for instance [64]). Another approach treats the mitosis phenomenon more directly, as a localized event at the end of each cell cycle, corresponding to a doubling transmission condition which must be satisfied by the cell density at the interface between cell cycles (see for instance [71]). These doubling transmission conditions are handled thanks to discontinuous flux conditions at the corresponding interfaces.

## Chapitre 4. Adaptive mesh refinement strategy for a non conservative transport problem

---

Such conditions are tricky to discretize numerically, even on a uniform grid, and require a specific scheme in order to achieve high order of accuracy. We refer the reader to [7] for a complete study of a Finite Volume (FV) scheme well suited to this purpose. We validate our adaptive strategy on both the localized and distributed mitosis models. The specific application we have in mind is the numerical calibration of terminal follicular development modeled in [41]. Realistic simulations involve solving simultaneously about twenty coupled PDEs in two space dimensions, during about one dozen of cell cycles, with an initial cell population concentrated in the first cell cycle. The weak structure of the coupling is suited to an efficient implementation on parallel architecture, already described in [6]. Nevertheless one simulation takes several minutes to run and the extensive simulations required to calibrate the model therefore motivate our search for additional reduction of computational costs by using a MR adaptive finite volume strategy. In this paper we address the problem arising from coupling the MR strategy with the FV method designed in [7] in the scalar case and with a single processor architecture, leaving the vector case with parallel implementation to future works. Actually, the main difficulty arising from this coupling is already present in the scalar case, where the resulting adaptive scheme (FVMR) unfortunately exhibits unstable numerical noise. The mesh adaptation is tuned by thresholding the details of the MR analysis, which introduces oscillations. In the standard conservative set-up, this spurious noise is bounded by a small parameter, and remains so at future times. In our non conservative situation it is amplified by the discontinuous flux conditions and becomes apparent only after long time simulations - let us say more than ten cell cycles. This corresponds to a proliferation ratio of  $2^{10}$  which explains how any numerical error, well within any given tolerance at the beginning of the simulation, can end up having macroscopic consequences on the solution mesh or even its shape (see Figure 4.14). It is a quite specific set-up, obviously not encountered in all PDEs with discontinuous fluxes. Hopefully they do not all require any particular adaptation of the MR method. This is for instance the case for Multiresolution applied to a traffic flow model [17] with only one (fixed and a priori known) flux discontinuity. If the discontinuity is well captured by the initial graded tree, it will remain that way, which explains why no specific strategy is needed. We will show that the non conservative nature of the equation is indeed the factor responsible for the noise amplification.

The main object of this article is therefore twofold : first to fully document the unstable noise appearance when using the standard MR method on such an equation with doubling conditions. This is done in section 4.2 where we introduce a simple 1D model and detail the elementary numerical steps, showing how they lead to spurious numerical noise. Secondly, we propose a numerical strategy to stop the noise appearance, and then validate it by extensive numerical tests. This numerical validation, exposed in section 4.3, uses the FVMR method that we also use in the latest section 4.4 to solve more realistic biological models. Eventually we show that our method correctly handles both types of models - localized or distributed mitosis - with CPU and memory gain between 10 and 20 for an overall relative error remaining below  $10^{-3}$ .

## 4.2 Investigation of numerical noise appearance

In this section we investigate the circumstances under which numerical noise can appear. For this we consider a simplified 1D problem with piecewise constant speeds. We compute the approximated solution with a simple first order FV scheme. This allows us to perform one time step of the algorithm by hand and completely describe the interaction of the doubling condition with a two-level MR algorithm. We illustrate this example by numerical simulations performed with a method coupling the 2D FV scheme with a mean value MR adaptive strategy.

### 4.2.1 Description of the simplified set-up

We study the simple case of a 1D transport model with piecewise constant speeds and discontinuous flux conditions

$$\begin{cases} \partial_t \phi(t, x) + \partial_x (g(x)\phi(t, x)) = 0, & \text{for } x \in \mathbb{R}, t > 0 \\ \phi(0, x) = \phi_0(x), & \text{for } x \in \mathbb{R}, \end{cases} \quad (4.1)$$

with  $N_c$  interfaces numbered from 1 to  $N_c$  spaced by  $x_s$ . The speeds are constant in each subdomain

$$g(x) = g_{p-1} \text{ for } (p-1)x_s \leq x \leq px_s, \quad p = 1, \dots, N_c + 1.$$

Transmission conditions are defined by

$$g_p \phi(t, px_s^+) = k_p g_{p-1} \phi(t, (p-1)x_s^-) \quad (4.2)$$

where  $k_p$  is equal to 2 to model a doubling flux interface, or to 1 to model a continuous flux interface.

For the sake of simplicity we first choose  $g_p = 1$  and  $k_p = 2$  for all  $p = 0, \dots, N_c$  and we apply a doubling condition on the outer boundary

$$\phi(t, 0^+) = 2\phi(t, N_c x_s^-).$$

This model is a simple prototype for transport equations modeling cell proliferation with localized mitosis at the end of each cell cycle, which is relevant for the granulosa cells in ovarian follicles [41]. We will also consider an alternative model frequently encountered in the literature (see for instance [64] chapter 3), which consists in distributing the mitosis over the cell cycle duration, with a linear source term

$$\begin{cases} \partial_t \phi(t, x) + \partial_x (g(x)\phi(t, x)) = B\phi(t, x), & \text{for } x \in \mathbb{R}, t > 0, \\ \phi(0, x) = \phi_0(x), & \text{for } x \in \mathbb{R}. \end{cases} \quad (4.3)$$

The coefficient  $B$  in the source term must be such that the averaged total mass is doubled at the end of a cell cycle, which leads to

$$\exp(BT) = 2,$$

where  $T$  is the duration of the cycle.

## Chapitre 4. Adaptive mesh refinement strategy for a non conservative transport problem

---

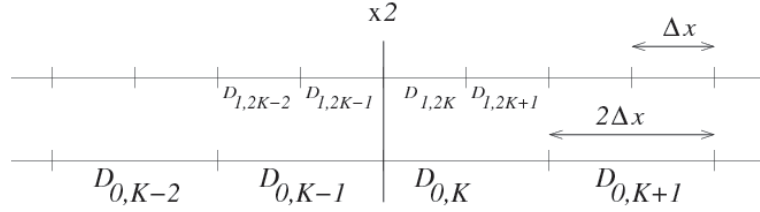


FIGURE 4.1 – Two level discretization with doubling interface located between meshes  $\mathcal{D}_{0,K-1}$  and  $\mathcal{D}_{0,K}$  on the coarser level, or  $\mathcal{D}_{1,2K-1}$  and  $\mathcal{D}_{1,2K}$  on the finer level.

### 4.2.2 Finite volume discretization on a two-level MR hierarchy

We start by discretizing the solution on two dyadically embedded levels  $\Delta x_0 = 2\Delta x$  where we denote the meshes on the finer level by

$$\mathcal{D}_{1,k} = [k\Delta x, (k+1)\Delta x],$$

and the meshes on the coarser level by

$$\mathcal{D}_{0,k} = \mathcal{D}_{1,2k} \cup \mathcal{D}_{1,2k+1} = [k\Delta x_0, (k+1)\Delta x_0].$$

In our double index notation  $\ell, k$ , the first number  $\ell$  indicates the level,  $\ell = 0$  for the coarser and  $\ell = 1$  for the finer. We further assume that  $x_s = 2K\Delta x = K\Delta x_0$ , meaning that the  $p$ -th doubling interface is located between meshes  $\mathcal{D}_{0,pK-1}$  and  $\mathcal{D}_{0,pK}$  on the coarser level, and between meshes  $\mathcal{D}_{1,2pK-1}$  and  $\mathcal{D}_{1,2pK}$  on the finer level, as illustrated in Figure 4.1 for the first interface ( $p = 1$ ).

The FV approximation of the solution is defined on the finer level by

$$\phi_{1,k}^n \approx \frac{1}{\Delta x} \int_{\mathcal{D}_{1,k}} \phi(n\Delta t, x) dx,$$

and the solution on the coarser level is defined as

$$\phi_{0,k}^n = \frac{1}{2} (\phi_{1,2k}^n + \phi_{1,2k+1}^n). \quad (4.4)$$

Knowing the mean values of the solution on the coarser level, approximated mean values on the finer level can be obtained by local reconstruction. Arbitrary orders of polynomial approximation can be used but we will consider in the sequel only the zeroth order reconstruction, also known as *Haar* reconstruction in the wavelet context

$$\hat{\phi}_{1,2k}^n = \hat{\phi}_{1,2k+1}^n = \phi_{0,k}^n, \quad (4.5)$$

and the second order reconstruction

$$\begin{aligned} \hat{\phi}_{1,2k}^n &= \frac{1}{8} \phi_{0,k-1}^n + \phi_{0,k}^n - \frac{1}{8} \phi_{0,k+1}^n, \\ \hat{\phi}_{1,2k+1}^n &= -\frac{1}{8} \phi_{0,k-1}^n + \phi_{0,k}^n + \frac{1}{8} \phi_{0,k+1}^n. \end{aligned} \quad (4.6)$$



## Chapitre 4. Adaptive mesh refinement strategy for a non conservative transport problem

---

We use a FV scheme to compute the numerical solution as

$$\phi_j^{n+1} = \phi_j^n - \frac{\Delta t}{\Delta x_j} \left( F_{j+1/2}^n - F_{j-1/2}^n \right), \quad (4.7)$$

where  $F_{j+1/2}^n$  is the numerical flux at time  $t_n$  through the edge between meshes  $\mathcal{D}_j$  and  $\mathcal{D}_{j+1}$ , and  $\Delta x_j$  the size of mesh  $\mathcal{D}_j$ . A simple upwind first order scheme on a uniform grid consists, in this case with unit speed, in taking

$$F_{j+1/2}^n = \begin{cases} \phi_j^n & \text{if } j \neq 0, pK - 1 \text{ and } 1, 2pK - 1, \\ \mathbf{2}\phi_j^n & \text{otherwise,} \end{cases} \quad (4.8)$$

where the factor  $\mathbf{2}$  in bold takes into account the doubling flux condition (4.2) when  $k_p = 2$ , for  $p = 1, \dots, Nc$ .

We can therefore compute the numerical solution on the finer level as

$$\phi_{1,k}^{n+1} = \begin{cases} (1 - \lambda)\phi_{1,k}^n + \lambda\phi_{1,k-1}^n & \text{if } k \neq 2pK, \\ (1 - \lambda)\phi_{1,k}^n + \mathbf{2}\lambda\phi_{1,k-1}^n & \text{if } k = 2pK, \end{cases} \quad \text{for } p = 1, \dots, Nc, \quad (4.9)$$

with

$$\lambda = \frac{\Delta t}{\Delta x},$$

which must satisfy  $0 < \lambda \leq 1$  in order to ensure the stability of the solution. If the solution is smooth enough - with a criterion that we will precise below - we can compute it instead on the coarser level, using  $\lambda/2$  instead of  $\lambda$

$$\phi_{0,k}^{n+1} = \begin{cases} (1 - \frac{\lambda}{2})\phi_{0,k}^n + \frac{\lambda}{2}\phi_{0,k-1}^n & \text{if } k \neq pK, \\ (1 - \frac{\lambda}{2})\phi_{0,k}^n + \frac{\lambda}{2}\mathbf{2}\phi_{0,k-1}^n & \text{if } k = pK, \end{cases} \quad \text{for } p = 1, \dots, Nc.$$

Note that the numerical solution becomes locally small but nonzero, even when starting from a piecewise constant solution : due to the numerical diffusion of the numerical scheme, the solution rapidly becomes nonzero upstream of the initial condition. For instance for the simple scheme (4.9) we have

$$\begin{aligned} \phi_{1,k}^n &= (1 - \lambda)\phi_{1,k}^{n-1} + \lambda\phi_{1,k-1}^{n-1}, \\ &= (1 - \lambda)^2\phi_{1,k}^{n-2} + 2(1 - \lambda)\lambda\phi_{1,k-1}^{n-2} + \lambda^2\phi_{1,k-2}^{n-2}, \\ &\vdots \\ &= (1 - \lambda)^n\phi_{1,k}^0 + \dots + \lambda^n\phi_{1,k-n}^0. \end{aligned} \quad (4.10)$$

In the general case with an arbitrary number of levels and arbitrary high dimensions, one time step of the FVMR scheme consists in a loop over levels starting from the finest. For each level, a first loop computes the numerical fluxes between meshes active at the current level using (4.8), possibly requiring local reconstruction if neighboring meshes entering the flux estimation are coarser than the current level. Then a second loop updates the solution on the meshes of the current level using (4.7).

Having in mind this algorithm is necessary to correctly handle the cases of neighboring meshes of different sizes

## Chapitre 4. Adaptive mesh refinement strategy for a non conservative transport problem

---

- if the left hand side neighbor of mesh  $\mathcal{D}_{1,2k}$  is the coarser mesh  $\mathcal{D}_{0,k-1}$ , then the solution  $\hat{\phi}_{1,2k-1}^n$  at time  $n$  on mesh  $\mathcal{D}_{1,2k-1}$  must be reconstructed, using (4.5) or (4.6), in order to compute  $F_{(1,2k)-1/2}^n$
- if the left hand side neighbor of mesh  $\mathcal{D}_{0,k}$  is the finer mesh  $\mathcal{D}_{1,2k-1}$ , then

$$F_{(0,k)-1/2}^n = F_{(1,2k-1)+1/2}^n = \phi_{1,2k-1}^n$$

has already been computed during the loop on the finer level

- if the right hand side neighbor of mesh  $\mathcal{D}_{1,2k+1}$  is the coarser mesh  $\mathcal{D}_{0,k+1}$ , no reconstruction is required to compute  $F_{(1,2k+1)+1/2}^n$  for the simple first order scheme (4.8). In the case of a higher order scheme,  $\hat{\phi}_{1,2k+2}^n$  would have to be reconstructed
- if the right hand side neighbor of mesh  $\mathcal{D}_{0,k}$  is the finer mesh  $\mathcal{D}_{1,2k+2}$ , then

$$F_{(0,k)+1/2}^n = F_{(1,2k+2)-1/2}^n = \hat{\phi}_{1,2k+1}^n$$

has already been computed during the loop on the finer level, using reconstructed value  $\hat{\phi}_{1,2k+1}^n$

The choice between the two levels of discretization on a given mesh  $\mathcal{D}_{0,k}$  in the domain is made by comparing the solution on its subdivisions  $\mathcal{D}_{1,2k} \cup \mathcal{D}_{1,2k+1}$  with the solution reconstructed on the same mesh using the values on the coarser grid with formulas (4.5) or (4.6). The local smoothness of the solution is indeed measured by the *detail*, defined by

$$d_{1,k}^n = \phi_{1,2k}^n - \hat{\phi}_{1,2k}^n \quad (4.11)$$

- In the Haar case, a null detail corresponds to a locally constant function

$$\begin{aligned} d_{1,k}^n &= \phi_{1,2k}^n - \phi_{0,k}^n \\ &= \frac{1}{2}(\phi_{1,2k}^n - \phi_{1,2k+1}^n). \end{aligned} \quad (4.12)$$

- In the quadratic basis, a null detail corresponds to a locally quadratic function

$$\begin{aligned} d_{1,k}^n &= \phi_{1,2k}^n - \left( \frac{1}{8}\phi_{0,k-1}^n + \phi_{0,k}^n - \frac{1}{8}\phi_{0,k+1}^n \right) \\ &= -\frac{1}{16}(\phi_{1,2k-2}^n + \phi_{1,2k-1}^n) + \frac{1}{2}(\phi_{1,2k}^n - \phi_{1,2k+1}^n) \\ &\quad + \frac{1}{16}(\phi_{1,2k+2}^n + \phi_{1,2k+3}^n). \end{aligned} \quad (4.13)$$

We refer for instance to [45] for the approximation theory results establishing the link between  $d_{1,k}^n$  and the local smoothness of the solution.

In order to monitor the adaptive FV scheme, the absolute value of the details are tested at each time step against a level dependent threshold  $\varepsilon_\ell$

$$\varepsilon_\ell = \varepsilon 2^{\ell-L+1}, \quad \text{for } \ell = 1, \dots, L-1. \quad (4.14)$$

## Chapitre 4. Adaptive mesh refinement strategy for a non conservative transport problem

---

In the numerical simulations, we refer to the MR threshold to indicate the threshold value on the finest MR level  $\varepsilon = \varepsilon_{L-1}$ . The set of indices of the significant details at time  $t_n$  is designed as

$$\Gamma_\varepsilon^n = \{(\ell, k), \|d_{\ell,k}\| \leq \varepsilon\},$$

and enlarged into a set  $\tilde{\Gamma}_\varepsilon^{n+1}$  according a heuristic rule [45], accounting for singularity displacement (and formation in the nonlinear case) between time  $t_n$  and  $t_{n+1}$ . Here in the two level case, if  $|d_{1,k}^n| \geq \varepsilon_1$ , then not only  $\mathcal{D}_{0,k}$  but also the two neighbors  $\mathcal{D}_{0,k-1}$  and  $\mathcal{D}_{0,k+1}$  should be subdivided to compute the solution at the next time step. The whole predictive procedure of adaptive grid refinement is summarized in the general case with  $L$  MR levels in Algorithm 1. The solution  $(\phi_{\ell,k}^n)_{(\ell,k) \in \tilde{\mathcal{S}}^{n+1}}$  is reconstructed on the adaptive grid at any given time  $t_n$  with Algorithm 2. It is then updated to time  $t_{n+1}$  with a FV scheme. Theoretically and practically – although it implies to renounce the MR computational benefits– it is also possible to compute the reconstructed solution  $\phi_\varepsilon^n$  on the finest grid using Algorithm 2 and setting all  $d_{l,k}^n = 0$  for  $(l, k) \in \tilde{\Gamma}_0^{n+1} \setminus \tilde{\Gamma}_\varepsilon^{n+1}$ . In the case of a scalar hyperbolic equation numerically discretized with a Lipschitz flux, the  $L^1$  norm error  $a_n$  between  $\phi_\varepsilon^n$  and the FV solution on the finest level  $\phi_{unif}^n$  has been shown in [30] to be bounded

$$a_n := \|\phi_{unif}^n - \phi_\varepsilon^n\| \leq C(t_n)\varepsilon. \quad (4.15)$$

The threshold  $\varepsilon$  should be chosen so that  $a_n$  is of the same order as the estimate of the discretization error  $e_n$

$$e_n := \|\phi_{unif}^n - \phi_{exact}^n\| \leq \bar{C}(t_n)\Delta x^p, \quad (4.16)$$

where  $p$  is the numerical order of the reference FV scheme on the uniform grid. Of course in realistic set-ups the exact solution is unavailable and one also wishes to avoid computing the FV reference solution on the fine grid. Estimates (4.15) and (4.16) can be used to calibrate the threshold value on a smaller scale domain in time and space, or for a simplified model. In our case, we will see in the numerical tests that choosing  $\varepsilon$  between  $10^{-2}$  and  $10^{-3}$  supplies a good compromise between accuracy and efficiency.

### 4.2.3 Explication of the numerical noise appearance

In this paragraph we describe the mechanism for numerical noise appearance due to the coupling of Multiresolution with doubling fluxes.

Suppose that  $|d_{K-i}^n| < \varepsilon$ , for  $i \geq 0$  and  $|d_{K+1}^n| \geq \varepsilon$ . Due to the preceding rule, the mesh  $\mathcal{D}_{0,K} = \mathcal{D}_{1,2K} \cup \mathcal{D}_{1,2K+1}$  is subdivided, to account for possible displacement of the singularity (see Algorithm 1). The mesh  $\mathcal{D}_{0,K-1}$  is kept at the coarser level, but the value of the solution on its right subdivision  $\mathcal{D}_{1,2K-1}$ , which enters the FV scheme to advance the solution on  $\mathcal{D}_{1,2K}$ , is available using the prediction operator (4.6) in the quadratic case.

## Chapitre 4. Adaptive mesh refinement strategy for a non conservative transport problem

---

Let us compute the solution at time  $n + 1$  in the neighborhood of the doubling interface (see Figure 4.1)

$$\begin{aligned}\phi_{0,K-1}^{n+1} &= \phi_{0,K-1}^n - \frac{\lambda}{2} \left( \hat{\phi}_{1,2K-1}^n - \phi_{0,K-2}^n \right), \\ \phi_{1,2K}^{n+1} &= (1 - \lambda)\phi_{1,2K}^n + 2\lambda\hat{\phi}_{1,2K-1}^n, \\ \phi_{1,2K+1}^{n+1} &= (1 - \lambda)\phi_{1,2K+1}^n + \lambda\phi_{1,2K}^n, \\ \phi_{1,2K+2}^{n+1} &= (1 - \lambda)\phi_{1,2K+2}^n + \lambda\phi_{1,2K+1}^n, \\ \phi_{1,2K+3}^{n+1} &= (1 - \lambda)\phi_{1,2K+3}^n + \lambda\phi_{1,2K+2}^n.\end{aligned}$$

We should now check the smoothness of the solution after this update. Using (4.13) and (4.4) we can compute  $d_K^{n+1}$  as

$$\begin{aligned}d_{1,K}^{n+1} &= \phi_{1,2K}^{n+1} - \hat{\phi}_{1,2K}^{n+1} \\ &= \phi_{1,2K}^{n+1} - \left( \frac{1}{8}\phi_{0,K-1}^{n+1} + \phi_{0,K}^{n+1} - \frac{1}{8}\phi_{0,K+1}^{n+1} \right) \\ &= -\frac{1}{8}\phi_{0,K-1}^{n+1} + \frac{1}{2}(\phi_{1,2K}^{n+1} - \phi_{1,2K+1}^{n+1}) + \frac{1}{16}(\phi_{1,2K+2}^{n+1} + \phi_{1,2K+3}^{n+1}).\end{aligned}$$

We plug in the expressions of the solution at time  $n + 1$  and obtain

$$\begin{aligned}d_K^{n+1} &= -\frac{\lambda}{16}\phi_{0,K-2}^n + \frac{\lambda}{2}(2\hat{\phi}_{1,2K-1}^n - \phi_{1,2K}^n) + \frac{\lambda}{16}(\phi_{1,2K+1}^n + \phi_{1,2K+2}^n) - \frac{1}{8}\phi_{0,K-1}^n \\ &\quad + \frac{\lambda}{16}\hat{\phi}_{1,2K-1}^n + \frac{1-\lambda}{2}(\phi_{1,2K}^n - \phi_{1,2K+1}^n) + \frac{1-\lambda}{16}(\phi_{1,2K+2}^n + \phi_{1,2K+3}^n).\end{aligned}$$

The effect of the doubling flux on the detail is an additional term

$$d_{1,K}^{n+1} = \hat{d}_{1,K}^{n+1} + \frac{\lambda}{2}\hat{\phi}_{1,2K-1}^n, \quad (4.17)$$

compared to the detail  $\hat{d}_K^{n+1}$  if no doubling condition is imposed. This means that if the solution is smooth at time  $t_n$ , but the solution is large enough before the doubling interface, then the detail may become significant due to the doubling.

We now explain what happens next, if the detail  $d_{1,K}^{n+1}$  which was negligible at time  $t_n$  becomes larger than the MR threshold  $\varepsilon$  at time  $n + 1$ . In the quadratic case, as described in Algorithm 1, the mesh  $\mathcal{D}_{0,K-1}$  just upstream of the doubling interface should be predictively refined using (4.6)

$$\begin{cases} \hat{\phi}_{1,2K-2}^{n+1} = \frac{1}{8}\phi_{0,K-2}^{n+1} + \phi_{0,K-1}^{n+1} - \frac{1}{8}\phi_{0,K}^{n+1}, \\ \hat{\phi}_{1,2K-1}^{n+1} = -\frac{1}{8}\phi_{0,K-2}^{n+1} + \phi_{0,K-1}^{n+1} + \frac{1}{8}\phi_{0,K}^{n+1}, \end{cases}$$

where

$$\begin{aligned}\phi_{0,K}^{n+1} &= \frac{1}{2}(\phi_{1,2K}^{n+1} + \phi_{1,2K+1}^{n+1}), \\ &= \frac{1-\lambda}{2}(\phi_{1,2K}^n + \phi_{1,2K+1}^n) + \frac{\lambda}{2}(2\hat{\phi}_{1,2K-1}^n + \phi_{1,2K}^n).\end{aligned} \quad (4.18)$$

## Chapitre 4. Adaptive mesh refinement strategy for a non conservative transport problem

---

Here again the difference between the values of the solution on the subdivisions, which should be small since the solution is only predictively refined, is enhanced by the value  $\hat{\phi}_{1,2K-1}^n$  upstream of the interface which contributes twice instead of once. We see that in a single time step the doubling condition coupled with the MR, has introduced an oscillation of the order of  $\hat{\phi}_{1,2K-1}^n$ .

Suppose we start from an initial condition localized in the first slab before the first doubling interface, for instance  $\phi_k^0 = \delta_{0,k}$ . Due to the numerical diffusion effects described earlier (see (4.10)), after  $M$  time steps ( $M$  depending on  $\varepsilon$  and  $\lambda$ ), the solution  $\phi_k^M$  is nonzero and smooth for  $k = 0, \dots, M$ . The MR scheme will therefore coarsen the grid in this region. As the solution is transported over several slabs, its amplitude is doubled at each interface, and depending on its local amplitude we can fall in the situation depicted by (4.17). Furthermore, another factor of noise appearance is the dependence of the MR threshold level on the discretization level according to the rule (4.14).

We see from (4.17) that the decision to refine the mesh after crossing a doubling interface depends not only on the smoothness but also on the local amplitude. Thanks to (4.14), the same (small) amplitude discretized on a coarse level will be more likely to trigger the refinement, and might even trigger the local refinement on two consecutive levels (see Algorithm 1). We propose an example for such a situation in Appendix 4.6. We describe a situation with three levels of discretization, where the solution has sensibly the same small amplitude but is refined partly on the intermediate level, and partly on the coarsest one. We show that refinement rules together with rules (4.17) and (4.14) lead to coarsen the solution wherever it was previously on the intermediate level, and to refine it on the finest level on some locations where it was before on the coarsest one.

At this point we also stress out that the difference between the two wavelets (4.12) and (4.13) is quite important, specially if we solve the 1D test case presented above with the 2D FV code. To illustrate this point we perform a simulation on a domain consisting of twelve subdomains of unit width and height with vertical doubling interfaces. The (horizontal) speed is equal to 1. The initial condition is piecewise constant

$$\phi(t=0, x, y) = \begin{cases} 10 & \text{for } 0 \leq x < 1 \text{ and } 0.2 \leq y < 0.3 \\ 0 & \text{elsewhere} \end{cases}$$

We first use a two level MR hierarchy, with five meshes per subdomain in the  $x$  direction on the coarser level. Figure 4.2 shows the solution computed at time 20 on the uniform finer grid, which counts ten meshes per subdomain in the  $x$  direction, along with the solution computed with the FVMR algorithm and Haar or quadratic reconstruction. Note that due to the detail amplification (4.17) the MR solutions with both Haar and quadratic reconstructions exhibit more refined meshes upstream than downstream of the support of the solution. Figure 4.3 shows the solution with a color code from 0 to 10 corresponding to the initial condition; the spurious noise is visible at this scale. Note that in the Haar case, the upstream amplification (4.18) does not happen. In the case of the quadratic reconstruction,

## Chapitre 4. Adaptive mesh refinement strategy for a non conservative transport problem

---

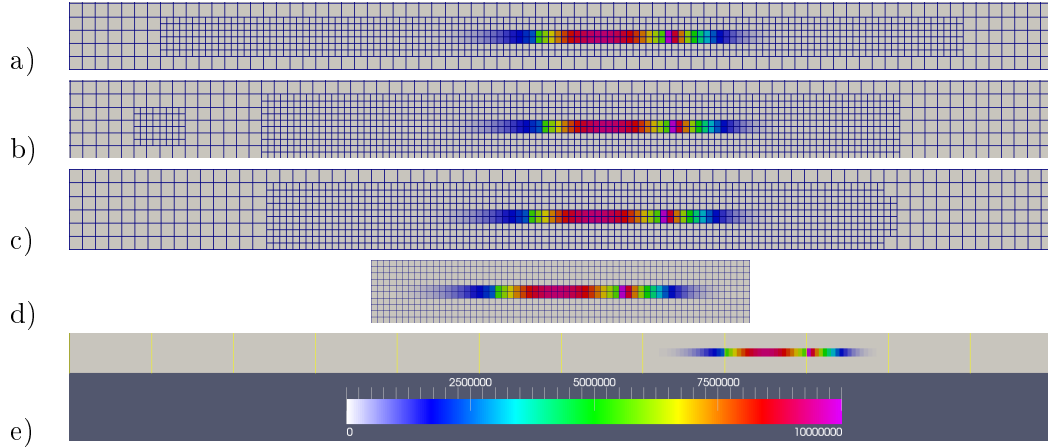


FIGURE 4.2 – Snapshots of the solution  $\phi(t, x, y)$  at  $t = 20$  on the whole computational domain and color code from 0 to  $10^7$ . Panels from top to bottom : a) Zoom on the support of the solution, MR with Haar reconstruction, b) MR with quadratic reconstruction, c) MR with quadratic reconstruction and doubling threshold strategy (4.20) d) finer grid no MR, e) full finer grid no MR. Notice the presence of the spurious noise (refined meshes upstream of and contiguous to the support of the solution in panel a) and b), as well as isolated upstream refined meshes in panel b)), and the absence of noise (in panel c)).

solving the 1D transport equation (4.1) with the 2D MR scheme also generates some nonzero solution in the orthogonal direction, which gets progressively amplified by the doubling interfaces. The difference between the panels corresponding to the Haar (panel a)) and quadratic predictions (panel b)) highlights the appearance of noise due to the quadratic prediction operator in 2D : the solution becomes nonzero in the transversal direction above and below the support of the initial condition. The quadratic prediction performs a better compression of the solution than the Haar one, therefore the mesh is coarsened faster away from the support of the solution. However in both cases the details are later amplified by (4.17) and, in the quadratic case, also by (4.18), which explains the resurgence of refined meshes upstream of the support of the solution (panels b)). In this simple set-up isolated cells appear on the finest level only in the quadratic case, but in more complicated situations the same phenomenon also occurs in the case of the Haar prediction.

### 4.2.4 Adaptive doubling strategy

To prevent the formation of this "numerical noise" several strategies can be implemented :

- Forcibly refining the solution on the finest discretization grid in the vicinity of interfaces, without considering its local smoothness. This may become quite greedy and altogether compensate the benefits of the MR strategy.
- Adapting the MR threshold to the average amplitude of the solution. This should

## Chapitre 4. Adaptive mesh refinement strategy for a non conservative transport problem

---

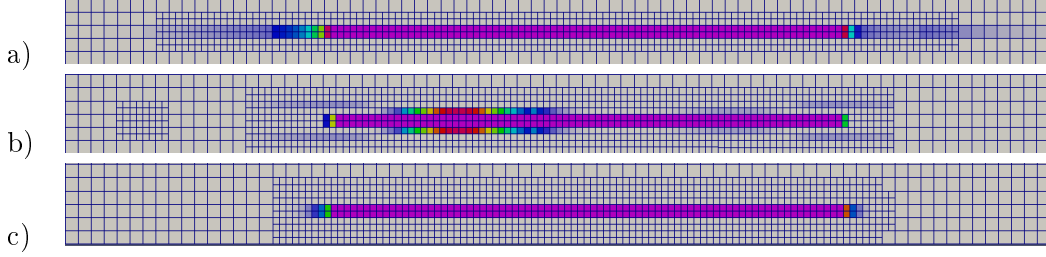


FIGURE 4.3 – Zoom on the support of the solution at  $t = 20$  and computational grid with color code from 0 to 10. From top to bottom a) MR with Haar reconstruction, b) MR with quadratic reconstruction, c) MR with quadratic reconstruction and doubling threshold (4.20). The thresholding error is more visible in the Haar case (panel a)) than in the quadratic reconstruction (panels b) and c)).

be done carefully to avoid rapid changes in the threshold and subsequently rapid coarsening of the solution. We will document later that the optimal strategy is to adapt the threshold to the total mass of the solution

$$\varepsilon_{t,\ell} = \varepsilon_\ell M(t) \quad (4.19)$$

where the level dependent  $\varepsilon_\ell$  is defined by (4.14), with  $\varepsilon$  calibrated for a unit mass solution at initial time.

- Modifying the transmission condition (4.2) with an adaptive strategy consisting in applying the doubling only if the solution is above a given threshold  $\delta$

$$g_p \phi(t, px_s^+) = \begin{cases} 2g_{p-1} \phi((p-1)x_s^-) & \text{if } |\phi(t, (p-1)x_s^-)| \geq \delta, \\ g_{p-1} \phi((p-1)x_s^-) & \text{otherwise.} \end{cases} \quad (4.20)$$

In practice  $\delta$  should be chosen close to  $\varepsilon$ , and we will study the relationship more closely in paragraph 4.3.3.

We also assume that the support of the initial condition is entirely located in the first slab before the first doubling interface. We have actually exemplified in Appendix 4.6 that a small and smooth solution lying on both sides of an interface, could right away generate spurious oscillations. Restricting the initial support ensures that no additional oscillations will be created by coupling the MR with the doubling condition. The bottom panels of Figures 4.2 and 4.3 display the effect of the adaptive doubling strategy in the case of second order reconstruction, setting the parameter  $\delta$  equal to  $\varepsilon = 10^{-2}$ . The numerical noise has completely disappeared.

### 4.2.5 Formalization and generalization of the doubling strategy

The result (4.17) also applies to Haar reconstruction (4.12). This means that the appearance of the spurious noise is due not only to the discontinuities in the solution induced by the localized doubling, but also to the enhancement of the thresholding error. In particular, even if we modified the prediction scheme in the vicinity of

## Chapitre 4. Adaptive mesh refinement strategy for a non conservative transport problem

---

interfaces in order to account for the doubling of the solution, this would not stop the increase of noise.

To formalize this remark, we generalize the MR scheme introduced in Paragraph 2.2 to an arbitrary number of levels  $L$ . We introduce the linear prediction operator  $P_\ell^{\ell-1}$  such that formula (4.6) can be expressed generically as

$$\hat{\phi}_\ell^n = P_\ell^{\ell-1} \phi_{\ell-1}^n,$$

and the reconstruction formulas in Algorithm 2 as

$$\phi_\ell^n = P_\ell^{\ell-1} \phi_{\ell-1}^n + d_\ell. \quad (4.21)$$

Operator  $P_\ell^{\ell-1}$  is a  $2^\ell N_0 \times 2^{\ell-1} N_0$  matrix, that is zero everywhere except at three consecutive coefficients per row

$$\begin{aligned} (P_\ell^{\ell-1})_{2k, k-1} &= 1/8, & (P_\ell^{\ell-1})_{2k, k} &= 1, & (P_\ell^{\ell-1})_{2k, k+1} &= -1/8, \\ (P_\ell^{\ell-1})_{2k+1, k-1} &= -1/8, & (P_\ell^{\ell-1})_{2k+1, k} &= 1, & (P_\ell^{\ell-1})_{2k+1, k+1} &= 1/8, \end{aligned}$$

with periodic boundary conditions

$$\begin{aligned} (P_\ell^{\ell-1})_{0, 2^{\ell-1} N_0 - 1} &= 1/8, & (P_\ell^{\ell-1})_{2^\ell N_0 - 2, 0} &= -1/8, \\ (P_\ell^{\ell-1})_{1, 2^{\ell-1} N_0 - 1} &= -1/8, & (P_\ell^{\ell-1})_{2^\ell N_0 - 1, 0} &= 1/8. \end{aligned} \quad (4.22)$$

Note that other types of boundary conditions can be designed but periodic ones are coherent with the boundary conditions used in the application to the biological model. Formula (4.21) can be applied recursively, leading to

$$\phi_{L-1}^n = \prod_{\ell=1}^{L-1} P_\ell^{\ell-1} \phi_0^n + \sum_{\ell=1}^{L-1} \prod_{j=\ell+1}^{L-1} P_j^{j-1} d_\ell^n. \quad (4.23)$$

The adaptive solution after thresholding with the MR threshold  $\varepsilon$  can be expressed as

$$\phi_{L-1}^{n, \varepsilon} = \prod_{\ell=1}^{L-1} P_\ell^{\ell-1} \phi_0^n + \sum_{\ell=1}^{L-1} \prod_{j=\ell+1}^{L-1} P_j^{j-1} d_{\ell, \varepsilon}^n, \quad (4.24)$$

where  $d_{\ell, \varepsilon}^n$  is the vector of thresholded details on level  $\ell$

$$(d_{\ell, \varepsilon}^n)_{\ell, k} = \begin{cases} d_{\ell, k}^n & \text{if } \|d_{\ell, k}^n\| \geq \varepsilon_\ell, \\ 0 & \text{otherwise.} \end{cases}$$

From (4.23) and (4.24) it is straightforward to obtain an  $O(\varepsilon)$  estimate of the error between the unthresholded solution  $\phi_{L-1}^n$  and the thresholded one  $\phi_{L-1}^{n, \varepsilon}$ . In [30], a strategy was introduced to enlarge the tree of significant details (with respect to the threshold  $\varepsilon$ ), ensuring that the adaptive solution on the corresponding mesh satisfies the same error estimate at the next time step time  $n + 1$ . We also recall a useful result from [35] showing that the reconstructed solution on any intermediate level is



## Chapitre 4. Adaptive mesh refinement strategy for a non conservative transport problem

---

vitiated by an  $O(\varepsilon)$  perturbation due to MR thresholding and reconstruction, which we can summarize as

$$\phi_{\ell,k}^{n,\varepsilon} = \phi_{\ell,k}^n + O(\varepsilon). \quad (4.25)$$

We apply the adaptive doubling strategy (4.20) along with the elementary FV scheme (4.9) generalized to an  $L$  level MR hierarchy with  $\lambda_\ell = 2^{\ell-L+1}\lambda$

$$\phi_{\ell,k}^{n+1,\varepsilon} = \begin{cases} (1 - \lambda_\ell)\phi_{\ell,k}^{n,\varepsilon} + \lambda_\ell\phi_{\ell,k-1}^{n,\varepsilon} & \text{if } k \neq pK \text{ or } |\phi_{\ell,k-1}^{n,\varepsilon}| \leq \delta \\ (1 - \lambda_\ell)\phi_{\ell,k}^{n,\varepsilon} + \lambda_\ell\mathbf{2}\phi_{\ell,k-1}^{n,\varepsilon} & \text{if } k = pK \text{ and } |\phi_{\ell,k-1}^{n,\varepsilon}| > \delta, \end{cases}$$

with  $p = 1, \dots, N_c$ . We see that we have a conditional maximum principle in the sense that

$$\text{if } \phi_{\ell,k}^{n,\varepsilon} \leq \delta \text{ and } \phi_{\ell,k-1}^{n,\varepsilon} \leq \delta \text{ then } \phi_{\ell,k}^{n+1,\varepsilon} \leq \delta. \quad (4.26)$$

Relations (4.25) and (4.26) together show that parameter  $\delta$  should be correctly calibrated in relation with the MR threshold  $\varepsilon$ , to prevent the MR perturbation in (4.25) from being amplified and becoming the unwanted spurious noise.

### 4.3 Numerical validation of the adaptive thresholds strategies

In this paragraph we perform numerical tests to validate the adaptive strategies (4.19) and (4.20). We then study the relationship between the two thresholds  $\delta$  and  $\varepsilon$ .

#### 4.3.1 Effect of the adaptive doubling strategy on the solution

To justify the doubling strategy designed to tackle the spurious noise introduced by the MR we show the influence of applying (4.20) instead of (4.2), independently of the MR, and even of any numerical scheme, by computing the exact solution of the 1D problem (4.1). As shown in [7], the exact solution for a single interface located at  $x = x_s$  with condition (4.2) is

$$\phi_1(t, x) = \begin{cases} \phi_0(x - g_0t) & \text{for } x < x_s, \\ \tilde{\phi}_1(t_s) & \text{for } (x_s - g_1t) \leq (x - g_1t) < x_s, \\ \phi_0(x - g_1t) & \text{for } (x - g_1t) \geq x_s, \end{cases}$$

where  $\tilde{\phi}_1$  is the trace of the solution on the right of the interface, defined by

$$\tilde{\phi}_1(t) = \frac{1}{g_1}k_1(g_0\phi_0(x_s - g_0t)), \quad (4.27)$$

and  $t_s$  is the delay after which the effect of interface  $x_s$  is felt at position  $x$  and time  $t$  defined by

$$t_s = t - \frac{(x - x_s)}{g_1}. \quad (4.28)$$

## Chapitre 4. Adaptive mesh refinement strategy for a non conservative transport problem

---

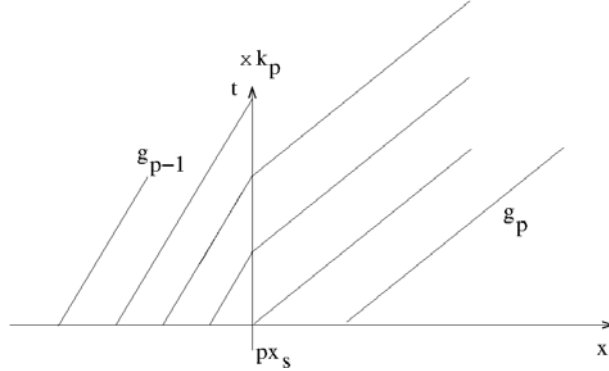


FIGURE 4.4 – Set-up for the  $p^{\text{th}}$  interface. The characteristic curves in the right part of the domain transport either the initial condition if they cross the  $x$ -axis or the trace of the solution on the  $px_s$  interface, multiplied by the  $k_p$  coefficient, if they cross the vertical  $x = px_s$  axis

We now show how the exact solution for a sequence of  $N_c$  interfaces can be defined recursively. Let us denote by  $\phi_{p-1}(t, x)$  the solution for a sequence of  $p-1$  interfaces, with  $p > 1$  and add a  $p^{\text{th}}$  interface at  $px_s$  with a coefficient  $k_p$  for the transmission condition and a speed  $g_p$  on the right hand side (see Figure 4.4). For  $x < px_s$  the effect of the  $p^{\text{th}}$  interface is not felt and the solution is  $\phi_{p-1}(t, x)$ . For  $x \geq px_s$ , depending on time  $t$ , the effect of the interface is felt or not. If  $(x - px_s) < g_p t$ , events crossing the  $p^{\text{th}}$  interface are multiplied by the  $k_p$  coefficient and transported at speed  $g_p$ . The effects of these events are felt at position  $x$  after a delay  $(x - px_s)/g_p$ . If  $(x - px_s) > g_p t$ , the effect of the  $p^{\text{th}}$  interface is not felt yet and the initial condition defined on  $x > px_s$  is transported at speed  $g_p$ . We eventually have

$$\phi_p(x, t) = \begin{cases} \phi_{p-1}(x, t) & \text{for } x < px_s, \\ \tilde{\phi}_p(t_s^p) & \text{for } (px_s - g_p t) \leq (x - g_p t) < px_s, \\ \phi_0(x - g_p t) & \text{for } x - g_p t \geq px_s, \end{cases} \quad (4.29)$$

where  $\tilde{\phi}_p$  is the trace of the solution on the right of the  $p^{\text{th}}$  interface, defined by

$$\tilde{\phi}_p(t) = \begin{cases} \frac{1}{g_p} k_p g_{p-1} \phi_{p-1}(px_s, t) & \text{if } p > 1 \\ \frac{1}{g_p} k_p g_{p-1} \phi_0(x_s - g_{p-1} t) & \text{otherwise} \end{cases} \quad (4.30)$$

and  $t_s^p$  is defined by

$$t_s^p = t - \frac{(x - px_s)}{g_p}.$$

We mimic set-up (4.1) by solving equation (4.29) numerically using recursiveness, with  $g_p = 1$  for all  $p$  and an initial condition  $\phi_0$  defined by

$$\phi_0(x) = \exp(-5(x - 0.5)^2). \quad (4.31)$$

## Chapitre 4. Adaptive mesh refinement strategy for a non conservative transport problem

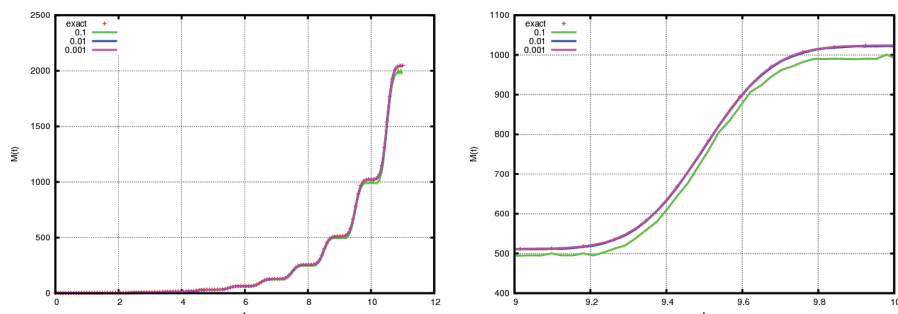


FIGURE 4.5 – Mass with respect to time computed with the recursive definition (4.29), without and with the doubling strategy (4.20) for different values of the parameter  $\delta$ . Full time range on the left and zoom on the final region on the right.

We also mimic strategy (4.20), that we plan to apply to our real problem, and use the doubling condition in (4.30) only if the solution is larger than the doubling threshold  $\delta$ . At each time step, we compute the total mass by numerically integrating the solution over the whole  $x$  abscissa range  $[0, 20]$ . Figure 4.5 displays the total mass with respect to time, for different values of the threshold parameter  $\delta$

$$M^{\delta,n} = \Delta x \sum_j \phi^\delta(n\Delta t, j\Delta x)$$

along with the exact solution

$$M^n = \Delta x \sum_j \phi(n\Delta t, j\Delta x).$$

Figure 4.6 displays the L1 norm over the time interval of the error between the thresholded and exact masses, normalized by the L1 norm of the mass of the exact solution

$$Err(\delta) = \frac{\sum_n |M^{\delta,n} - M^n|}{\sum_n |M^n|}, \quad (4.32)$$

where  $\phi^\delta$  (respectively  $\phi$ ) is computed on the discrete time-space grid  $\Delta t \times \Delta x$  with (4.29) and with (respectively without) the adaptive doubling strategy (4.20). The error is displayed as a function of the threshold parameter  $\delta$ , in logarithmic axes, which highlights its linear dependence onto the threshold.

In Figure 4.6 we also display the error behavior for the same simulation in the case when the parameter  $\delta$  changes with time, which would happen if the MR threshold  $\varepsilon$  followed the time-dependent rule (4.19). We choose in fact

$$\delta_t = \delta M(t), \quad (4.33)$$

where  $M(t)$  is the mass of the solution integrated over the whole spatial range, at time  $t$

$$M(t) = \int_0^{N x_s} \phi(t, x) dx.$$

## Chapitre 4. Adaptive mesh refinement strategy for a non conservative transport problem

---

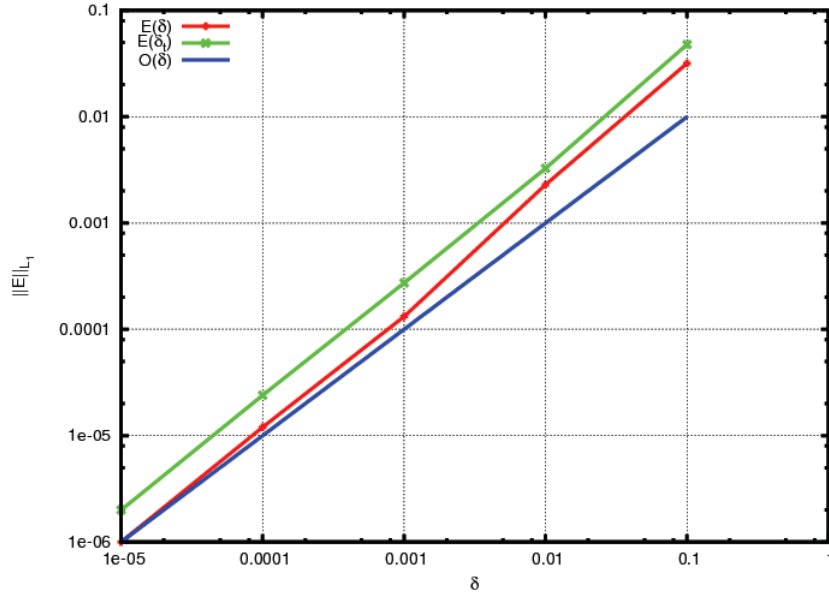


FIGURE 4.6 – L1 norm error (4.32) of mass with respect to the parameter  $\delta$  of the doubling strategy (4.20) with ( $E(\delta_t)$ ) and without ( $E(\delta)$ ) time adaptivity (4.33). The curve  $O(\delta)$  indicates the theoretical asymptotic behavior which is a straight line of slope one in logarithmic axes.

These numerical experiments show that whichever the selected thresholding strategy for the MR method, the influence of the adaptive doubling strategy (4.20) is well controlled by parameter  $\delta$ . Except for the unreasonably large value of  $\delta = 0.1$ , the mass behavior computed with the adaptive doubling strategy is undistinguishable from the exact solution. Both error curves in Figure 4.6 exhibit a better behavior than the asymptotic  $O(\delta)$ .

In Figure 4.7 we check that the numerical scheme FVMR described in the previous section correctly reproduces this macroscopic mass behavior. We run a 2D simulation of the 1D simple model (4.1) and we compare it to case (4.3) where the doubling is uniformly distributed over the cycle, with a linear birth term amounting to the same overall doubling. In Figure 4.8 snapshots of solutions at  $t = 7.48$  and  $t = 8$  are displayed, allowing one to compare at the microscopic level the effect of either applying a doubling condition locally at the end of interface or smoothly distributing proliferation over all the cycle. At both times the solutions sensibly have the same mass, we have selected the snapshot at times when the red and dashed green curves intersect in Figure 4.7. The top panel, at  $t = 7.48$ , corresponds to a time when the support of the solution is equally distributed on one side and the other of an interface between cell cycles and the induced discontinuity in the solution is clearly visible on the left panel. The bottom panel, at  $t = 8$  corresponds to a time when the support of the solution is well within the same cell cycle, and both solutions do not substantially differ.

## Chapitre 4. Adaptive mesh refinement strategy for a non conservative transport problem

---

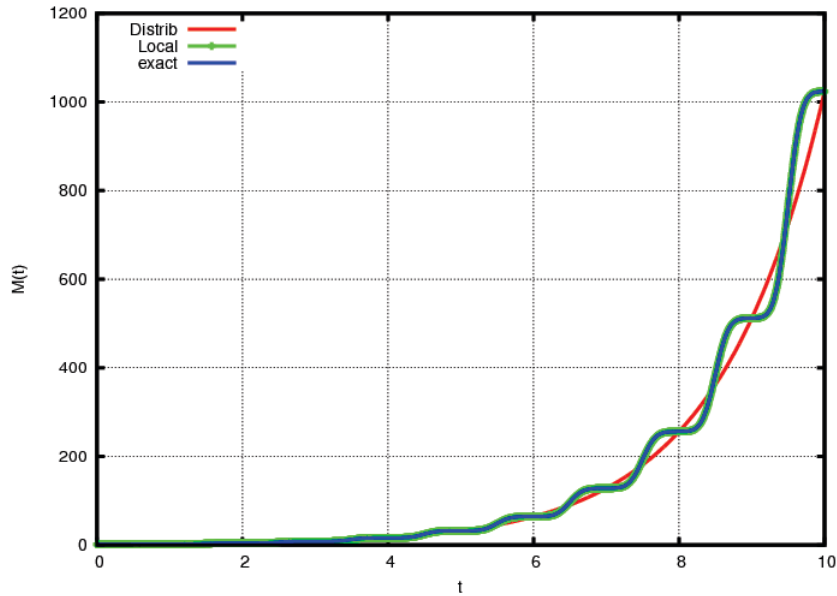


FIGURE 4.7 – Mass over eleven cycles as a function of time for localized and distributed doubling. Third order scheme, four MR levels,  $\varepsilon = 10^{-3}$ . The exact solution (4.29) is superimposed on the FVMR solution.

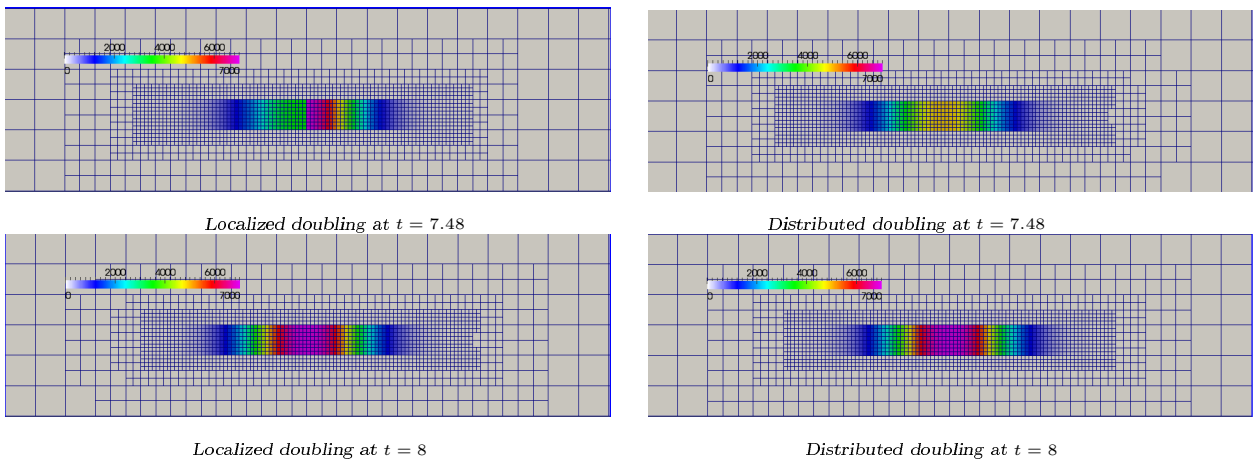


FIGURE 4.8 – Snapshot of the density  $\phi(t, x, y)$  at time  $t = 7.48$  (top panels) and  $t = 8$  (bottom panels). Left panels : localized doubling, right panels : distributed gain over the cell cycle. The color bar giving the amplitude of the density (between 0 and 7000) is the same for all panels. The initial solution, given by (4.31), has a unit mass.

### 4.3.2 Time adaptation of the MR threshold $\varepsilon_t$

To study the influence of the time adaptation of the MR threshold  $\varepsilon_t$  (4.19), we perform different simulations of simple test case (4.1) with smooth initial condition (4.31), the third order FV scheme and a 4 level MR hierarchy with the quadratic reconstruction. We test by turns, the time adaptation of the MR threshold (4.19), and another adaptation of the MR threshold based on the maximum value of the solution

$$\varepsilon_{t,\ell} = \varepsilon_\ell \max_{x,y} |\phi(t, x, y)|. \quad (4.34)$$

In each case we perform the simulation with or without the adaptive doubling strategy (4.20). The six possible cases are referred in the graphs with the following legends

The  $L^1$ -norm of the relative error at a given time  $t_n = n\Delta t$  is defined as

$$\|E\|_{L^1} = \frac{\sum_{i,j} |\phi_{L-1;i,j}^{\varepsilon,n} - \phi_{L-1;i,j}^n|}{\sum_{i,j} |\phi_{L-1;i,j}^n|}, \quad (4.35)$$

where  $\phi_{L-1;i,j}^n$  is the FV solution at time  $t_n = n\Delta t$  on the finest MR level  $L - 1$  of discretization, with index  $i$  in the  $x$  direction and  $j$  in the  $y$  direction, and  $\phi_{L-1;i,j}^{\varepsilon,n}$  is the adaptive scheme solution, reconstructed at time  $t_n$  on the finest level  $L - 1$ . This error is computed at five different times and displayed in Figure 4.9 against the MR threshold level  $\varepsilon$  on the left panels, the CPU gain on the center panels, and the memory gain on the right panels. The CPU gain is the ratio of the CPU time for the simulation on the uniform finest grid to the CPU time for the FVMR simulation. The memory gain is the ratio of the number of meshes in the uniform finest grid to the maximum number of meshes in the adaptive grid during the FVMR simulation. The left panels show that, whichever the thresholding strategy, the expected asymptotic behavior in  $O(\varepsilon)$  is satisfied (see equation (4.15)). Nevertheless, as time increases, the normalization factors in the adaptive strategies (4.19) and (4.34) increase, resulting in more and more distinct curves. In practice, this type of curve is computed for a reduced version of the full scale simulation in order to select a threshold value  $\varepsilon$  ensuring a correct balance between the MR (4.15) and discretization (4.16) errors. In our case we will use the optimal values of the thresholds  $\delta$  and  $\varepsilon$  calibrated from this simple 1D set-up in future 2D and vector simulations of the biological model. In other situations, one could calibrate the MR threshold in 2D tests before addressing the 3D cases where the reference solution is beyond reach. We have represented the error in  $L^1$  norm in space at various times during the simulation, which gives more detailed information than the global  $L^1$  error over the whole timespan, and allows us in particular to see the variation in the error as the time adaptive threshold increases. For a constant threshold (red and green lines) the error remains almost unchanged with time. This is not a surprise since we have already observed in [30] that the theoretical estimate (4.15) predicting a linear

growth in time was very pessimistic. Also quite expectedly, the difference between the error curves increases with time in the case of time adaptive thresholding with increasing value of  $\varepsilon$ . Note that the MR error at a given time has built up progressively and does not entirely depend on the current value of  $\varepsilon(t)$  : at time  $t = 10$  there is a factor of order 10 between the red and dark blue curves, which correspond respectively to the constant and mass dependent threshold (Eq. (4.19) and (4.33)), although at that time the mass, and therefore  $\varepsilon(t)$ , has already been multiplied by  $10^3$ . At the end of the simulation (at  $t = 20$ ) the error ratio between both curves is still about 10, although  $\varepsilon(t)$  has been multiplied by  $10^6$ .

The center and right panels allow one to assess the performance of the thresholding strategies with respect to computational costs, in terms of CPU time as well as memory requirement. For early times all strategies seem equivalent, but after ten cycles, the simulations corresponding to constant MR thresholding start having worse performances than those using either of adaptive strategies (4.19) and (4.34), which remain basically undistinguishable. The simulation with constant MR thresholding and no doubling adaptation, displayed with  $\times$ , has the worst performances, because at large times, spurious noise has started to develop, requiring fine mesh discretization and therefore, more memory and more CPU time.

We now validate the adaptive doubling strategy (4.20) by extensive numerical tests. We compare its efficiency when it is coupled with one or another method of MR thresholding. We also estimate the optimal value for the threshold parameter  $\delta$ .

### 4.3.3 Link between the MR and the doubling thresholds

In this paragraph we study the relation between the MR threshold parameter  $\varepsilon$  and the threshold  $\delta$  in the adaptive doubling interface (4.20). Still for 1D test case (4.1) with the piecewise constant initial condition, we run extensive tests for different values of both thresholds. We also study the influence of the number of levels in the MR hierarchy, the precision of the FV algorithm, and the effect of normalizing the MR threshold by the overall mass of the solution, as it increases with time (see Eq. (4.19)).

Figure 4.10 displays the  $L_1$  norm of error at final time  $t = 20$ , between the adaptive solution and the solution computed on the uniform finest grid. The adaptive solution is computed with different values of the MR threshold  $\varepsilon$ , fixed once for all times, and for values of the threshold  $\delta$  varying between  $\varepsilon/100$  and  $10\varepsilon$ . For two, three and four levels in the MR hierarchy we display the results in two different fashions : The standard error visualization is shown in the left panels, where the interesting feature is the linear trend of the error with  $\varepsilon$  which is preserved by the adaptive doubling strategy. However normalizing by  $\varepsilon$  both the error on the vertical axis and  $\delta$  on the horizontal axis highlights the correlation between  $\varepsilon$  and  $\delta$  since the minimum of the error is observed for a similar ratio range for each set of curves.

Figure 4.10 displays results computed with the first order upwind FV scheme (4.7-4.8). The value of  $\delta$  which minimizes the error lies between  $0.2\varepsilon$  and  $2\varepsilon$  and increases with both  $\varepsilon$  and the number of levels. We also made the same test for four levels

## Chapitre 4. Adaptive mesh refinement strategy for a non conservative transport problem

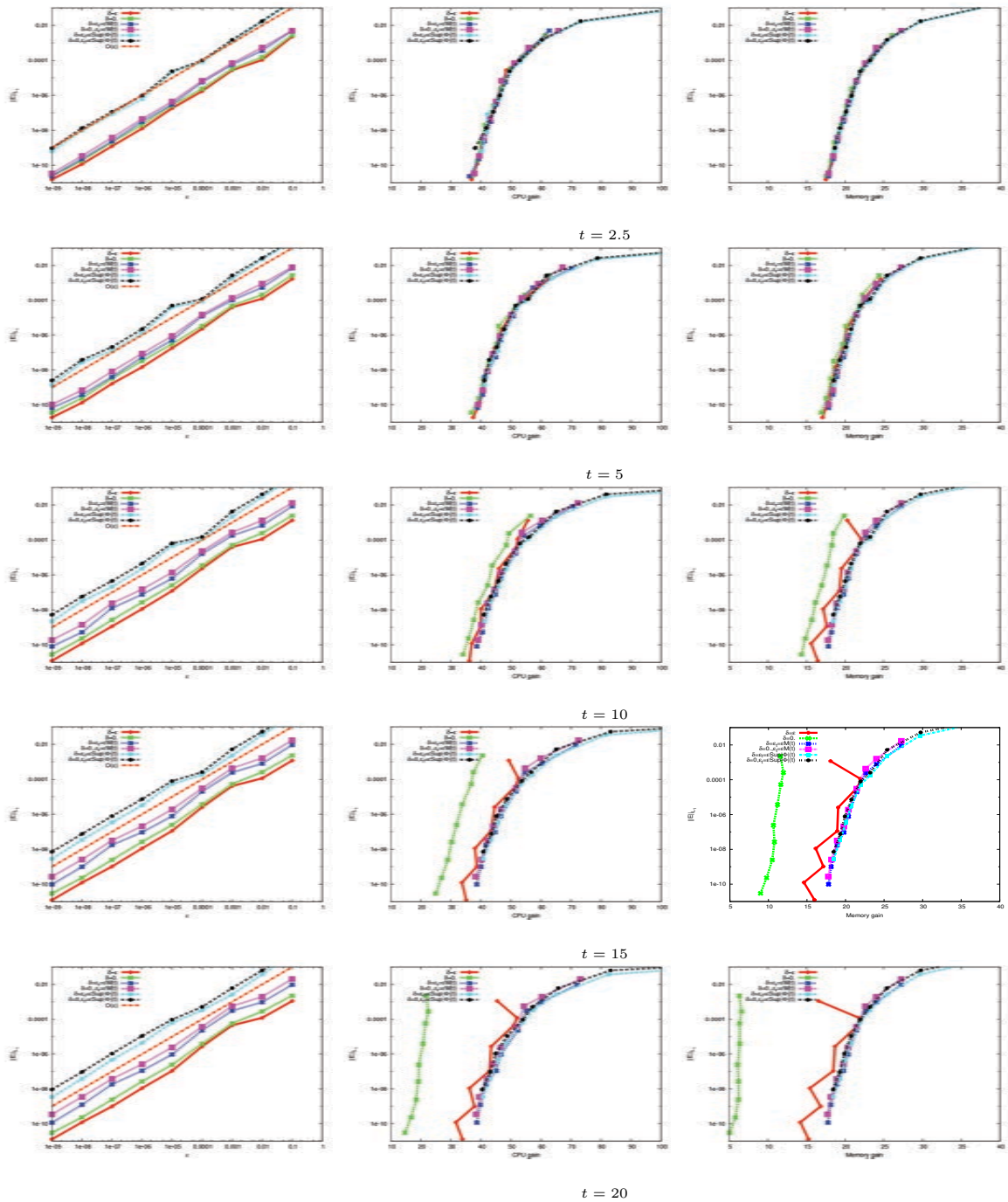


FIGURE 4.9 – L1 norm of relative error (4.35) at different times, with respect to the MR threshold  $\varepsilon$  (left panel), the CPU gain (center panel) and the memory gain (right panel). Effect of adaptive MR threshold  $\varepsilon_t$  and doubling threshold  $\delta$  in the case of the simple 1D test case, 4 MR levels, third order FV scheme. The line labelled  $O(\varepsilon)$  in the left panel graphs indicates the theoretical asymptotic behavior of the error as a function of  $\varepsilon$ , which is a straight line of slope one in logarithmic axes. The complete legend code is described in Table 4.1.



and the third order FV scheme. The results are displayed in the top panels of Figure 4.11; the value of  $\delta$  minimizing the error is shifted from about  $\varepsilon$  in the first order case to about  $2\varepsilon$ .

We then couple the adaptive doubling strategy (4.20) with the adaptation of the MR threshold to the increasing size of the solution. The same error study as previously is performed, for four MR levels and the third order FV scheme. The results, displayed in the bottom panels of Figure 4.11, still exhibit that there is an optimal value for  $\delta$ . It seems to be roughly around  $1.5\varepsilon$ .

After these extensive numerical tests, we are confident that coupling both adaptive thresholds (4.19) and (4.20) enables the FVMR strategy to robustly handle doubling flux conditions of type (4.2). We will adopt in the sequel the value  $\delta = \varepsilon$  which is close to the optimal value in all cases.

## 4.4 Application to a biological model

### 4.4.1 Model and numerical method

In this section we apply the strategy presented above to the numerical simulation of a biological model dedicated to the cell dynamics in ovarian follicular development introduced in Echenim *et al.* [25] and references therein. The model is multiscale to account for the selection process of ovulatory follicles which involves the cellular, follicular and ovarian levels. At the microscopic level, the granulosa cell population is structured according to the cell age (position within the cell cycle) and the cell maturity (level of sensitivity towards hormonal control). In each ovarian follicle, the granulosa cell population is described by a density function whose changes are ruled by conservation laws. The multiscale structure arises from the formulation of a hierarchical control operating on the aging and maturation velocities as well on the source terms of the conservation law; this control depends on the first moment in the maturity variable of the density. Diagrams summarizing the main features of the model are depicted in Figure 4.12. The functional space in cell age  $\times$  cell maturity is subdivided into subdomains corresponding to different cell states : proliferation ( $\Omega_1^p \cup \Omega_2^p$ ) or differentiation ( $\Omega_3$ ), and also with respect to the cell sensitivity towards the hormonal control exerted by the pituitary gland (sensitive phase  $\Omega_1^p \cup \Omega_3$ , insensitive phase  $\Omega_2^p$ ). The mitosis is the endpoint of the cell cycle which is completed when the two daughter cells are separated from each other. In our set-up the mitosis happens at values  $x_s = p$  for  $p = 1, \dots, N_c - 1$ , on the internal boundaries between  $\Omega_2^p$  and  $\Omega_1^{p+1}$ , and between  $\Omega_2^{N_c}$  and  $\Omega_1^1$  when the density reaches the right side of the domain due to the periodic boundary condition at this point. On the intermediate internal boundaries between  $\Omega_1^p$  and  $\Omega_2^p$ , at values  $x_s = p + 1/2$  for  $p = 1, \dots, N_c$ , the flux condition is continuous. The vertical component of the velocity is positive in  $\Omega_1^p$  and consequently all cells eventually enter  $\Omega_3$ . In the differentiation phase  $\Omega_3$ , cells do not proliferate any more and cannot overcome an asymptotical maturity level, corresponding to a null vertical velocity (see [60]). The position of the zero of  $h(\gamma, u)$  (see Eq (4.40)) varies with the control term  $u$  but the properties of function  $h$

## Chapitre 4. Adaptive mesh refinement strategy for a non conservative transport problem

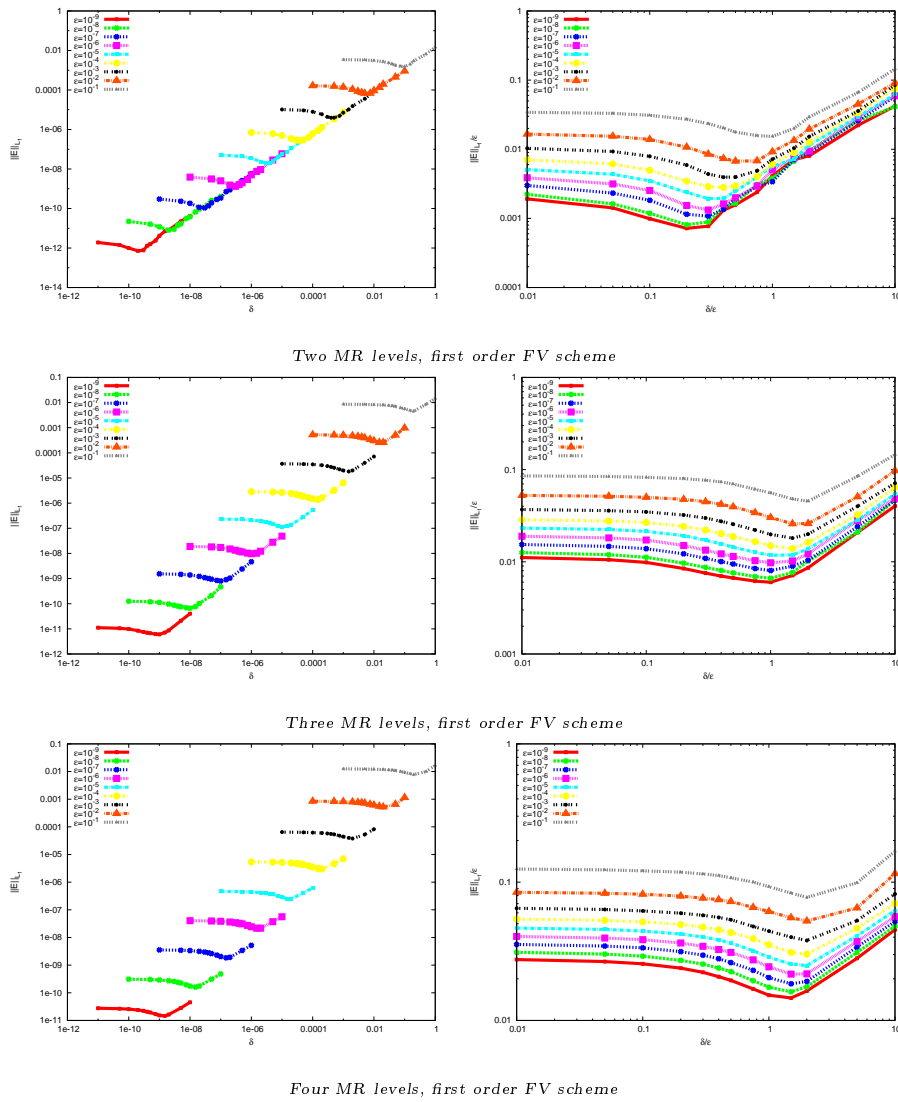


FIGURE 4.10 – Relation between the MR threshold parameter  $\varepsilon$  and the threshold  $\delta$  in the adaptive doubling interface (4.20).  $L_1$  norm of the error with the solution computed on the uniform finest grid at final time. Simple 1D test case (4.1) with first order FV scheme. Left panels : error as a function of  $\delta$ . Right panels : error/ $\varepsilon$  as a function of  $\delta/\varepsilon$ , for different numbers of MR levels (top panels :  $L = 2$ , middle panels :  $L = 3$ , bottom panels :  $L = 4$ ).

## Chapitre 4. Adaptive mesh refinement strategy for a non conservative transport problem

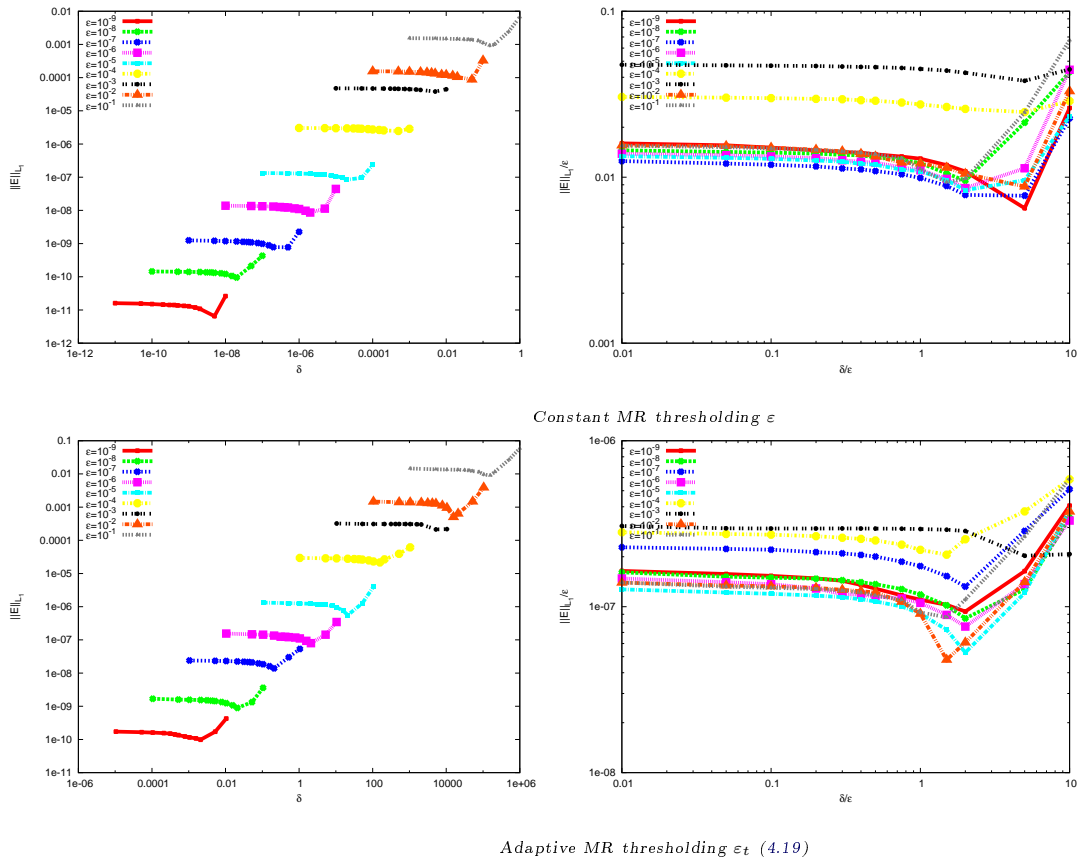


FIGURE 4.11 – Relation between the MR threshold parameter  $\varepsilon$  and the threshold  $\delta$  in the adaptive doubling interface (4.20).  $L_1$  norm of the error with the solution computed on the uniform finest grid at final time. Simple 1D test case (4.1), four MR levels, third order FV scheme. Left panels : error as a function of  $\delta$ . Right panels : error/ $\varepsilon$  as a function of  $\delta/\varepsilon$ . Top panels : constant MR threshold  $\varepsilon$ , bottom panels : adaptive MR threshold  $\varepsilon_t$ .

## Chapitre 4. Adaptive mesh refinement strategy for a non conservative transport problem

---

ensure that it remains in  $\Omega_3$ . We can thus apply periodic boundary conditions to the density –which are easy to implement in our adaptive mesh context– on boundaries  $y = 0$  and  $y = 1$ . We refer the interested reader to [7] for a detailed exposition of the motivation underlying the choice of a localized transmission condition of type (4.2), and to [73, 69] for biological evidence of localized mitotic phenomenon). We also refer the reader to Appendix 4.5 for the description of the model in the single follicle case and to [7] for a more general presentation, including a description of the third order numerical scheme designed to handle the discontinuous flux conditions. Here we solely address the coupling of the MR with the FV scheme in the context of a localized or distributed mitosis. Actually we even neglect at this stage the macroscopic scale where the different follicles are coupled through the interaction between the ovaries and the hypothalamus/pituitary complex, formerly exposed in [6], and only consider the case of a single follicle.

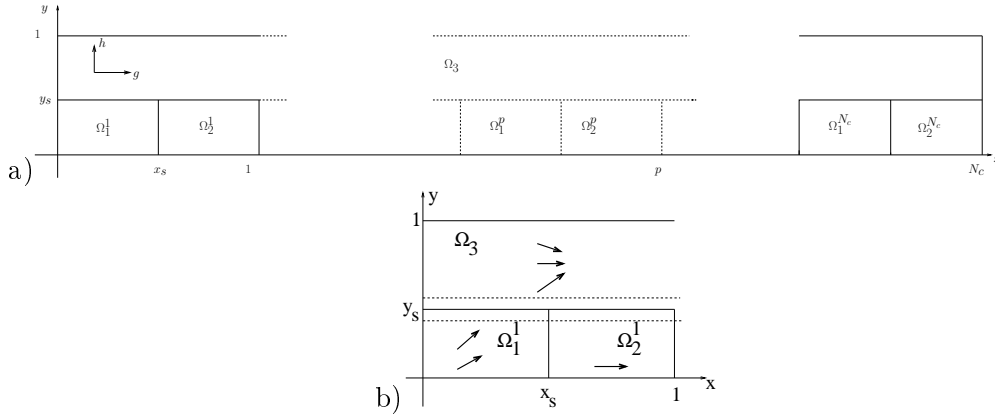


FIGURE 4.12 – *Computational domain for the biological model of follicular development. In panel a) the cell cycles  $\Omega_1^p \cup \Omega_2^p$ ,  $p = 1, \dots, N_c$  of unit width are periodically reproduced.  $\Omega_1^p$  corresponds to phase G1, while  $\Omega_2^p$  aggregates phases S, G2 and M of the  $p$ th cell cycle.  $\Omega_3$  is the differentiation phase, where the cells do not proliferate anymore. Panel b) is a zoom on the first cell cycle. The source term is active in the zone between the horizontal dash lines, where cells are subject to apoptosis. The arrows indicate the possible directions of the transport in the cell phases.*

We use the third order FV scheme designed in [7] and we couple it with the MR strategy on four levels. The domain consists of eight cell cycles with ten meshes in each cycle at the coarsest level in the  $x$  direction, and ten meshes in the  $y$  direction. The CFL number ensuring the stability of the scheme is set to 0.4. Note that the velocity coefficients depend on the solution (see equations (5.2-4.44) in Appendix 4.5). The time step value is therefore time-dependent and computed at each time iteration. We adapt the stability rule designed in [6] to the adaptive case so that

$$\Delta t^n = \min \left( CFL \min_{\mathcal{D}_{\ell,(i,j)} \in \tilde{S}^n} \frac{\Delta x_\ell}{\max(|g_{\ell,(i,j)}^n|, |h_{\ell,(i,j)}^n|)}, \frac{1}{\max_{\mathcal{D}_{\ell,(i,j)} \in \tilde{S}^n} |\Lambda_{\ell,(i,j)}^n|} \right) \quad (4.36)$$

where  $g_{\ell,(i,j)}^n$  (respectively  $h_{\ell,(i,j)}^n$ ) is the discrete value of the speed  $g$  given by

## Chapitre 4. Adaptive mesh refinement strategy for a non conservative transport problem

---

(5.2) (resp.  $h$  given by (4.40)) and  $\Lambda_{\ell,(i,j)}^n$  is the discrete value of the source term  $\Lambda(x, y, U(t))$  given by (4.41) on the mesh  $\mathcal{D}_{\ell,(i,j)} \in \tilde{\mathcal{S}}^n$ . A simpler condition would consist in using the uniform case definition with the space step corresponding to the finest MR level

$$\Delta t^n = \min \left( CFL \frac{\Delta x_{L-1}}{\max_{\mathcal{D}_{\ell,(i,j)} \in \tilde{\mathcal{S}}^n} (|g_{\ell,(i,j)}^n|, |h_{\ell,(i,j)}^n|)}, \frac{1}{\max_{\mathcal{D}_{\ell,(i,j)} \in \tilde{\mathcal{S}}^n} |\Lambda_{\ell,(i,j)}^n|} \right)$$

but since the speed and source coefficients are space dependent, the mesh has to be fully surveyed at each time iteration in any cases and formula (4.36) is more accurate and possibly less stringent. The numerical value of the biological constants appearing in definitions (5.2), (4.40) and (4.44) are gathered in Table 4.2. In the following paragraph we detail a set of numerical experiments which highlight the robustness of our numerical method in this biological set-up. We show in particular, that for this standard set of parameters spurious noise will appear at late times of the simulation unless the adaptive strategies presented earlier are used.

### 4.4.2 Numerical performances

The reference solution is computed on the finest level of resolution, comprising 51200 meshes and it takes 540 seconds on a mono processor Apple laptop to reach the final time  $t = 20$ . Snapshots of the solution at representative times are displayed in Figure 4.13, along with the adaptive mesh corresponding to  $\varepsilon = 0.01$ . Noticeably, at initial time (top panel), due to the periodicity of the MR reconstruction operator (see Eq. (4.22)), the mesh is refined near the right edge of the domain although the density is zero in this area.

As in the simple test case, we perform simulations with or without time adaptation of the MR threshold (4.19) and (4.34), and with or without the adaptive doubling strategy (4.20). Figure 4.14 dedicated to snapshots of the solution at  $t = 20$  shows the influence of adaptive thresholds on the spurious noise removal. A zoom of the density computed with the different methods is displayed in the left panels. Both solutions look very similar although if we let the simulation run longer the solution in the bottom panel eventually deteriorates. In the right panels we display the adaptive grid superimposed to the density with a color code adapted to the amplitude of the spurious noise.

We then compute the L1 norm of the relative error (4.35) as a function of  $\varepsilon$  at different times, corresponding to the snapshots displayed in Figures 4.13 and 4.14, with respect to the solution obtained using the uniform finest grid.

In Figure 4.15 we display the error curves against the MR threshold  $\varepsilon$  (left panel), the CPU gain (center panel) and the memory gain (right panel). The same legend as in Figure 4.9 is used, and described in Table 4.1. As in the simple 1D test case, the asymptotic behavior of the error in  $O(\varepsilon)$  is verified, except for extreme values of the threshold parameter  $\varepsilon$ . In the left panels, the error curves corresponding to simulations with the adaptive doubling strategy (4.20) (with red +, blue \* and cyan

## Chapitre 4. Adaptive mesh refinement strategy for a non conservative transport problem

---

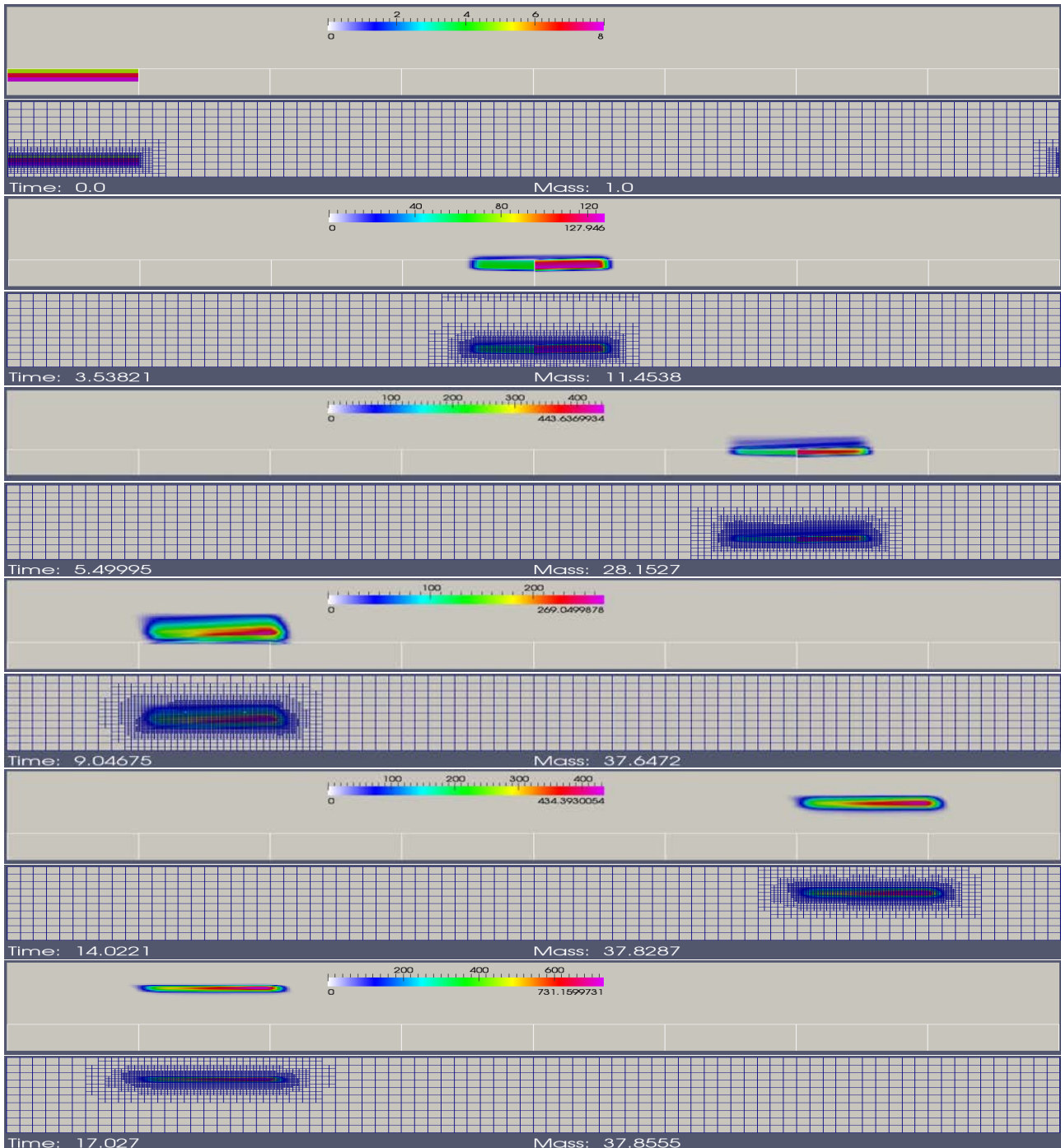


FIGURE 4.13 – Snapshots of the density of granulosa cells computed with the FVMR strategy, adaptive doubling (4.20) and MR threshold (4.19), four MR levels, third order scheme,  $\varepsilon = 10^{-2}$ , and biological constants from Table 4.2. For each time, the adaptive grid is shown below the density, displayed with a time-dependent color code. The boundaries between biological cell cycles are marked with white lines.

## Chapitre 4. Adaptive mesh refinement strategy for a non conservative transport problem

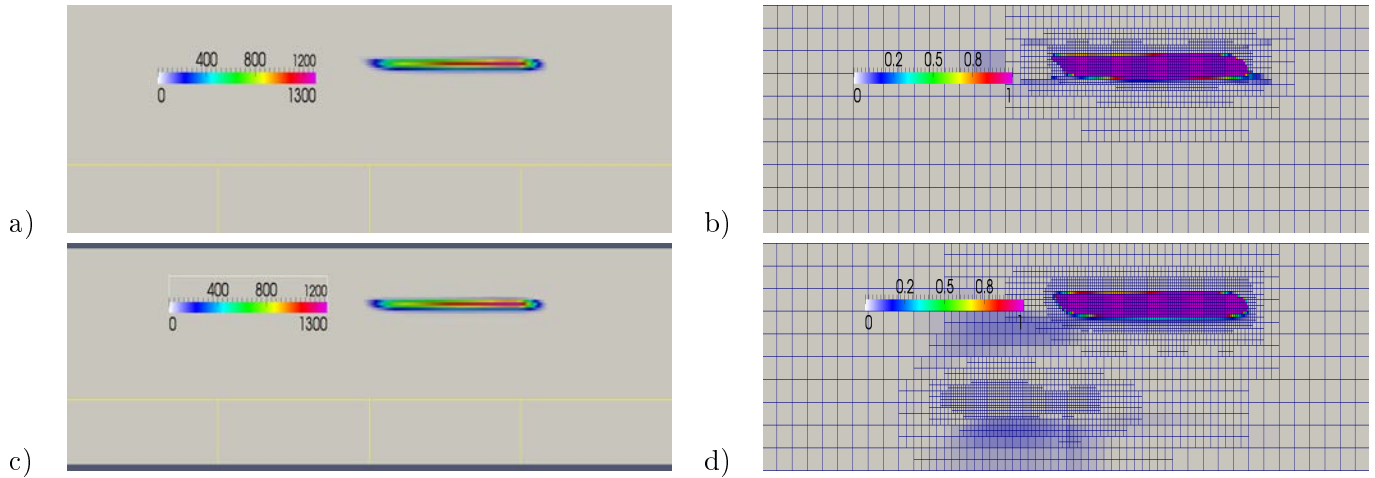


FIGURE 4.14 – Snapshots of the density at final time  $t = 20$ . a-b) with the adaptive doubling (4.20) and MR threshold strategy (4.19), c-d) : unconditional doubling and constant MR threshold. b-d) The color code adapted to the initial condition is used in order to highlight the spurious noise. Without the adaptive doubling strategy, the spurious noise appears and triggers additional refinement (bottom part of panel d)).

■) lie lower than their counterparts without the adaptive doubling strategy (with green ×, pink □ and black ●).

In the center and right panels, the performances of the different algorithms in terms of CPU time and memory requirements are displayed for a given accuracy measured by the relative error (4.35). The differences between the different strategies are not as spectacular as is the simple test case. Due to the differentiation phenomenon and the associated cell cycle exit, the overall proliferation rate is around 37 at the final time  $t = 20$ , much smaller -by a factor of 200- than in the 1D test case, so that the spurious noise is less important. Nonetheless, for late times, the curves corresponding to the simulation without MR nor doubling threshold adaptation (with green ×) clearly exhibit worse performances than the others. It is also noticeable that as time increases, the range of memory gain shifts to the left. Indeed, the size of the adaptive grid increases, because the density spreads out when it crosses the boundary between the proliferation and differentiation phase. The range of CPU gain also shifts accordingly to the left, since the computing time is directly and almost linearly related to the size of the grid.

The relation between the MR threshold  $\varepsilon$  and the adaptive doubling threshold  $\delta$  is studied in Figure 4.16, where the relative error with the uniform grid solution is displayed against  $\delta$  for different values of  $\varepsilon$ . Here again an optimal value is clearly visible. The curves in the right panel, where the product  $\varepsilon E(\delta, \varepsilon)$  is displayed against  $\delta/\varepsilon$  are almost all superimposed, except for the two extreme values of  $\varepsilon$ . This corroborates the asymptotic behavior of the MR error in  $O(\varepsilon)$ , which is always verified except if  $\varepsilon$  is too large ( $= 0.1$ ) or too small (in that case the numerical accuracy of

## Chapitre 4. Adaptive mesh refinement strategy for a non conservative transport problem

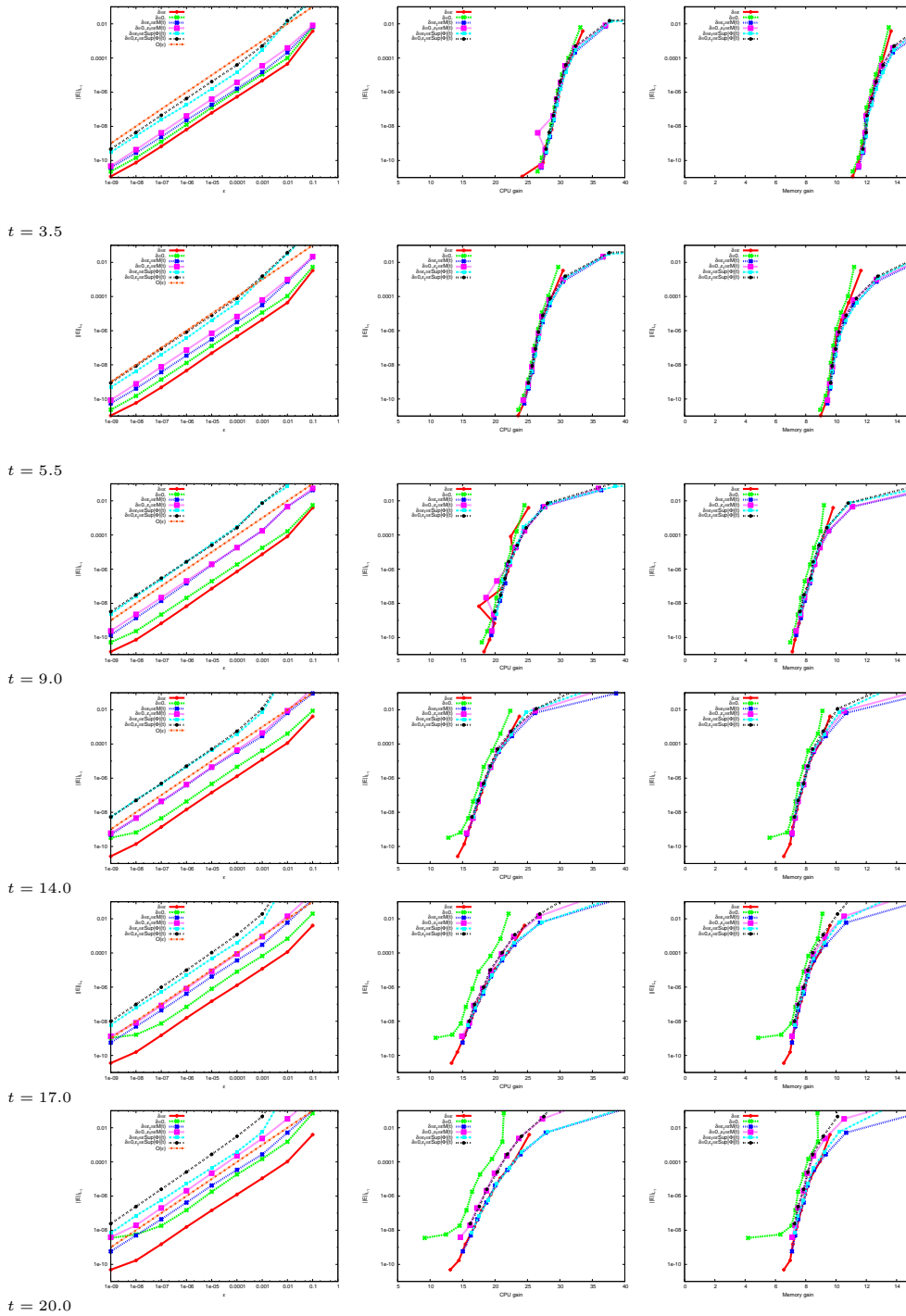


FIGURE 4.15 – L1 norm of relative error (4.35) at different times, with respect to the MR threshold  $\varepsilon$  (left panel), the CPU gain (center panel) and the memory gain (right panel). Effect of adaptive MR threshold  $\varepsilon_t$  and doubling threshold  $\delta$  in the case of the biological model. The line labeled  $O(\varepsilon)$  in the left panels indicates the theoretical asymptotic behavior of the error as a function of  $\varepsilon$  which is a straight line of slope one in logarithmic axes. The complete legend code is described in Table 4.1.



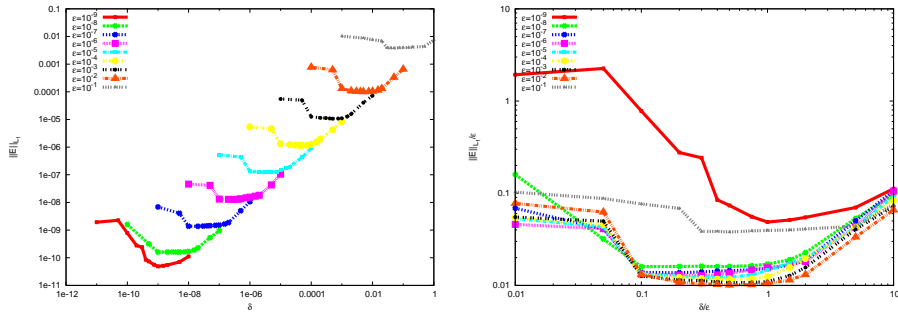


FIGURE 4.16 – Relation between the MR threshold parameter  $\varepsilon$  and the threshold  $\delta$  in the adaptive doubling interface (4.20).  $L_1$  norm of the error with the solution computed on the uniform finest grid at final time. Biological model, four MR levels, third order FV scheme. Error as a function of  $\delta$  (left), error as a function of  $\delta/\varepsilon$ , error/ $\varepsilon$  as a function of  $\delta/\varepsilon$ , for different values of  $\varepsilon$ . Constant MR threshold  $\varepsilon$ .

the computer penalizes the asymptotic behavior).

#### 4.4.3 Comparison of the numerical performances for both models of mitosis

As previously explained in the Introduction and the beginning of Paragraph 4.4, the mitosis is an almost punctual event in time. Nevertheless cell proliferation is commonly modeled by a distributed gain term [64]. In this paragraph we study the influence of treating the mitosis in either a localized or distributed manner on the convergence of the MR algorithm with  $\varepsilon$ . In contrast with the simple model illustrated in Figure 4.7, we are now in a realistic biological context including the 2D dependence of the cell density on age and maturity variables. However, to make the comparison meaningful, the aging velocity is left constant in time and piecewise constant in age and maturity

$$g(x, y, u) = \begin{cases} \gamma_2 = 1.2 & \text{for } (x, y) \in \Theta_1, \\ 1 & \text{for } (x, y) \in \Theta_2 \cup \Omega_3, \end{cases} \quad (4.37)$$

instead of using its general closed-loop model formulation (5.2). We can compute the equivalent distributed linear source term  $B$ , which results in doubling the cell mass after one cell cycle has elapsed

$$B = \frac{2 \ln(2) \gamma_2}{1 + \gamma_2}.$$

In order to reach a realistic mass gain before all cells have left the proliferation stage, we also reduce the time constant of the maturation velocity  $\tau_h = 0.2$ . Figure 4.17 displays the dependence of the relative error between the adaptive and uniform solutions with respect to the threshold  $\varepsilon$ , the CPU and the memory requirement. The left panels in Figure 4.17 show that the theoretical behavior in  $O(\varepsilon)$  is observed

## Chapitre 4. Adaptive mesh refinement strategy for a non conservative transport problem

---

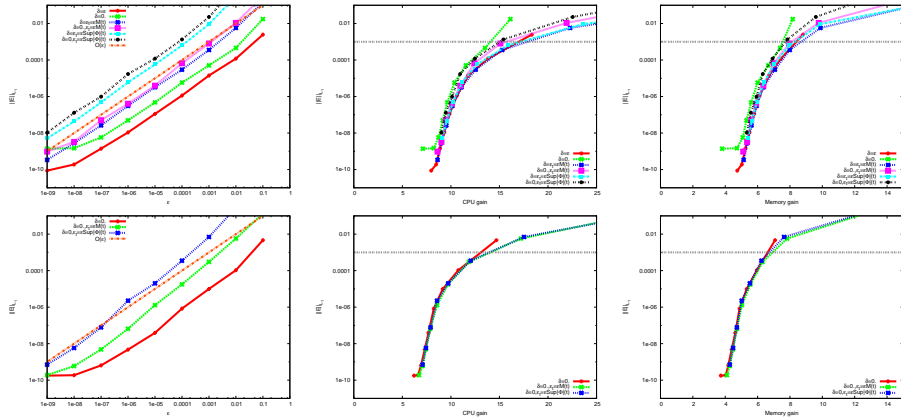


FIGURE 4.17 – L1 norm of relative error (4.35) with respect to the threshold parameter  $\varepsilon$  (left panels), CPU gain (center panels) and memory gain (right panels). Top panels : Localized doubling with uncontrolled aging velocity (4.37), bottom panels : equivalent distributed doubling. The complete legend code is described in Table 4.1.

except for very small values of  $\varepsilon$  where the error estimates reaches the numerical accuracy. The reference solution on the uniform finest grid with 51200 meshes requires 535 seconds of CPU time. The center and right panels show that an average gain of 10 in CPU can be achieved, for a relative error of  $10^{-3}$ , while reducing the memory by a factor of six. Comparison of the top and bottom panels corresponding respectively to the localized and distributed models shows that the adaptive scheme exhibits comparable characteristics in both cases.

## Conclusion

We have adapted a MR procedure to FV in order to simulate numerically non conservative transport equations with specific discontinuous flux conditions on interfaces between subdomains. In the absence of any specific care, we almost systematically observe the appearance of spurious numerical noise in long time simulations.

In this paper we have explained on a simple but generic model the reasons underlying this artifact and proposed a strategy to avoid it. We show that the doubling condition should be applied according to an adaptive rule only when the solution is above some threshold  $\delta$ , to be taken of the order of the MR threshold  $\varepsilon$ . This stops the spurious noise appearance, while preserving all the robust characteristics of the MR method, namely its  $O(\varepsilon)$  asymptotic behavior, and interesting gains in CPU time and memory requirements. Accessorily, we have also qualitatively compared the modeling of biological mitosis by a discontinuous flux condition, to the distributed birth term often encountered in the literature, and shown that the later is also well handled by our FVMR scheme. This method will next be applied to real case simulations involving large dimension vector systems.

## 4.5 Appendix : Description of the model for follicular development

This paragraph presents the details of the biological model illustrated by the numerical simulations commented in Paragraph 4.4. It is meant to make the paper self-contained and introduce the notations. Details on the biological bases of the model can be found in the founding paper [41] and the review [27]. Since our goal here is to design the coupling of MR with adaptive doubling strategy, the vector nature of the model is not relevant, so that we only present its scalar version, referring the reader to [6, 7] for a more general presentation of the numerical scheme. Let us denote  $\phi(t, x, y)$  the density of granulosa cells at time  $t$ , age  $x$  and cell maturity  $y$ . It satisfies the following equation :

$$\frac{\partial \phi(x, y, t)}{\partial t} + \frac{\partial (g(x, y, u(t))\phi(x, y, t))}{\partial x} + \frac{\partial (h(x, y, u(t))\phi(x, y, t))}{\partial y} = -\Lambda(x, y, U(t))\phi(x, y, t), \quad (4.38)$$

set in the computing domain  $\Omega$  in the  $(x, y)$  plane,

$$\Omega = \{(x, y), 0 \leq x \leq N_c \times D_c, 0 \leq y \leq 1\}$$

where  $N_c$  is the number of cell cycles and  $D_c$  is the duration of one cycle. The domain  $\Omega$  is divided in  $2N_c + 1$  zones :  $\Omega_1^p, \Omega_2^p$ , for  $p = 1, \dots, N_c$  and  $\Omega_3$ , corresponding to different cell states illustrated in Figure 4.12 and hence different definitions of the speeds and source terms. Phase  $\Omega_2^p$  in the model aggregates the three latest phases (S, G2, M) of the  $p^{th}$  cell cycle

$$\left\{ \begin{array}{l} \Omega_1^p = \{(x, y) \in \Omega, \quad pD_c \leq x \leq (p+1/2)D_c, \quad 0 \leq y \leq y_s\}, \quad p = 0, \dots, N_c - 1, \\ \Theta_1 = \cup_{p=1}^{N_c} \Omega_1^p \\ \Omega_2^p = \{(x, y) \in \Omega, \quad (p+1/2)D_c \leq x \leq (p+1)D_c, \quad 0 \leq y \leq y_s\}, \\ p = 0, \dots, N_c - 1, \quad \Theta_2 = \cup_{p=1}^{N_c} \Omega_2^p \\ \Omega_3 = \{(x, y) \in \Omega, \quad y_s \leq y\}. \end{array} \right.$$

Each cell cycle consists of the  $\Omega_1^p \cup \Omega_2^p$  subdomain and  $\Theta_i$  for  $i = 1, 2$  denotes the disconnected union of the  $N_c$  corresponding phases  $\Omega_p^i$ , for  $p = 1, \dots, N_c$ .

The aging function  $g$  appearing in (4.38) is defined by

$$g(x, y, u) = \begin{cases} \gamma_1 u + \gamma_2 & \text{for } (x, y) \in \Theta_1 \\ 1 & \text{for } (x, y) \in \Theta_2 \cup \Omega_3 \end{cases} \quad (4.39)$$

where  $\gamma_1, \gamma_2$  are real positive constants.

The maturation function  $h$  is defined by

$$h(x, y, u) = \begin{cases} \tau_h(-y^2 + (c_1 y + c_2)(1 - \exp(\frac{-u}{\bar{u}}))) & \text{for } (x, y) \in \Theta_1 \cup \Omega_3 \\ 0 & \text{for } (x, y) \in \Theta_2 \end{cases} \quad (4.40)$$

## Chapitre 4. Adaptive mesh refinement strategy for a non conservative transport problem

---

where  $\tau_h$ ,  $c_1$ ,  $c_2$  and  $\bar{u}$  are real positive constants.

The source term, that represents cell loss through apoptosis, is defined by

$$\Lambda(x, y, U) = \begin{cases} \bar{\Lambda} \exp\left(-\left(\frac{y - y_s}{\bar{\gamma}}\right)^2\right) (1 - U) \chi_{\{|y - \gamma_s| \leq \Lambda_w\}}(y) & \text{for } (x, y) \in \Theta_1 \cup \Omega_3 \\ 0 & \text{for } (x, y) \in \Theta_2 \end{cases} \quad (4.41)$$

where  $\bar{\Lambda}$ ,  $y_s$  and  $\bar{\gamma}$  are real positive constants, and  $\chi_E(y)$  is the characteristic function of subset  $E$ .

The equations in the PDE system (4.38) are linked together through the argument  $u(t)$  appearing in the speeds  $g(x, y, u)$  and  $h(x, y, u)$  and the argument  $U(t)$  appearing in the source term  $\Lambda(x, y, U)$ .  $U(t)$  and  $u(t)$  represent respectively the plasma FSH level and the locally bioavailable FSH level and depend on the first maturity moment of the density

$$M(t) = \int_0^1 \int_0^{N_c D_c} y \phi(x, y, t) dx dy. \quad (4.42)$$

The plasma FSH level  $U(t)$  showing up in the arguments of the source term in (4.38) is defined by

$$U(t) = U_{\min} + \frac{U_{\max} - U_{\min}}{1 + \exp(c(M(t) - \bar{M}))}, \quad (4.43)$$

where  $U_{\min}$ ,  $c$  and  $\bar{M}$  are real positive constants. The locally bioavailable FSH level  $u(t)$  showing up in the arguments of the speeds in (4.38) is defined by

$$u(t) = \min\left(b_1 + \frac{\exp(b_2 M(t))}{b_3}, 1\right) U(t) \quad (4.44)$$

where  $b_1$ ,  $b_2$  and  $b_3$  are real positive constants.

The precise definition of the required transmission conditions along the successive cell cycles of the domain has been formulated in [71]. For each cycle  $p = 1, \dots, N_c$ ,

- the flux on the  $x$ -axis is continuous on the interface between  $\Omega_1^p$  and  $\Omega_2^p$

$$\phi(t, x^+, y) = (\gamma_1 u + \gamma_2) \phi(t, x^-, y), \quad x = (p - 1/2)D_c, \quad 0 \leq y \leq y_s. \quad (4.45)$$

- The flux is doubling on the interface between  $\Omega_2^p$  and  $\Omega_1^{p+1}$ , which accounts for the birth of two daughter cells from one mother cell at the end of each cell cycle

$$(\gamma_1 u + \gamma_2) \phi(t, x^+, y) = 2\phi(t, x^-, y), \quad x = pD_c, \quad 0 \leq y \leq y_s. \quad (4.46)$$

- A homogeneous Dirichlet condition holds to the north of the interface between  $\Omega_2^p$  and  $\Omega_3$

$$\phi(t, x, y_s^+) = 0, \quad (p - 1/2)D_c \leq x \leq pD_c. \quad (4.47)$$

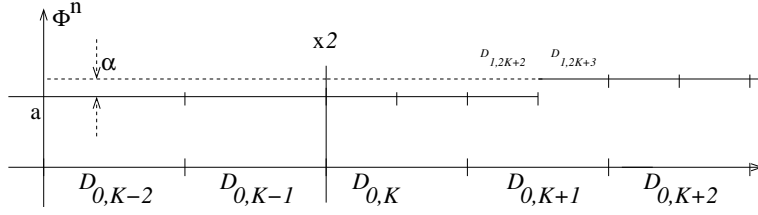


FIGURE 4.18 – Set-up for Appendix 4.6. Three level discretization with doubling interface located between meshes  $\mathcal{D}_{0,K-1}$  and  $\mathcal{D}_{0,K}$  on the coarsest level. At time  $t_n$  the solution is piecewise constant with a discontinuity of size  $\alpha$  located between meshes  $\mathcal{D}_{1,2K+2}$  and  $\mathcal{D}_{1,2K+3}$ , resulting in a gradually adaptive grid.

## 4.6 Appendix : Example of spurious numerical noise appearance

### 4.7 Numerical noise appearance

We show on a simple example that spurious noise can appear due to doubling transitions and requires refinement on the finest model, while the real discontinuities get smoothed out by the numerical scheme diffusion.

Suppose for instance that at time  $t_n$  the solution is locally almost piecewise constant, with a small discontinuity in the mesh  $\mathcal{D}_{0,K+1}$  requiring both this mesh and its immediate neighbors to be subdivided, as in Figure 4.18. To fix ideas, denote by  $\alpha = 4\varepsilon/3$  the size of the discontinuity

$$\begin{aligned} \phi_{0,K-i}^n &= a, \text{ for } i \geq 1 \\ \phi_{1,2K}^n &= \phi_{1,2K+1}^n = \phi_{1,2K+2}^n = a \\ \phi_{1,2K+3}^n &= \phi_{1,2K+4}^n = \phi_{1,2K+5}^n = a + \alpha \\ \phi_{0,K+i}^n &= a + \alpha, \text{ for } i \geq 3 \end{aligned}$$

The details at time  $n$  can be computed

$$\begin{aligned} d_{1,K}^n &= \phi_{1,2K}^n - \left( \frac{1}{8}\phi_{0,K-1}^n + \phi_{0,K}^n - \frac{1}{8}\phi_{0,K+1}^n \right) \\ &= a - \left( \frac{1}{8}a + a - \frac{1}{8}\frac{a+a+\alpha}{2} \right) = \frac{\alpha}{16} \end{aligned}$$

$$\begin{aligned} d_{1,K+1}^n &= \phi_{1,2K+2}^n - \left( \frac{1}{8}\phi_{0,K}^n + \phi_{0,K+1}^n - \frac{1}{8}\phi_{0,K+2}^n \right) \\ &= a - \left( \frac{1}{8}a + \frac{a+a+\alpha}{2} - \frac{1}{8}(a+\alpha) \right) = \frac{3}{8}\alpha \end{aligned}$$

## Chapitre 4. Adaptive mesh refinement strategy for a non conservative transport problem

---

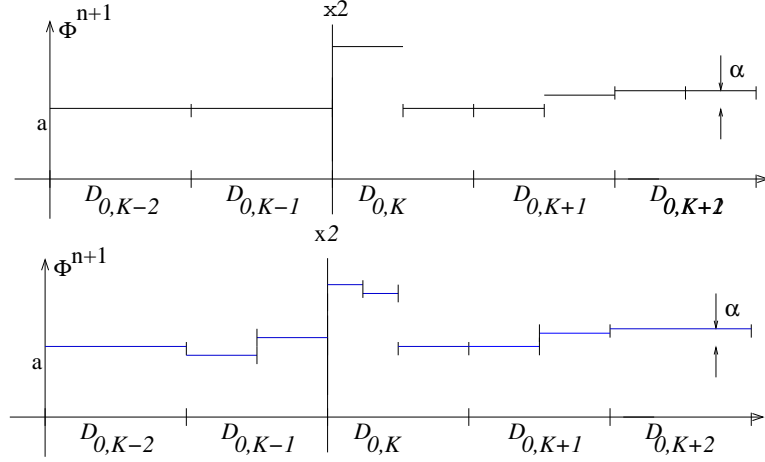


FIGURE 4.19 – Discrete solution in example 4.6. Top panel : updated solution at time  $t_{n+1}$  with formula (4.48). Bottom panel : solution computed with detail thresholding (4.49) according to Algorithms 1 and 2.

$$\begin{aligned}
 d_{1,K+2}^n &= \phi_{1,2K+4}^n - \left( \frac{1}{8}\phi_{0,K+1}^n + \phi_{0,K+2}^n - \frac{1}{8}\phi_{0,K+3}^n \right) \\
 &= a + \alpha - \left( \frac{1}{8} \frac{a + a + \alpha}{2} + a + \alpha - \frac{1}{8}(a + \alpha) \right) = \frac{-\alpha}{16}
 \end{aligned}$$

We have tested these details against the level dependent threshold (4.14) and we have found

$$\begin{aligned}
 |d_{1,K}^n| &= |d_{1,K+2}^n| < \varepsilon_1, \\
 \varepsilon_1 &\leq |d_{1,K+1}^n| < \varepsilon_2,
 \end{aligned}$$

justifying the discretization on the coarsest and intermediate levels. Let us now update the solution at the next time step, using  $\lambda/2 = \Delta t/\Delta x_1$  on the intermediate level and  $\lambda/4 = \Delta t/\Delta x_0$  on the coarsest one (note that to update the solution on mesh  $(1, 2K)$  we will need to reconstruct the solution on its neighbor  $\hat{\phi}_{1,2K-1}^n = a$ ). We obtain the following values, also sketched out in the top panel of Figure 4.19

$$\left\{ \begin{array}{l}
 \phi_{0,K-i}^{n+1} = \phi_{0,K-1}^{n+1} = \phi_{0,K-1}^n - \frac{\lambda}{4}(\phi_{0,K-1}^n - \phi_{0,K-2}^n) = a \\
 \phi_{1,2K}^{n+1} = \phi_{1,2K}^n - \frac{\lambda}{2}(\phi_{1,2K}^n - 2\phi_{1,2K-1}^n) = a + \lambda a \\
 \phi_{1,2K+1}^{n+1} = \phi_{1,2K+1}^n - \frac{\lambda}{2}(\phi_{1,2K+1}^n - \phi_{1,2K}^n) = a \\
 \phi_{1,2K+2}^{n+1} = \phi_{1,2K+2}^n - \frac{\lambda}{2}(\phi_{1,2K+2}^n - \phi_{1,2K+1}^n) = a \\
 \phi_{1,2K+3}^{n+1} = \phi_{1,2K+3}^n - \frac{\lambda}{2}(\phi_{1,2K+3}^n - \phi_{1,2K+2}^n) = a + \alpha(1 - \frac{\lambda}{2}) \\
 \phi_{1,2K+5}^{n+1} = \phi_{1,2K+4}^{n+1} = \phi_{1,2K+4}^n - \frac{\lambda}{2}(\phi_{1,2K+4}^n - \phi_{1,2K+3}^n) = a + \alpha \\
 \phi_{0,K+i}^{n+1} = \phi_{0,K+3}^{n+1} = \phi_{0,K+3}^n - \frac{\lambda}{4}(\phi_{0,K+3}^n - \phi_{0,K+2}^n) = a + \alpha
 \end{array} \right. \quad (4.48)$$

From these updated values we compute the averaged values on the coarsest level

**Chapitre 4. Adaptive mesh refinement strategy for a non conservative transport problem**

---

$$\begin{aligned}\phi_{0,K}^{n+1} &= \frac{a + \lambda a + a}{2} = a + a \frac{\lambda}{2} \\ \phi_{0,K+1}^{n+1} &= \frac{a + a + \alpha(1 - \frac{\lambda}{2})}{2} = a + \frac{\alpha \lambda}{2 \cdot 2} \\ \phi_{0,K+2}^{n+1} &= a + \alpha\end{aligned}$$

and the details at time  $n + 1$

$$\begin{aligned}d_{1,K}^{n+1} &= \phi_{1,2K}^{n+1} - \left( \frac{1}{8} \phi_{0,K-1}^{n+1} + \phi_{0,K}^{n+1} - \frac{1}{8} \phi_{0,K+1}^{n+1} \right) \\ &= a + \lambda a - \left( \frac{1}{8} a + a + a \frac{\lambda}{2} - \frac{1}{8} (a + \frac{\alpha \lambda}{2}) \right) = \frac{\lambda a}{2} + \frac{\alpha \lambda}{32}\end{aligned}$$

$$\begin{aligned}d_{1,K+1}^{n+1} &= \phi_{1,2K+2}^{n+1} - \left( \frac{1}{8} \phi_{0,K}^{n+1} + \phi_{0,K+1}^{n+1} - \frac{1}{8} \phi_{0,K+2}^{n+1} \right) \\ &= a - \left( \frac{1}{8} (a + a \frac{\lambda}{2}) + a + \frac{\alpha \lambda}{2 \cdot 2} - \frac{1}{8} (a + \alpha) \right) = -\frac{a \lambda}{16} - \frac{\alpha \lambda}{4} + \frac{\alpha}{8}\end{aligned}$$

$$\begin{aligned}d_{1,K+2}^{n+1} &= \phi_{1,2K+4}^{n+1} - \left( \frac{1}{8} \phi_{0,K+1}^{n+1} + \phi_{0,K+2}^{n+1} - \frac{1}{8} \phi_{0,K+3}^{n+1} \right) \\ &= a + \alpha - \left( \frac{1}{8} (a + \frac{\alpha \lambda}{2}) + a + \alpha - \frac{1}{8} (a + \alpha) \right) = \frac{-\alpha \lambda}{32} + \frac{\alpha}{8}\end{aligned}$$

If we use  $\lambda = 1/2$  and the value  $\alpha = 4\varepsilon/3$  we obtain

$$\begin{aligned}d_{1,K}^{n+1} &= \frac{-a}{32} \\ d_{1,K+1}^{n+1} &= \frac{a}{4} + \frac{\varepsilon}{48} \\ d_{1,K+2}^{n+1} &= \frac{7\varepsilon}{48}\end{aligned}$$

we see that as soon as

$$4 \frac{47}{48} \varepsilon \leq a < 16\varepsilon$$

for instance say  $a = 4\varepsilon$  we have

$$\begin{cases} \varepsilon_2 \leq |d_{1,K}^{n+1}|, \\ |d_{1,K+1}^{n+1}| < \varepsilon_1, \\ |d_{1,K+2}^{n+1}| < \varepsilon_1, \end{cases} \quad (4.49)$$

meaning that meshes  $\mathcal{D}_{1,2K}$  and  $\mathcal{D}_{1,2K+1}$  have to be subdivided on the finest level while subdivisions  $\mathcal{D}_{0,K+1}$  and  $\mathcal{D}_{0,K+2}$  have to be aggregated again. The resulting reconstructed solution, obtained after applying Algorithms 1 and 2, is sketched out in bottom panel of Figure 4.19. This is of course a theoretical situation but it shows nevertheless how some spurious structure requiring fine meshes may appear in an upstream region and be well detached from the support of the genuine solution.

## Chapitre 4. Adaptive mesh refinement strategy for a non conservative transport problem

---

**Require:** The solution is known by its mean values on the coarsest level

and the details  $\{d_{\ell,k}^n\}_{(\ell,k) \in \Gamma_\varepsilon^n}$  ;

Prediction :

**for** level  $\ell = L - 1 \searrow 1$  **do**

**for**  $k = 0 \nearrow N_0 2^{\ell-1}$  **do**

**if**  $(\ell, k) \in \Gamma_\varepsilon^n$  and  $|d_{\ell,k}^n| \geq \varepsilon_\ell$  **then**

$(\ell, k + i) \in \tilde{\Gamma}_\varepsilon^{n+1}$  for  $|i| \leq 1$

**if**  $\ell < L - 1$  and  $|d_{\ell,k}^n| \geq 2\varepsilon_\ell$  **then**

$(\ell + 1, 2k)$  and  $(\ell + 1, 2k + 1) \in \tilde{\Gamma}_\varepsilon^{n+1}$

**end if**

**end if**

**end for**

**end for**

Gradualness :

**for** level  $\ell = L - 1 \searrow 1$  **do**

**for**  $k = 0 \nearrow N_0 2^{\ell-1}$  **do**

**if**  $(\ell, k) \in \tilde{\Gamma}_\varepsilon^{n+1}$  **then**

**for**  $|i| \leq 1$  **do**

$(\ell - 1, \lfloor k/2 \rfloor + i) \in \tilde{\Gamma}_\varepsilon^{n+1}$

**end for**

**end if**

**end for**

**end for**

**for**  $k = 0 \nearrow N_0 2^{\ell-1}$  **do**

**if**  $(\ell, k) \in \tilde{\Gamma}_\varepsilon^{n+1} \setminus \Gamma_\varepsilon^n$  **then**

$d_{\ell,k}^n = 0$

**end if**

**end for**

**end for**

**Algorithm 1:** Prediction of the set  $\tilde{\Gamma}_\varepsilon^{n+1}$  of significant details at time  $t_{n+1}$  from the details at time  $t_n$ . General case with  $L$  MR levels numbered from the coarsest 0 to the finest  $L - 1$



Chapitre 4. Adaptive mesh refinement strategy for a non conservative transport problem

---

**Require:** The solution is known by its mean values on the coarsest grid, and the details for  $(k, \ell) \in \tilde{\Gamma}_\varepsilon^{n+1}$  ;

**for** level  $\ell = 1 \nearrow L - 1$  **do**

**for**  $k = 0 \nearrow N_0 2^{\ell-1}$  **do**

**if**  $(\ell, k) \in \tilde{\Gamma}_\varepsilon^{n+1}$  **then**

$\{\mathcal{D}_{\ell,2k}, \mathcal{D}_{\ell,2k+1}\} \in \tilde{S}^{n+1}$  and  $\mathcal{D}_{\ell-1,k} \notin \tilde{S}^{n+1}$

      compute  $\phi_{\ell,2k}^n = \hat{\phi}_{\ell,2k}^n + d_{\ell,k}^n$

      and  $\phi_{\ell,2k+1}^n = \hat{\phi}_{\ell,2k+1}^n - d_{\ell,k}^n$

**end if**

**end for**

**end for**

**Algorithm 2:** Construction of the solution on the adaptive grid  $\tilde{S}^{n+1}$  from the set of significant details  $\tilde{\Gamma}_\varepsilon^{n+1}$

	Unconditional doubling (4.2)		Adaptive doubling (4.20)	
Constant MR threshold	$\delta = 0$	×	$\delta = \varepsilon$	+
Adaptive MR threshold (4.19)	$\delta = 0, \varepsilon_t = \varepsilon M(t)$	■	$\delta = \varepsilon_t = \varepsilon M(t)$	*
Adaptive MR threshold (4.34)	$\delta = 0, \varepsilon_t = \varepsilon \text{Sup} \Phi (t)$	●	$\delta = \varepsilon_t = \varepsilon \text{Sup} \Phi (t)$	■

TABLE 4.1 – Legend codes for Figures 4.9 and 4.15

**Chapitre 4. Adaptive mesh refinement strategy for a non conservative transport problem**

---

Parameter	Description	Value
$U_{\min}$	FSH plasma level (eq. (5.1)) minimum level	0.075
$U_{\max}$	minimum level	0.15
$c$	slope parameter	2.0
$\bar{M}$	abscissa of the inflection point	4.5
$\bar{\Lambda}$	Apoptosis source term (eq. (4.41)) intensity factor	1
$\Lambda_w$	half width	0.05
$\bar{\gamma}$	scaling factor	0.1
$y_s$	cellular maturity threshold	0.3
$b_1$	intrafollicular FSH level (eq. (4.44)) basal level	0.08
$b_2$	exponential rate	2.2482
$b_3$	scaling factor	1450.
$\gamma_1$	Aging function (eq. (5.2)) rate	0.2
$\gamma_2$	origin	1.0
$c_1$	Maturation function (eq. (4.40))	0.68
$c_2$		0.08
$\bar{u}$		0.02
$\tau_h$		0.3

TABLE 4.2 – Values of the parameters for the biological model simulations



# Calibration

## Sommaire

<b>5.1 Degrés de liberté</b> . . . . .	<b>122</b>
5.1.1 Conditions initiales . . . . .	122
5.1.2 Fonctions de vitesses . . . . .	123
5.1.3 Variables de contrôle . . . . .	127
<b>5.2 Spécifications biologiques</b> . . . . .	<b>134</b>
5.2.1 Mesures disponibles . . . . .	134
5.2.2 Trajectoires folliculaires . . . . .	136
<b>5.3 Aides à la calibration</b> . . . . .	<b>136</b>
5.3.1 Contraintes a priori . . . . .	136
5.3.2 Visualisation à une échelle microscopique . . . . .	137
5.3.3 Diagnostic a posteriori . . . . .	141
<b>5.4 Méthode de calibration pour un follicule</b> . . . . .	<b>146</b>
5.4.1 Détermination de la maturité asymptotique . . . . .	146
5.4.2 Détermination des paramètres affectant la dynamique . . . . .	149
5.4.3 Exemple de situation de référence à 1 follicule . . . . .	150
<b>5.5 Méthode de calibration pour une cohorte de follicules</b> . . . . .	<b>153</b>
5.5.1 Réglage de la pression sélective . . . . .	154
5.5.2 Abaques multi follicules . . . . .	155
5.5.3 Exemple de situation de référence à 10 follicules . . . . .	157

**Introduction.** Ce chapitre est consacré à la calibration et au contrôle des sorties du modèle. Par calibration, nous entendons la détermination de valeurs des paramètres compatibles avec un cahier des charges énonçant une liste de spécifications de nature qualitative, semi-quantitative ou quantitative.

Une des spécificités de ce modèle est que **l'on a des résultats de mesure à l'échelle macroscopique ou mésoscopique, que l'on cherche à respecter en réglant des paramètres à l'échelle microscopique**. Cette caractéristique est commune à d'autres modèles multi-échelles, et les méthodes décrites dans ce chapitre pourraient se généraliser. Afin de fixer les idées, rappelons qu'il y a 12 paramètres locaux (à multiplier par le nombre de follicules considérés, soit en pratique 20), et 10 paramètres globaux (communs à tous les follicules).

Afin de réduire l'espace des paramètres, notre approche combine une compréhension fine des termes entrant dans la formulation du modèle (e.g. propriétés des fonctions

de vitesse), qui permet dans une certaine mesure, en la couplant aux connaissances biologiques, d'énoncer des contraintes sur les paramètres ou des combinaisons de paramètres, et l'utilisation de méthodes de tir et de visualisation adaptées. On extrait un maximum d'information mathématique de ces dernières.

Dans une première section, on présente les différents degrés de liberté faisant intervenir les paramètres du modèle. Ensuite, on présente les spécifications biologiques du modèle. L'objectif est de caractériser des situations, qui permettent de valider ou d'infirmer une simulation numérique. Pour cela, on rappelle les principales mesures biologiques actuellement à notre disposition, puis on définit des trajectoires folliculaires.

Dans la troisième partie nous combinons les résultats des précédentes sections pour définir des contraintes : contraintes a priori sur les paramètres, contraintes a posteriori sur les sorties du modèle.

Dans une quatrième partie, nous présentons une méthode permettant de calibrer le modèle dans le cas d'un seul follicule, étape préliminaire à la calibration d'une cohorte.

La dernière partie est consacrée à la calibration d'une cohorte, et repose sur les résultats obtenus en troisième partie. Nous montrons comment régler la pression sélective, comment introduire (à bon escient) de l'hétérogénéité, tout en s'assurant de la non prédestination des follicules.

## 5.1 Degrés de liberté

### 5.1.1 Conditions initiales

Les conditions initiales fixent les valeurs prises par les densités de cellules au sein de chaque follicule au temps 0 de la simulation. Ce sont des fonctions localement constantes par morceaux, de la forme :

$$\phi_0(a, \gamma) = \begin{cases} \mu_1 \frac{c_0}{1 - \exp(-c_0)} \exp(-c_0 a) & \text{pour } 0 < a < 1 \text{ et } 0.10 < \gamma < 0.15, \\ \mu_2 \frac{c_0}{1 - \exp(-c_0)} \exp(-c_0 a) & \text{pour } 0 < a < 1 \text{ et } 0.15 < \gamma < 0.20, \\ \mu_3 \frac{c_0}{1 - \exp(-c_0)} \exp(-c_0 a) & \text{pour } 0 < a < 1 \text{ et } 0.20 < \gamma < 0.25, \\ 0 & \text{sinon.} \end{cases}$$

On donne un exemple d'une telle fonction sur la Figure 5.1 avec  $\mu_1 = 8$ ,  $\mu_2 = 7$  et  $\mu_3 = 5$ . Ce type de condition initiale correspond à une population entièrement proliférante, avec une distribution asynchrone des cellules dans le cycle cellulaire et une répartition constante par morceaux en maturité. Comme détaillé par exemple dans [78] dans le cas d'un modèle compartimental ODE, il y a dans ce cas exactement deux fois plus de cellules qui viennent de naître que de cellules qui vont se diviser

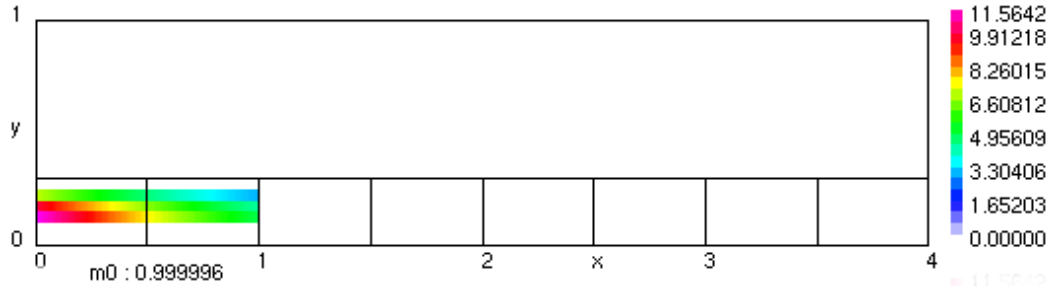


FIGURE 5.1 – Exemple de fonction de condition initiale pour un follicule sur un domaine avec 4 cycles cellulaires. En abscisse, l'âge (position dans le cycle cellulaire), en ordonnée, la maturité cellulaire. La fonction est bivariée, localement constante par morceaux en maturité  $\gamma$  ( $y$  sur la figure), et de distribution exponentielle en âge ( $x$  sur la figure).

( $A(0) = 2A(T_c)$  avec  $T_c$  la durée de cycle). Le respect de cette contrainte s'écrit, dans notre cas :

$$(g_1 u(t=0) + g_2) \phi_0(a=0^+, \gamma) = 2\phi_0(a=1, \gamma)$$

Autrement dit

$$c_0 = \ln\left(\frac{2}{(g_1 u(t=0) + g_2)}\right)$$

### 5.1.2 Fonctions de vitesses

**Fonction  $g_f$ .** La fonction  $g_f$  est la fonction de vitesse de vieillissement (ou progression dans le cycle cellulaire, le long de l'axe des abscisses), définie par

$$g_f(a, \gamma, u_f) = \begin{cases} g_1 u_f + g_2 & \text{en } G1, \\ 1 & \text{en } SM \cup D. \end{cases}$$

C'est une fonction de trois variables réelles : les deux variables d'espace  $a$  et  $\gamma$ , et une variable de contrôle, active ou non, selon le sous domaine. Sur la Figure 5.2, on représente une famille de fonctions  $g_f$  pour un  $f$  donné, indexée par des valeurs de contrôles distinctes, pour toute valeur de  $\gamma$  prise dans le segment  $[0, \gamma_s]$  (coupe horizontale en zone de prolifération SM-G1). On note que l'amplitude de cette fonction dépend linéairement du contrôle.

**Propriété de la fonction  $g_f$ .** Si  $u_1 \leq u_2$ , alors pour tout couple  $(a, \gamma)$  on a

$$g_f(a, \gamma, u_1) \leq g_f(a, \gamma, u_2).$$

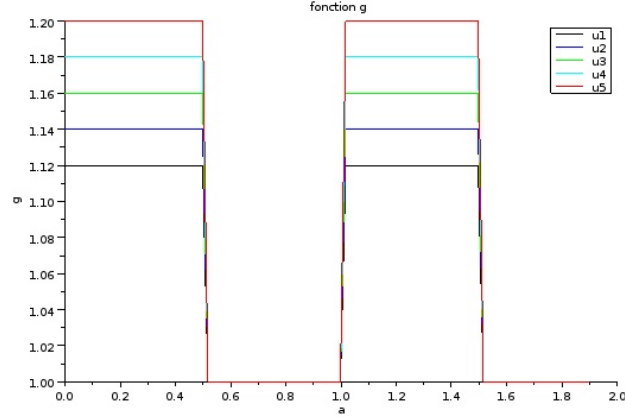


FIGURE 5.2 – Exemple de fonction de vitesse de vieillissement (progression selon l’axe des abscisses). En abscisse, l’âge cellulaire. En ordonnée, la vitesse de vieillissement. La valeur de maturité (axe des ordonnées) est fixée à  $\gamma = 0$  car la fonction de vieillissement est constante en  $\gamma$  pour  $\gamma \in [0, \gamma_s]$ . On représente plusieurs courbes, chacune étant associée à une valeur de variable de contrôle  $u_f$ . La fonction de vieillissement est une fonction croissante du contrôle.

**Fonction  $h_f$  : vitesse de maturation.** La fonction de vitesse en ordonnée  $h_f$  est une fonction de trois variables réelles : les deux variables d’espace  $a$  et  $\gamma$ , et une variable de contrôle  $u_f$ , définie par

$$h_f(a, \gamma, u_f) = \begin{cases} \tau_f(-\gamma^2 + (c_1\gamma + c_2)(1 - e^{(-\frac{u_f}{a})})) & \text{en } G1 \cup D, \\ 0 & \text{en } SM. \end{cases}$$

Sur la Figure 5.3, on représente une famille de fonctions  $h_f$ , pour un  $f$  donné, à  $a$  fixé en G1 (coupe verticale) en fonction de  $\gamma$ , indexée par des valeurs de contrôles distincts.

**Zéros de la fonction  $h_f$ .** La fonction  $h_f$  est un polynôme de degré 2 en  $\gamma$ , dont les racines sont fonctions de  $u_f$ . On note que ces racines sont indépendantes du paramètre  $\tau_f$ . Cette fonction a alors les mêmes racines que le polynôme

$$P(\gamma) = -\gamma^2 + (c_1\gamma + c_2)\theta(u_f),$$

avec  $\theta(u_f) = (1 - e^{(-\frac{u_f}{a})})$ , à contrôle  $u_f$  fixé. Nous pouvons calculer le discriminant

$$\Delta = c_1^2\theta(u_f)^2 + 4c_2\theta(u_f),$$

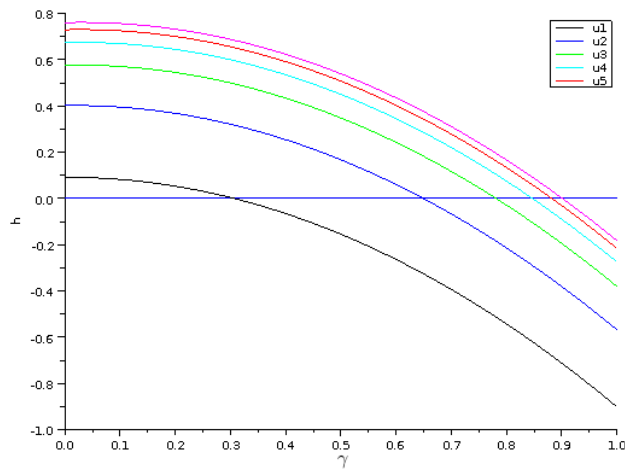


FIGURE 5.3 – Exemple de fonction de vitesse de maturation (progression selon l'axe des ordonnées). En abscisse, la maturité cellulaire, en ordonnée la vitesse de maturation, pour l'abscisse  $a = 0$  fixée. On a représenté l'axe  $h = 0$ , pour bien identifier les zéros de  $h_f$  dans le demi-plan positif. On a tracé la fonction  $h_f(a, \gamma, u_f)$  pour différentes valeurs du contrôle mésoscopique  $u_f$ , les paramètres  $c_1, c_2, \bar{u}$  étant fixés.  $h_f$  est une fonction constante de  $u_f$  (mais bornée), si bien que l'écart entre les graphes de  $h_f$  pour différentes valeurs de  $u_f$  diminue quand  $u_f$  augmente.



## Chapitre 5. Calibration

---

avec  $\theta(u_f) = (1 - e^{(-\frac{u_f}{\bar{u}})})$ , puis expliciter les deux racines réelles

$$\begin{cases} r^- = \frac{-\sqrt{c_1^2\theta(u_f)^2 + 4c_2\theta(u_f) + c_1\theta(u_f)}}{2}, \\ r^+ = \frac{+\sqrt{c_1^2\theta(u_f)^2 + 4c_2\theta(u_f) + c_1\theta(u_f)}}{2}, \end{cases}$$

sachant que seule la racine positive a un intérêt dans notre contexte.

**Monotonie de  $r^+$  par rapport au contrôle.** La position des racines de  $h_f$  est monotone par rapport au contrôle  $u_f$ .

Preuve : Supposons que

$$u_1 \leq u_2.$$

Alors, par décroissance de la fonction  $u \rightarrow \exp(-u)$  et par composition, on obtient

$$\theta(u_1) = 1 - \exp\left(-\frac{u_1}{\bar{u}}\right) \leq 1 - \exp\left(-\frac{u_2}{\bar{u}}\right) = \theta(u_2),$$

et par conséquent

$$r^+(u_1) = \frac{+\sqrt{c_1^2\theta(u_1)^2 + 4c_2\theta(u_1) + c_1\theta(u_1)}}{2} \leq \frac{+\sqrt{c_1^2\theta(u_2)^2 + 4c_2\theta(u_2) + c_1\theta(u_2)}}{2} = r^+(u_2).$$

Conclusion,  $u_1 \leq u_2$  implique  $r^+(u_1) \leq r^+(u_2)$ .

**Contrainte a priori sur la racine positive  $r^+$ .** Étant donné deux réels  $U_{\max}$  et  $\bar{u}_{\min}$  (représentant des variables de contrôle), à contrôle  $u_f$  fixé, on a l'encadrement

$$\gamma_s < r^+ < 1,$$

si on a les inégalités

$$\begin{cases} \frac{\gamma_s^2}{c_1\gamma_s + c_2} < \theta(\bar{u}_{\min}), \\ \theta(U_{\max}) < \frac{1}{c_1 + c_2}, \\ \bar{u}_{\min} \leq u_f \leq U_{\max}. \end{cases}$$

Preuve. Supposons que

$$r^+ < 1,$$

Alors on peut écrire, en revenant à la définition de  $r^+$ ,

$$\sqrt{\Delta} + c_1\theta(u_f) < 2,$$

puis passer au carré

$$\Delta < 4 + c_1^2\theta(u_f)^2 - 4c_1\theta(u_f),$$

et simplifier l'expression pour obtenir

$$\theta(u_f)(c_1 + c_2) < 1,$$

## Chapitre 5. Calibration

---

et ainsi obtenir la première inégalité

$$\theta(u_f) < \frac{1}{c_1 + c_2}.$$

Supposons à présent que

$$\gamma_s < r^+.$$

On peut alors écrire, toujours en revenant à la définition

$$2\gamma_s - c_1\theta(u_f) < \sqrt{(\Delta)},$$

puis en passant au carré et en simplifiant, comme précédemment,

$$\gamma_s^2 < \theta(u_f)(c_1\theta + c_2),$$

On conclut alors que

$$\frac{\gamma_s^2}{c_1\gamma_s + c_2} < \theta(u_f).$$

**Notations.** A présent, par souci de simplicité, nous noterons

$$\begin{cases} \gamma_0^{\min} = r^+(\bar{u}_{\min}), \\ \gamma_0^{\max} = r^+(U_{\max}). \end{cases}$$

**Comportement asymptotique.** Rappelons un résultat concernant le comportement asymptotique des solutions, démontré dans [60]. L'annulation de la vitesse de progression en ordonnée implique qu'en temps long, et à contrôle constant, toute densité tend à se concentrer autour d'une maturité asymptotique, définie par le zéro supérieur de la fonction  $h_f$ . On illustre ce phénomène sur la Figure 5.4. On comprend alors l'intérêt d'utiliser cette propriété pour contrôler le résultat d'une simulation.

### 5.1.3 Variables de contrôle

**Fonction  $U$  : contrôle macroscopique.** On définit le contrôle macroscopique par

$$U(t) = S(M_1(t)) + U_0(t),$$

avec  $S$  la fonction sigmoïde définie par :

$$S(M_1) = U_{\min} + \frac{U_{\max} - U_{\min}}{1 + \exp(c(M_1 - \bar{M}))}. \quad (5.1)$$

Sur la Figure 5.5 on a représenté une telle sigmoïde. On note qu'il y a deux valeurs asymptotiques reliées par une zone de transition.

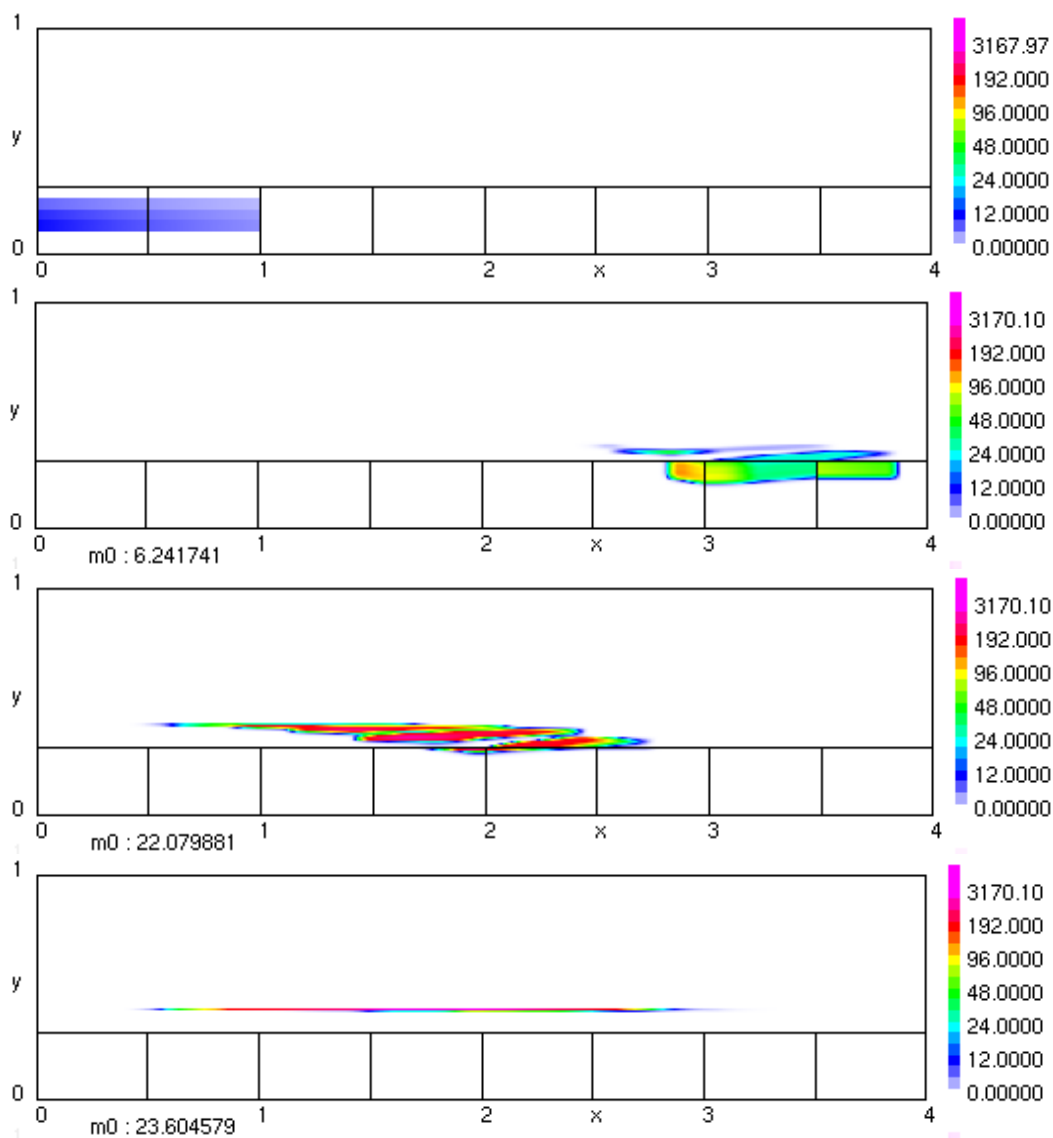


FIGURE 5.4 – *Comportement asymptotique des solutions. Instantanés de la densité aux temps  $t = 0, 2, 4, 8$ . La densité de cellules, initialement positionnée dans la partie inférieure gauche du domaine (cycle 1), se déplace au cours du temps vers la droite (vieillesse) et vers le haut (maturation), pour se concentrer en temps long autour d'une maturité asymptotique (ici  $\gamma = 0.4$ ), qui est le zéro positif de la fonction  $h_f$  correspondant à la valeur asymptotique du contrôle  $u_f$  en temps long. On note que dans cet exemple, du fait des conditions aux limites périodiques, la densité a parcouru plus d'une fois tout le domaine de gauche à droite.*

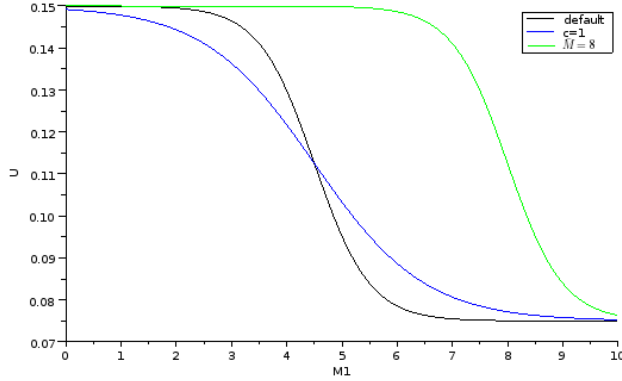


FIGURE 5.5 – Exemple de fonction déterminant la variable de contrôle macroscopique. En abscisse, la maturité globale (ou maturité ovarienne), en ordonnée, la valeur du contrôle macroscopique (ou concentration plasmatique de FSH). C'est une fonction sigmoïde monotone décroissante qui possède un seuil (dit seuil de maturité globale), correspondant au maximum en valeur absolue de la dérivée seconde en temps.

**Point d'inflexion et pente de la fonction  $U$ .** Considérons la fonction  $S$  :

$$S(M_1) = U_{\min} + \frac{U_{\max} - U_{\min}}{1 + \exp(c(M_1 - \bar{M}))}.$$

Posons  $\tilde{S}(M) = \frac{1}{1 + \exp(c(M_1 - \bar{M}))}$ . Calculons la dérivée de  $\tilde{S}$  (en  $M_1$ )

$$\tilde{S}'(M_1) = \frac{-c \exp(c(M_1 - \bar{M}))}{(1 + \exp(c(M_1 - \bar{M})))^2} = -c\tilde{S}(M_1) \frac{\exp(c(M_1 - \bar{M}))}{1 + \exp(c(M_1 - \bar{M}))}.$$

Or, on peut écrire

$$\frac{\exp(c(M_1 - \bar{M}))}{1 + \exp(c(M_1 - \bar{M}))} = \frac{\exp(c(M_1 - \bar{M})) + 1}{1 + \exp(c(M_1 - \bar{M}))} - \frac{1}{1 + \exp(c(M_1 - \bar{M}))}.$$

Par conséquent,

$$\tilde{S}'(M_1) = -c\tilde{S}(M_1)(1 - \tilde{S}(M_1)).$$

Conclusion

$$S'(M_1) = (U_{\max} - U_{\min})\tilde{S}'(M_1) = -c(U_{\max} - U_{\min})\tilde{S}(M_1)(1 - \tilde{S}(M_1)).$$

Calculons à présent la dérivée seconde

$$\tilde{S}''(M_1) = -c(\tilde{S}'(M_1)(1 - \tilde{S}(M_1)) - \tilde{S}(M_1)\tilde{S}'(M_1)).$$

En réinjectant l'expression de la dérivée, on obtient

$$\tilde{S}''(M_1) = -c(-c\tilde{S}(M_1)(1 - \tilde{S}(M_1))^2 + c\tilde{S}(M_1)^2(1 - \tilde{S}(M_1))),$$

autrement dit

$$\tilde{S}''(M_1) = c^2(\tilde{S}(M_1)(1 - \tilde{S})^2 - \tilde{S}(M_1)^2(1 - \tilde{S}(M_1))).$$

puis, en factorisant, et en utilisant l'expression algébrique suivante

$$a(1 - a)^2 - a^2(1 - a) = a(1 - a)(1 - a - a) = a(1 - a)(1 - 2a),$$

on obtient

$$\tilde{S}''(M_1) = c^2\tilde{S}(M_1)(1 - \tilde{S}(M_1)(1 - 2\tilde{S}(M_1))).$$

On note que

$$\tilde{S}(\bar{M}) = \frac{1}{2},$$

et en particulier  $\tilde{S}''(\bar{M}) = 0$ . Autrement dit, l'abscisse du point d'inflexion de  $\tilde{S}$  se situe au point  $M = \bar{M}$ , et l'ordonnée de  $S(\bar{M})$  est  $\frac{U_{\max} - U_{\min}}{2}$  (résultat bien connu d'ailleurs).

Conclusion,  $S''(M_1) = (U_{\max} - U_{\min})\tilde{S}''(M_1)$ . On note alors que

$$\begin{cases} S''(m) = 0, \\ S'(m) = -(U_{\max} - U_{\min})\frac{c}{2}. \end{cases}$$

donc le paramètre  $c$  règle la valeur de la pente de la sigmoïde.

**Fonction  $b_f$  : contrôle mésoscopique.** La variable de contrôle mésoscopique en un follicule  $f$  donné est donnée par

$$u_f(t) = b_f(m_1^f(t))U(t),$$

avec

$$b_f(m) = \min\left(b_1 + \frac{\exp(b_2 m)}{b_3}, 1\right),$$

fonction de biodisponibilité locale.

La Figure 5.6 donne un exemple d'une telle fonction. Qualitativement, le comportement est exponentiel (par construction) jusqu'à atteinte d'une valeur limite, et la fonction devient alors constante.

**Propriété de la fonction  $b_f$ .** Si  $b_f < 1$  alors

$$\frac{db_f}{dm}(m) = \frac{b_2}{b_3} \exp(b_2 m),$$

donc les paramètres  $b_2, b_3$  règlent la dérivée de la fonction  $b_f$ . (voir Figure 5.6, où l'on voit l'impact de différents choix de  $b_2$ ).

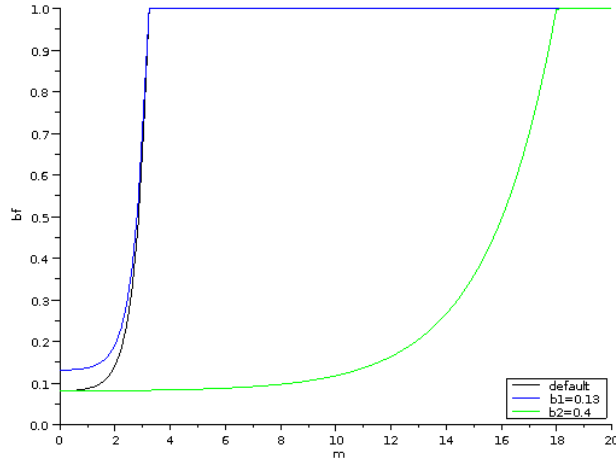


FIGURE 5.6 – Exemple de fonction déterminant la valeur des contrôles mésoscopique. En abscisse, la maturité folliculaire, en ordonnée, la valeur de  $b_f$ , fonction de biodisponibilité locale. On note qu'elle comporte deux parties, une partie exponentielle au départ, puis à l'atteinte de la valeur 1, cette fonction devient constante. Autrement dit, la valeur du contrôle mésoscopique ne peut excéder celle de la valeur du contrôle macroscopique.

**Description de la fonction  $\Lambda$ .** Le taux de perte  $\Lambda(a, \gamma, U)$  en facteur de la densité dans le terme source est une composition de différentes fonctions : une gaussienne centrée en  $\gamma_s$ , une indicatrice  $1_{[\gamma_s^-, \gamma_s^+]}$  avec  $\gamma_s^-$  et  $\gamma_s^+$  des constantes réelles positives telles que  $0 < \gamma_s^- < \gamma_s < \gamma_s^+ < 1$ , et une fonction dépendant linéairement du contrôle macroscopique :

$$\Lambda(a, \gamma, U) = \begin{cases} K \exp\left(-\frac{(\gamma - \gamma_s)^2}{\bar{\gamma}}\right) (1 - U) 1_{[\gamma_s^-, \gamma_s^+]} & \text{en } G1 \cup D, \\ 0 & \text{en } SM. \end{cases}$$

La gaussienne permet de rendre la source maximale au passage de l'interface  $\gamma_s$ , l'indicatrice assure la nullité en dehors d'un certain intervalle, et le terme dépendant de la variable de contrôle macroscopique permet d'annuler la source à contrôle maximal, et de maximiser la source à contrôle minimal. La Figure 5.7 donne l'exemple d'une telle fonction.

**Description de l'ovulation :** Deux paramètres servent à décrire l'issue du processus de sélection, qui aboutit à l'ovulation.

- Le seuil de maturité ovarienne  $M_o$  définit de manière implicite l'instant de déclenchement de la décharge ovulatoire

$$t_f = \min\{t, M_1(t) \geq M_o\}$$

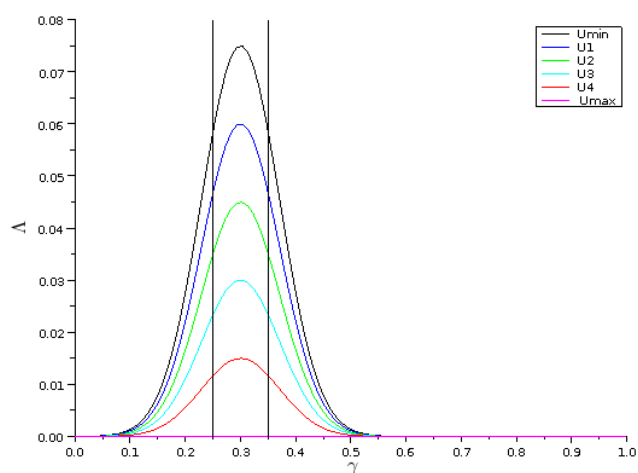


FIGURE 5.7 – Exemple de fonction de perte cellulaire (apoptose). En abscisse, la maturité cellulaire, en ordonnée, la valeur du taux de perte  $\Lambda(a, \gamma, U)$  pour différentes valeurs du contrôle global  $U$ . Cette fonction est constante en âge  $a$  (axe des abscisses), sauf en zone SM où elle est nulle. On note qu'elle est maximale en  $\gamma = \gamma_s$  (ici  $\gamma_s = 0.3$ ), au passage de l'interface entre zone de prolifération et zone différenciée, et qu'elle est nulle en dehors d'un support compact strictement inclus dans le domaine en zones G1 et D, et identiquement nulle en SM.

## Chapitre 5. Calibration

– Le seuil de maturité folliculaire  $M_f$  définit le caractère ovulant d'un follicule si

$$m_f(t_o) \geq M_f$$

**Compléments : Paramètres de simulation numérique.** D'autres paramètres n'interviennent que dans la simulation numérique du modèle par la méthode de volumes finis adaptatifs, présentée dans les chapitres 2, 3, 4 et 6. Il s'agit du nombre de mailles volumes finis par demi-cycle cellulaire  $N_m$ , du nombre de cycles cellulaires,  $N_c$ , de la position de la limite entre la zone de prolifération et la zone de différenciation,  $\gamma_s$ . Le nombre  $CFL$  qui règle la stabilité des calculs, doit être strictement compris entre 0 et 0.5, pour les schémas sélectionnés. Enfin, la grille étant adaptative, on règle deux paramètres supplémentaires : le nombre de niveaux de grilles  $L$ , et une valeur de seuil  $\epsilon$ . Rappelons que la dimension du domaine est fixe et normalisée de sorte que la longueur d'un cycle est 1 et  $\gamma_{\max} = 1$ . On a représenté sur la Figure 5.8 une telle grille.

Paramètre	Valeurs par défaut	Contraintes
$\gamma_s$	0.3	$0 < \gamma_s < 1$
$N_c$	8	$1 \leq N_c \leq 20$
$N_m$	5	$1 \leq N_m \leq 20$
CFL	0.4	$0 < CFL < 0.5$
$\epsilon$	0.01	$1.e^{-10} < \epsilon < 0$
L	4	$1 \leq L \leq 10$

TABLE 5.1 – Paramètres de simulation numérique (valeurs par défaut et contraintes).

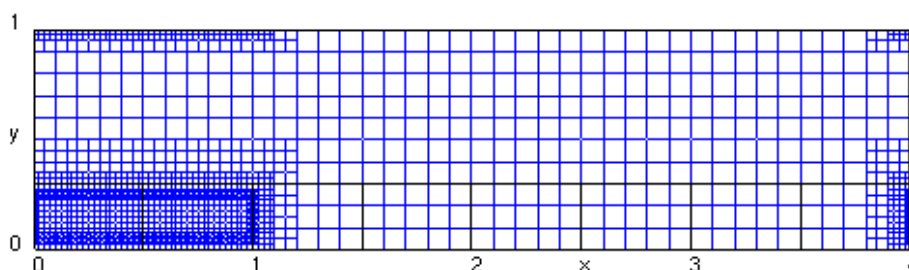


FIGURE 5.8 – Exemple de grille de calcul volumes finis adaptatifs par multirésolution avec  $N_m = 5$  mailles par demi-cycle,  $N_c = 3$  cycles cellulaires, une limite entre zone de prolifération et zone de différenciation positionnée à  $y = \gamma_s = 0.3$ , un nombre de niveaux  $L = 4$  et un seuil  $\epsilon = 0.01$ .



### 5.2 Spécifications biologiques

L'objectif de cette section est de donner des critères de validation d'une simulation basés sur les connaissances biologiques. Les **principaux problèmes** que l'on rencontre sont d'une part la **rareté des données cinétiques et quantitatives dont nous disposons à l'heure actuelle**, d'autre part le fait que **ces données sont disponibles principalement aux niveaux macroscopique (échelle ovarienne) et mésoscopiques (échelle folliculaire), que l'on utilise pour calibrer des paramètres au niveau microscopique**. Ce dernier point pose aussi le problème de l'unicité des valeurs de paramètres conduisant à des sorties macroscopiques identiques.

Nous allons dans un premier temps présenter les différentes mesures dont nous disposons et la méthode expérimentale associée (sans entrer dans les détails). Puis nous définirons les concepts de trajectoire ovulatoire et de trajectoire atrétique.

Nous nous intéressons ici au cas de la brebis, qui est le modèle étudié par les collègues biologistes de l'équipe BINGO (INRA Tours) mais les méthodes seraient transposables à d'autres espèces si l'on pouvait disposer du même ensemble d'observations. L'intérêt du modèle brebis est multiple :

1. C'est une espèce dont la physiologie ovarienne est beaucoup plus proche de l'espèce humaine que les espèces classiques de laboratoire (rat, souris).
2. On dispose dans cette espèce du jeu de données le plus complet, et rassemblant des informations sur la biologie cellulaire, mais aussi sur l'endocrinologie (séries temporelles de niveaux hormonaux).
3. Il existe des mutations naturelles affectant le nombre d'ovulations.

#### 5.2.1 Mesures disponibles

**Diamètre folliculaire.** La technique d'échographie ovarienne (ultrasonographique) permet d'obtenir des images 2D (ou 3D, suivant le type d'équipement), à partir desquelles on peut mesurer le diamètre de l'antrum (cavité à l'intérieur du follicule, la seule structure visible à l'échographie). On en déduit le volume du follicule en supposant que ce dernier est sphérique. Cette technique ne détecte que les follicules au-dessus d'une certaine taille ( $\approx 1\text{mm}$ ) et n'est fiable en terme de suivi individuel qu'au-dessus d'une taille encore plus élevée ( $\approx 2\text{mm}$ ). Dans l'article [66], par exemple, les auteurs présentent des courbes de volume de follicule (diamètre en mm) au cours du temps (en jours) chez la brebis.

**Effectif cellulaire.** Les mesures d'effectif cellulaire ne sont pas directement disponibles en fonction du temps : elles sont données en fonction du diamètre. La mise en correspondance de données de cinétique cellulaire (invasives), telles que : effectifs cellulaires, index mitotique, temps de doublement, et de mesures directes (échographie) ou indirectes (histologie) du diamètre en fonction du temps permet l'élaboration d'une relation entre effectif cellulaire et temps (voir [26]). Nous n'abor-

## Chapitre 5. Calibration

---

derons pas cette question en temps que telle ici, et notre objectif est plutôt de déterminer des valeurs de paramètres compatibles avec les contraintes biologiques.

**Fraction de croissance.** Mathématiquement, la fraction de croissance est une fonction du temps, définie comme le quotient de la masse cellulaire en prolifération (c'est-à-dire des cellules en phases G1 et SM) par la masse cellulaire totale. Cette notion est mésoscopique, au sens où chaque follicule possède sa propre fraction de croissance.

Dans l'article [65], les auteurs présentent des courbes de fraction de croissance de populations de cellules de granulosa pour des follicules ovariens de brebis, de races Romanov ou Ile de France. La fraction de croissance diminue avec le diamètre folliculaire en suivant un profil en cloche, à partir d'une valeur maximale proche de 1mm, jusqu'à devenir proche de zéro aux diamètres ovulatoires (6 ou 7 mm suivant les cas).

**Perte cumulée.** Des observations morphologiques par une approche histologique combinées à une technique de marquage *in situ* permettent de détecter la fragmentation de l'ADN. Cette technique dite TUNEL (voir [42] pour plus de précision) est utilisée dans [49] par exemple sur des cellules de granulosa extraites de follicules de brebis. La présence de taches noires sur les photos au microscope est le témoin de corps pycnotiques, qui sont des condensations du matériel nucléaire (pycnose) que l'on observe suite à l'apoptose. Cependant pour un follicule ovulatoire leur apparition reste minoritaire et on estime empiriquement à 10% le maximum de la perte cellulaire cumulée.

**Non prédestination.** Une notion fondamentale dans le processus de sélection est celle de non prédestination. A priori, parmi les follicules recrutés pour la phase finale de développement, aucun n'est prédestiné, au sens où, en environnement hormonal favorable (que l'on peut mimer avec une valeur de contrôle macroscopique fixée à  $U = U_{\max}$ ), les follicules ont plus de chance d'ovuler et donc à la fois d'atteindre un effectif cellulaire compatible avec l'ovulation (notion d'expression du capital prolifératif) et d'échapper à l'atrésie.

L'exemple typique du non respect de ce principe serait celui d'un follicule dont la maturité asymptotique est dans la zone du domaine où l'apoptose est active pour toute valeur du contrôle. Un tel follicule serait atrétique par nature, donc la sélection serait biaisée. L'atrésie est avant tout un phénomène induit par des conditions d'environnement hormonal défavorables.

Par ailleurs, en conditions d'environnement favorables, la plupart des follicules expriment leur potentiel de prolifération et peuvent atteindre un effectif cellulaire compatible avec l'ovulation.

**Courbes de FSH.** L'évolution du taux plasmatique de FSH au cours du cycle ovarien est un examen assez fréquent, effectué au moyen de prélèvement sanguins

dans le cadre du suivi de cycles ovariens. Sur les mesures expérimentales (voir par exemple dans la thèse [58]) on note que pendant la phase dite folliculaire, on a une chute du taux de FSH d'environ la moitié de la valeur maximale. Cette variation de la quantité de FSH sécrétée par l'hypophyse est régulée par la sécrétion d'hormones (inhibine et œstradiol) ovariennes, en quantité croissante jusqu'à l'ovulation, et qui peut être reliée directement à la maturité ovarienne. Ce rétro-contrôle est modélisé par la relation entre la quantité de FSH et la maturité ovarienne  $U = S(M)$ .

**Durée du processus de sélection** La phase folliculaire se termine par l'ovulation, déclenchée en réponse aux niveaux élevés d'œstradiol (provenant des follicules dominants), par la décharge ovulatoire de la neurohormone GnRH qui induit celle de l'hormone hypophysaire LH. L'ovulation survient environ 24h après le pic de LH, et environ 10 jours à partir de l'instant initial choisi dans le modèle (correspondant au début de la phase terminale du développement folliculaire). Cette durée correspond à un intervalle de 6 à 8 cycles cellulaires parcourus.

### 5.2.2 Trajectoires folliculaires

Comme nous l'avons vu précédemment, les follicules ovariens suivent un développement qui peut aboutir à deux situations : ovulation ou atrésie. Nous allons à présent nous appuyer sur les mesures précédentes pour définir les concepts de trajectoire ovulatoire et de trajectoire atrétique. Cela nous permet de donner des contraintes a priori sur les paramètres de dynamique (comme par exemple le respect de la non prédestination), et de classer les follicules en fin de simulation.

**Trajectoire ovulatoire.** On dira qu'une trajectoire folliculaire est ovulatoire lorsque l'on observe :

- Un accroissement de l'effectif cellulaire compris entre 20 et 30 fois l'effectif initial ;
- Une fraction de croissance qui part de 1 pour chuter près de 0 en environ 10 jours ;
- Une perte cumulée inférieure à 10%.

**Trajectoire atrétique.** On dira qu'une trajectoire folliculaire est atrétique lorsque l'on observe :

- Un décrochage de la courbe de masse cellulaire induit par la chute des niveaux de FSH ;
- Une courbe d'effectif cellulaire décroissante en fin de sélection.

## 5.3 Aides à la calibration

### 5.3.1 Contraintes a priori

A partir des spécifications précédemment établies, on peut imposer des contraintes a priori sur les paramètres. Le but n'est pas d'obtenir directement les paramètres admissibles, car on ne sait pas comment les obtenir tous a priori, mais de réduire

## Chapitre 5. Calibration

---

l'espace des paramètres, par exemple en donnant un bon ordre de grandeur, et en éliminant tous les paramètres conduisant à des situations que l'on peut caractériser comme non souhaitables a priori.

- En particulier les paramètres de la fonction de maturation décrite au paragraphe 5.1.2 doivent être choisis de telle sorte que la racine positive de cette fonction, qui correspond à la maturité cellulaire atteinte en temps long, satisfasse les deux inégalités suivantes :

$$\begin{cases} \gamma_0^{\min} \in [\gamma_s, \gamma_s^+], \\ \gamma_0^{\max} \in [\gamma_s^+, 1], \end{cases}$$

où rappelons-le, les indices min et max correspondent respectivement au contrôle le plus défavorable et le plus favorable. Ces inclusions assurent qu'en contrôle favorable, la maturité asymptotique permet une stabilisation de la masse cellulaire (sortie du cycle cellulaire en temps fini), tandis qu'en contrôle défavorable cette maturité est dans la zone où la perte cellulaire est active, entraînant une chute de masse cellulaire.

- Les données expérimentales sur le taux de FSH permettent d'imposer une relation sur les bornes de la sigmoïde définissant le contrôle global

$$U_{\min} = \frac{1}{2}U_{\max}$$

- L'observation simultanée des taux de FSH plasmatique et des taux d'œstradiol et d'inhibine permet d'introduire des contraintes a priori reliant la dynamique de la FSH au seuil d'ovulation

$$\frac{1}{5}M_oN_f \leq \bar{M} \leq M_oM_f$$

D'autres bornes a priori sont imposées sur les autres paramètres du modèle, en se basant sur les simulations numériques antérieures, et gagneraient à être affinées. L'ensemble des contraintes a priori est résumé dans la Table 5.2.

### 5.3.2 Visualisation à une échelle microscopique

**Contraintes de visualisation.** A l'échelle microscopique, nous pouvons visualiser les densités de cellules de granulosa au cours du temps dans les plans âge-maturité (un plan par follicule). Pour cela, nous pouvons utiliser des codes couleur classiques. Une première idée est d'utiliser un code couleur qui dépend du temps, et est propre à chaque follicule. A chaque temps  $t^n$ , une échelle de couleur est déterminée, variant de 0 jusqu'au maximum de l'amplitude au temps  $t^n$ . Cela permet de bien voir les nuances à chaque pas de temps. Cependant, cela ne met pas en valeurs les différences entre les follicules.

Pour pallier ce problème, on peut définir une échelle de couleur commune à tous les follicules. Mais cela pose un nouveau problème. Les densités étant très variables, l'échelle de couleur est bien adaptée pour le follicule avec la densité la plus élevée, mais il est difficile de distinguer les autres. Typiquement, les follicules ovulatoires sont bien visualisés, mais pas les atrétiques.

Paramètre	Contraintes
$\mu_1$	$0 \leq \mu_1 \leq 10$
$\mu_2$	$0 \leq \mu_2 \leq 10$
$\mu_3$	$0 \leq \mu_3 \leq 10$
$c_1$	$0.01 \leq c_1 \leq 0.1$
$c_2$	$0.1 \leq c_2 \leq 1$
$\bar{u}$	$0.001 \leq \bar{u} \leq 0.01$
$U_{\max}$	$0.1 \leq U_{\max} \leq 0.2$
$U_{\min}$	$U_{\min} = \frac{1}{2}U_{\max}$
$c$	$0 < c < 10$
$\bar{M}$	$\frac{1}{5}M_oN_f \leq \bar{M} \leq M_oN_f$
$M_f$	$1 < M_f < 20$
$M_o$	$M_o > M_f$
$b_1$	$0.01 \leq b_1 \leq 0.1$
$b_2$	$1 \leq b_2 \leq 10$
$b_3$	$1000 \leq b_3 \leq 2000$
$K$	$K > 0$
$\bar{\gamma}$	$0 < \bar{\gamma} < 1$
$\gamma_s^-$	$0 < \gamma_s^- < \gamma_s$
$\gamma_s^+$	$\gamma_s < \gamma_s^+ < 1$

TABLE 5.2 – Résumé des contraintes a priori sur les paramètres.

Nous proposons alors un code couleur adapté à la visualisation des densités dans le cadre de la dynamique de la population de follicules. Cette échelle de couleur est indépendante du follicule considéré, et même indépendante du temps.

**Code couleur adapté à la dynamique de population.** Nous utilisons pour cela le cercle chromatique (voir Figure 5.9). Il contient toutes les couleurs que l'on peut obtenir à partir de couleurs de base en visualisation informatique : les couleurs du système RGB (pour Red, Green, Blue). Nous restons dans l'esprit d'une échelle de température. Nous utilisons ainsi, dans l'ordre : blanc, bleu, cyan, vert, jaune, rouge, magenta, pour obtenir l'échelle de couleur représentée sur le panel droite de la Figure 5.9.

L'idée est d'utiliser une relation non linéaire pour passer d'une couleur à l'autre, contrairement à la plupart des codes couleur classiques, qui utilisent une relation linéaire. Dans notre problème, la densité est amenée à traverser différentes interfaces. Les variations de vitesse, ainsi que les accroissements de masse aux interfaces de type mitose, modifient l'amplitude de la densité. On propose d'introduire une saturation non-linéaire des couleurs basée sur la variation par paliers de l'amplitude de la solution, due aux mitoses successives.

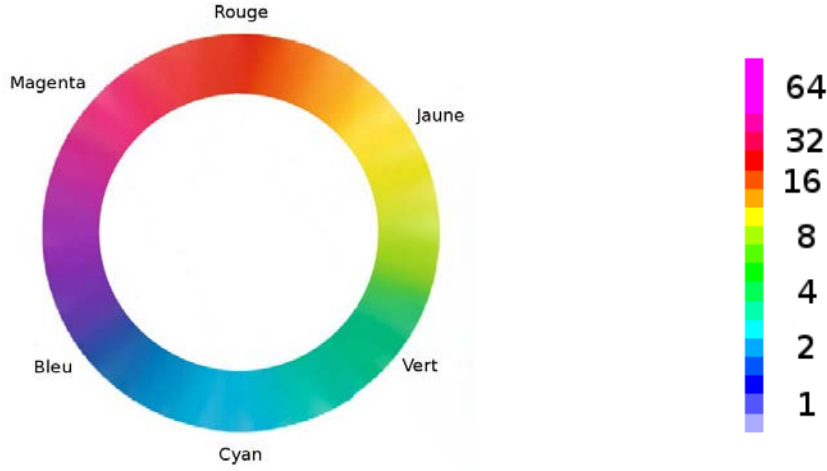


FIGURE 5.9 – Développement d'un code couleur adapté à la dynamique de la population de follicules. A gauche, le cercle chromatique. A droite, utilisation du cercle chromatique pour définir un code couleur, basé sur le maximum de la densité de population au temps initial (dans cet exemple le maximum vaut 1), qui double périodiquement.

En pratique, on procède ainsi. Nous commençons par identifier

$$z_{\text{ref}} = \sup_{a,\gamma,f} |\phi_0^f(a, \gamma)|,$$

le sup des densités des follicules au temps initial et on découpe en intervalles d'amplitude croissante en multiples entiers de la plage de variation globale de la solution pendant la simulation. On associe à chaque intervalle une portion du cercle chromatique. Pour déterminer la couleur à utiliser dans une cellule du maillage Volumes finis, on détermine dans quel intervalle se trouve la solution, dont la taille varie avec le nombre de doublements de population. Cela conduit à la relation non linéaire :

- $0 \leq z \leq z_{\text{ref}}$  : de blanc à bleu
- $z_{\text{ref}} \leq z \leq 2z_{\text{ref}}$  : de bleu à cyan
- $2z_{\text{ref}} \leq z \leq 4z_{\text{ref}}$  : de cyan à vert
- $4z_{\text{ref}} \leq z \leq 8z_{\text{ref}}$  : de vert à jaune
- $8z_{\text{ref}} \leq z \leq 16z_{\text{ref}}$  : de jaune à rouge
- $16z_{\text{ref}} \leq z \leq \sup_t \sup_f \sup_{a,\gamma} |\phi^f(a, \gamma, t)|$  : de rouge à magenta

Après la sortie du cycle (dans la partie haute du domaine), la densité se concentre en temps long, et cela a pour effet de colorer la densité en magenta. Cette échelle de couleur permet de bien capturer les différences entre les follicules, de façon indépendante du temps et sans privilégier les valeurs de la solution proche du maximum. On donne un exemple d'une telle visualisation sur la Figure 5.10.

**Code couleur par analogie avec la visualisation en Biologie.** Une autre possibilité est de visualiser les densités de cellules de granulosa en s'inspirant de

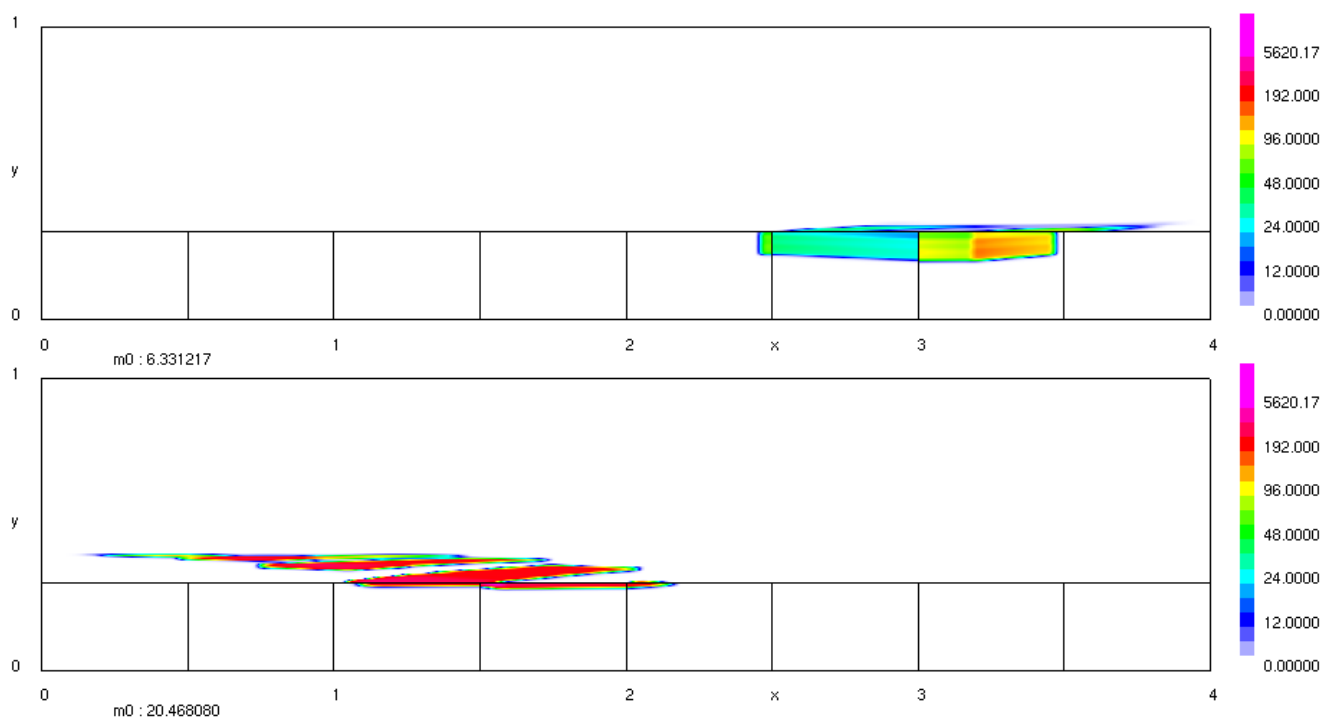


FIGURE 5.10 – Visualisation de deux follicules, l'un ovulatoire (en bas, avec un accroissement cellulaire d'environ 20) et l'autre atrétique (en haut), avec un code couleur non linéaire adapté à la dynamique de la population de follicules. Dans ce cas  $N_c = 4$ , et la densité du follicule ovulatoire (en bas) a déjà parcouru entièrement une fois le domaine (rappelons que l'on a des conditions aux limites périodiques).

## Chapitre 5. Calibration

l'article [69], dans lequel les auteurs proposent une méthode qui permet de différencier les phases G1 et SM du cycle par imagerie par fluorescence (après transfection cellulaire par des marqueurs adaptés). Pour cela, on définit deux échelles de couleurs linéaires, de blanc à rouge, et de blanc à vert. Les parties de densité en zone G1 ou D sont affichées avec l'échelle de rouge, et les parties en zone SM en échelle de vert. Nous affichons également en jaune les cellules dans la zone de transition entre la zone SM et la zone G1, qui sera utilisée au paragraphe 5.3.3 pour définir l'index mitotique. Sur la Figure 5.11, on donne un exemple d'une telle visualisation.

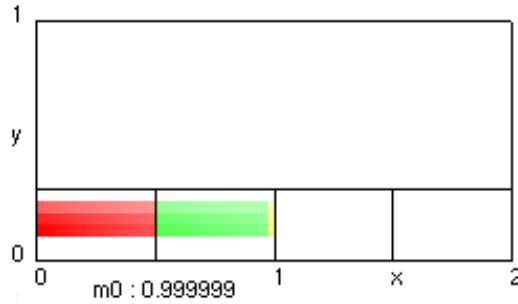


FIGURE 5.11 – *Visualisation d'un follicule avec un code couleur spécifique des phases cellulaires. Les densités non nulles sont affichées en rouge dans les sous-domaines correspondant aux phases G1 et D, en vert pour les phases SM. Les cellules en cours de mitose (notamment utilisées pour calculer l'index mitotique numérique) sont affichées en jaune.*

### 5.3.3 Diagnostic a posteriori

En fin de simulation, on peut se baser sur plusieurs critères pour effectuer l'interprétation biologique des résultats :

- Non prédestination : faire la simulation à contrôle global constant ( $U(t) = U_{\max}$ ).  
Observe-t-on un changement dans le nombre de follicules ovulatoires ?
- Vérification des trajectoires ovulatoires.

Pour aider au diagnostic, on utilise différentes sorties du modèle au niveau mésoscopique et macroscopique que nous présentons dans la suite de ce paragraphe. Elles seront dans la suite systématiquement représentées sur des figures à neuf panels semblables à la Figure 5.12.

**Masses et maturités folliculaires.** Les courbes de masse et de maturité folliculaires sont respectivement les moments d'ordre zéro

$$m_0^f(t) = \iint_{[0, N_c] \times [0, \gamma_s]} \phi_f(a, \gamma, t) da d\gamma,$$

et d'ordre un en maturité

$$m_1^f(t) = \iint_{[0, N_c] \times [0, \gamma_s]} \gamma \phi_f(a, \gamma, t) da d\gamma,$$



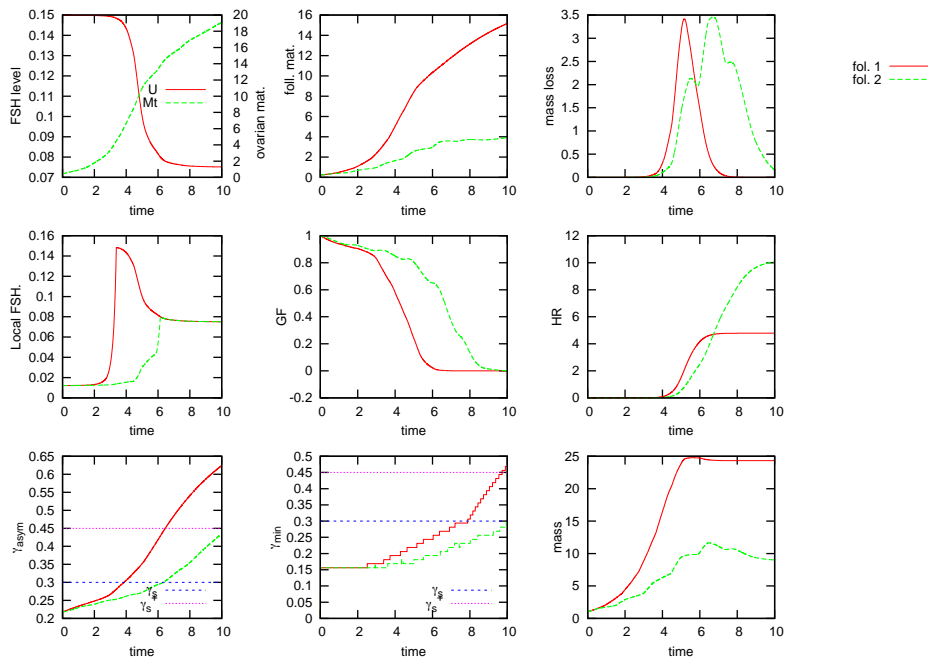


FIGURE 5.12 – Exemple de panel de diagnostic de simulation numérique. De gauche à droite, de haut en bas : Contrôle macroscopique, maturités mésoscopiques, pertes de masses mésoscopiques instantanées, contrôles mésoscopiques, fractions de croisances mésoscopiques, perte mésoscopiques cumulées de masse,  $\zeta^f$  mésoscopiques,  $\gamma_{min}^f$  mésoscopiques, masses mésoscopiques. Dans cet exemple la cohorte comprend deux follicules, qui diffèrent uniquement par leurs paramètres de vitesse de vieillissement.

## Chapitre 5. Calibration

---

des densités folliculaires. Rappelons ici que pour un follicule ovulatoire, l'accroissement de masse doit être compris entre 20 et 30 fois l'effectif initial, et qu'un follicule atrétique est caractérisé par un décrochement de la courbe de masse.

**Maturité ovarienne et contrôle macroscopique.** En haut à gauche de la Figure 5.12, on représente les courbes de maturité ovarienne et de valeur du contrôle macroscopique. Il faut que la courbe de niveau plasmatique de FSH au cours du temps  $U(t)$  chute à partir de sa valeur asymptotique  $U_{\max}$  pour qu'il y ait sélection parmi les follicules.

**Contrôle mésoscopique.** Au centre à gauche de la Figure 5.12, on trace les courbes de disponibilité locale en ressource FSH (une courbe par follicule). Typiquement, une courbe de ce style croit et atteint  $U(t)$  (qui se trouve être la plupart du temps  $U_{\min}$ ), dans le cas des follicules ovulatoires, et décroît pour les follicules atrétiques.

**Fraction de croissance.** Rappelons que l'on définit la fraction de croissance d'un follicule par

$$GF_f(t) = \frac{m_f(t)}{m_0^f(t)},$$

où  $m_f(t) = \iint_{[0, N_c] \times [0, \gamma_s]} \phi_f(a, \gamma, t) da d\gamma$  est l'effectif cellulaire dans la partie du domaine correspondant au cycle cellulaire.

**Courbes  $\zeta^f$ .** En bas à gauche de la Figure 5.12, on représente les courbes de maturité cellulaire moyenne définies par

$$\zeta^f(t) = \frac{m_1^f(t)}{m_0^f(t)}.$$

Cette grandeur indique ainsi la maturité asymptotique vers laquelle tend la densité en temps long (voir [60])

**Courbe  $\gamma_{\min}^f$ .** En bas au centre, on trace la courbe de la valeur minimale de la maturité, que l'on note  $\gamma_{\min}$  :

$$\gamma_{\min}^f = \min\{\gamma \in [0, 1] \mid \exists a \in [0, N_c], \phi_f(a, \gamma) > 0\}.$$

On l'évalue numériquement en comparant la densité avec une petite valeur de seuil, fixée arbitrairement à  $10^{-2}$ . Autrement dit, si la densité est une partie compacte du plan, cette quantité représente l'ordonnée minimale parmi tous les points composants cette densité.

## Chapitre 5. Calibration

---

**Courbe d'index mitotique.** L'index mitotique est une grandeur cinétique définie au temps  $t$  comme le rapport entre le nombre de cellules en train de se diviser et l'effectif total

$$I_m^f(t) = \frac{\text{mitotic cells}}{\text{total number of cells}}$$

La littérature fait état de mesures expérimentales d'index mitotique (voir par exemple [75] et [76]). Cette grandeur reste constante dans le cas de populations entièrement proliférantes et désynchronisées, alors qu'elle présente un profil complexe dans le cas de populations subissant d'autres processus cellulaires comme la différenciation et/ou l'apoptose (ce qui est le cas des cellules de granulosa). La durée  $T_m$  du phénomène de mitose peut être considérée comme relativement stable et d'environ 30 minutes, à rapporter à une durée de cycle de prolifération qui peut varier entre 12 et 18 heures. Dans notre système de variable normalisée, où la durée du cycle est de 1, nous aurons donc

$$\frac{1}{36} \leq T_m \leq \frac{1}{24}.$$

Pour calculer une estimation de l'index mitotique dans le cadre de notre modèle, nous devons prendre en compte la durée de la mitose qui débute en fin de phase SM et finit par la division cellulaire sur l'interface entre la zone SM du cycle en cours et la zone G1 du cycle suivant. Pour cela nous introduisons en plus du domaine de calcul

$$\Omega = \{(a, \gamma), 0 \leq a \leq N_c, 0 \leq \gamma \leq 1\},$$

où  $N_c$  est le nombre de cycles cellulaires, la zone de mitose

$$\Omega_{Mit} = \{(a, \gamma), p - T_m \leq a < p, \text{ for } p = 1, \dots, N_c, 0 \leq \gamma < \gamma_s\}$$

Ceci nous permet de définir précisément l'index mitotique numérique

$$I_m^f(t^n) = \frac{\sum_{\Omega_{i,j}^\ell \cap \Omega_{Mit} \neq \emptyset} |\Omega_{i,j}^\ell \cap \Omega_{Mit}| \phi_{i,j}^{n,\ell}}{\sum_{\Omega_{i,j}^\ell \subset \Omega} |\Omega_{i,j}^\ell| \phi_{i,j}^{n,\ell}} = \frac{\sum_{\Omega_{i,j}^\ell \cap \Omega_{Mit} \neq \emptyset} |\Omega_{i,j}^\ell \cap \Omega_{Mit}| \phi_{i,j}^{n,\ell}}{m_0^f(t)}$$

Il est à remarquer que la limite à gauche de la zone de mitose peut tomber à l'intérieur d'une maille de la grille volume finis (adaptative qui plus est). C'est pourquoi nous prenons soin de pondérer la valeur de la densité sur une maille par la surface contenue effectivement dans la zone de mitose.

Sur la Figure 5.13, on trace une courbe d'index mitotique dans le cas d'un transport simple en zone de prolifération G1-SM, avec vitesse verticale nulle, vitesse horizontale constante, prise égale à 1, pour deux types de condition initiale : constante par morceaux, et à décroissance exponentielle en âge, constante par morceaux en maturité. On note l'impact du choix de la condition initiale sur l'allure de la courbe. Dans le cas d'une population initiale distribuée exponentiellement en âge de manière à satisfaire la condition de doublement de la population au bout de la durée d'un cycle cellulaire (voir paragraphe 5.1.1) l'index mitotique reste bien constant, aux erreurs numérique près. Dans le cas d'une condition initiale uniforme, la discontinuité sur l'effectif introduite par la mitose se transmet à l'index mitotique, qui

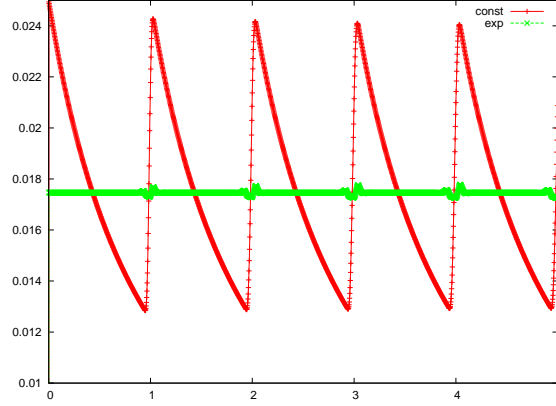


FIGURE 5.13 – Courbes d’index mitotique correspondantes aux deux conditions initiales envisagées dans le modèle (voir paragraphe 5.1.1), dans le cas entièrement proliférant ( $h_f = 0$ ) et sans discontinuité de vitesse entre la zone G1 et la zone SM ( $g_1 = 0, g_2 = 1$ ).

conserve néanmoins une valeur moyenne stable. Un des développements en cours est d’exploiter cette sortie sur des simulations du modèle complet et d’étudier l’impact de la différenciation et des discontinuités de vitesse en âge entre les zones G1 et SM.

**Perte instantanée et perte cumulée.** On définit la perte de masse instantanée  $r^f$  pour chaque follicule  $f = 1, \dots, N_f$  :

$$r^f(t) = \iint_{[0, N_c] \times [0, 1]} \Lambda(a, \gamma, U) \phi_f(a, \gamma, t) da d\gamma.$$

On évalue  $r^f(t)$  par quadrature numérique

$$r^f(t^n) \approx \sum_{i=1}^{N_x} \sum_{j=1}^{N_y} \Lambda(a_i, \gamma_j, U^n) \phi_f(a_i, \gamma_j, t^n) \Delta a \Delta \gamma.$$

On définit la perte cumulée au cours du temps pour chaque follicule

$$R^f(t) = \int_0^t r^f(s) ds,$$

qu’on évalue numériquement par

$$R^f(t) \approx \sum_{n=1}^N r^f(t^n) \Delta t^n.$$

On représente ces deux indicateurs sur la Figure 5.14. On constate que le taux de perte instantanée (sur le panel de gauche) est nul avant l’arrivée dans la zone où le terme source est actif, puis non nul pendant l’intervalle de temps où la densité traverse la zone d’apoptose ( $\gamma_s^- \leq \gamma \leq \gamma_s^+$ ), et redevient nulle quand la densité

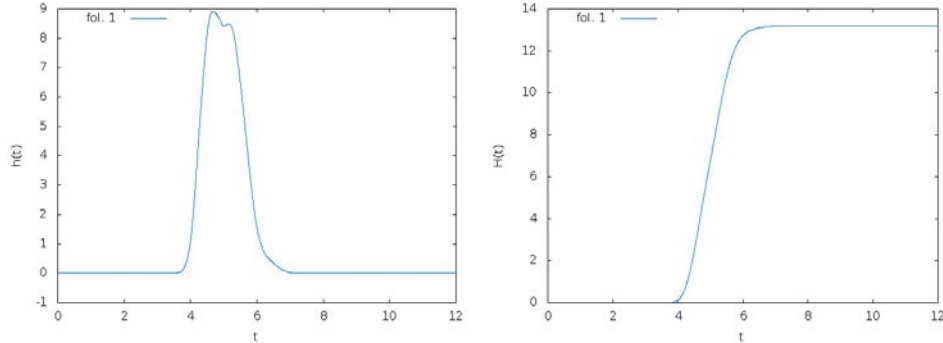


FIGURE 5.14 – Perte instantanée et perte cumulée. Résultat d’une simulation numérique réalisée avec les paramètres par défaut (voir Tableau 5.3) avec un contrôle en boucle fermée. Panel de gauche : Exemple de fonction de perte instantanée. En abscisse, le temps, en ordonnée, la perte cellulaire instantanée. Cette fonction est toujours positive ou nulle. Panel de droite : Exemple de fonction de perte cumulée, associée à la courbe présentée dans le panel de gauche.

est entièrement dans la zone  $\gamma > \gamma_s^+$ . La perte cumulée, représentée sur le panel de droite, est la primitive de la perte instantanée, le niveau atteint à la fin de la simulation peut être comparé au seuil approximatif de 10% que l’on s’est fixé comme limite maximale pour un follicule ovulatoire.

## 5.4 Méthode de calibration pour un follicule

Dans cette section, nous présentons une méthode de calibration pour un follicule en deux étapes :

1. Détermination de la maturité asymptotique, à contrôle maximal constant ;
2. Détermination des paramètres affectant la dynamique, à contrôle maximal constant ;

A l’issue de ces étapes, on obtient une situation de référence, que l’on peut ensuite perturber (aléatoirement ou de façon déterministe) pour en obtenir d’autres.

### 5.4.1 Détermination de la maturité asymptotique

On a détaillé dans le paragraphe 5.1.2 le rôle des zéros de la vitesse de maturation dans le comportement asymptotique en temps long de la densité. Il est important de noter que le paramètre  $\tau_f$  n’a pas d’influence sur les valeurs des zéros de  $h_f$ . On peut ainsi faire varier l’amplitude de  $h_f$  tout en gardant la même valeur asymptotique pour la solution, à contrôle constant. Le paragraphe suivant montre comment choisir les autres paramètres de la vitesse de maturation, qui eux, influent sur la position des zéros.

## Chapitre 5. Calibration

**Cubes de visualisation.** On cherche à déterminer les valeurs des paramètres  $c_1$ ,  $c_2$  et  $\bar{u}$  tels que la maturité asymptotique en temps long soit dans le domaine différencié, hors de la zone d'apoptose pour une valeur maximale du contrôle en fin de simulation (donc quand  $u_f = U_{\min}$ ), et dans la zone d'apoptose quand  $u_f = \bar{u}_{\min}$  (voir Figure 5.15). Ces contraintes nous permettent de respecter la notion de non-prédestination en terme d'apoptose, et des conditions pour obtenir un phénomène de sélection.

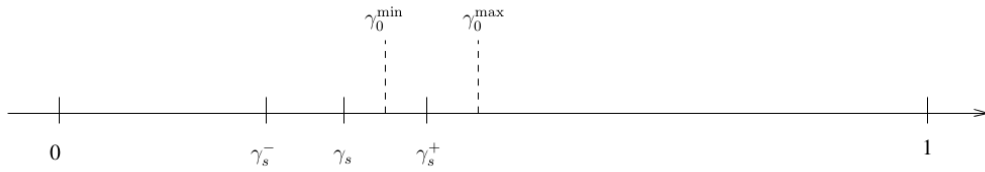


FIGURE 5.15 – Schéma illustrant le positionnement de la racine positive de  $h_f$ . On représente sur l'axe  $\gamma$  la position de l'interface  $\gamma_s$ , et de part et d'autre, les limites de la zone d'apoptose,  $\gamma_s^-$  et  $\gamma_s^+$ . Au dessus de l'axe, on note la position de la racine positive de  $h_f$ ,  $\gamma_0^{\min}$  et  $\gamma_0^{\max}$ , pour un contrôle local défavorable ( $\bar{u}_{\min}$  en fin de simulation) et pour un contrôle favorable ( $U_{\min}$ ), respectivement. La racine supérieure de la fonction  $h$  pour un contrôle favorable,  $\gamma_0^{\max}$ , est entre  $\gamma_s^+$  et 1, et sa racine pour un contrôle défavorable,  $\gamma_0^{\min}$ , est entre  $\gamma_s$  et  $\gamma_s^+$ .

On se place dans un repère 3D avec pour axes :  $c_1$  en abscisse,  $c_2$  en ordonnée et  $\bar{u}$  en altitude. On calcule sur une grille discrète les valeurs de la racine positive  $\gamma_0(c_1, c_2, \bar{u})$  de la fonction  $h_f$ , pour les deux valeurs limite du contrôle  $u_f = \bar{u}_{\min}$  et  $u_f = U_{\min}$ . La Figure 5.16 montre la visualisation dans le repère 3D  $(c_1, c_2, \bar{u})$  des valeurs de  $\gamma_0$  par un code couleur.

L'idée, comme nous l'avons vu précédemment, est de trouver une racine supérieure en zone différenciée hors apoptose (afin de stabiliser la masse cellulaire), et une racine inférieure en zone différenciée, dans le domaine d'apoptose (afin d'observer un décrochement de la masse cellulaire).

On trace deux cubes de visualisation (voir Figure 5.17). Un premier cube pour une valeur de variable de contrôle défavorable, qui définit un ensemble

$$E_1 = \{(c_1, c_2, \bar{u}) | \gamma_0^{\min} \in [\gamma_s, \gamma_s^+]\},$$

puis un second pour une valeur de variable de contrôle favorable, qui définit un autre ensemble

$$E_2 = \{(c_1, c_2, \bar{u}) | \gamma_0^{\max} \in [\gamma_s^+, 1]\}.$$

L'intersection  $E = E_1 \cap E_2$  définit une plage de paramètres admissibles pour les paramètres  $c_1, c_2, \bar{u}$ . On peut alors raffiner la représentation de ces deux cubes, en les représentant tous deux sur l'ensemble  $E$  seulement. Enfin, on peut, en raffinant l'ensemble  $E$ , préciser le choix de valeurs des racines  $\gamma_0^{\min}$  et  $\gamma_0^{\max}$  (on peut par

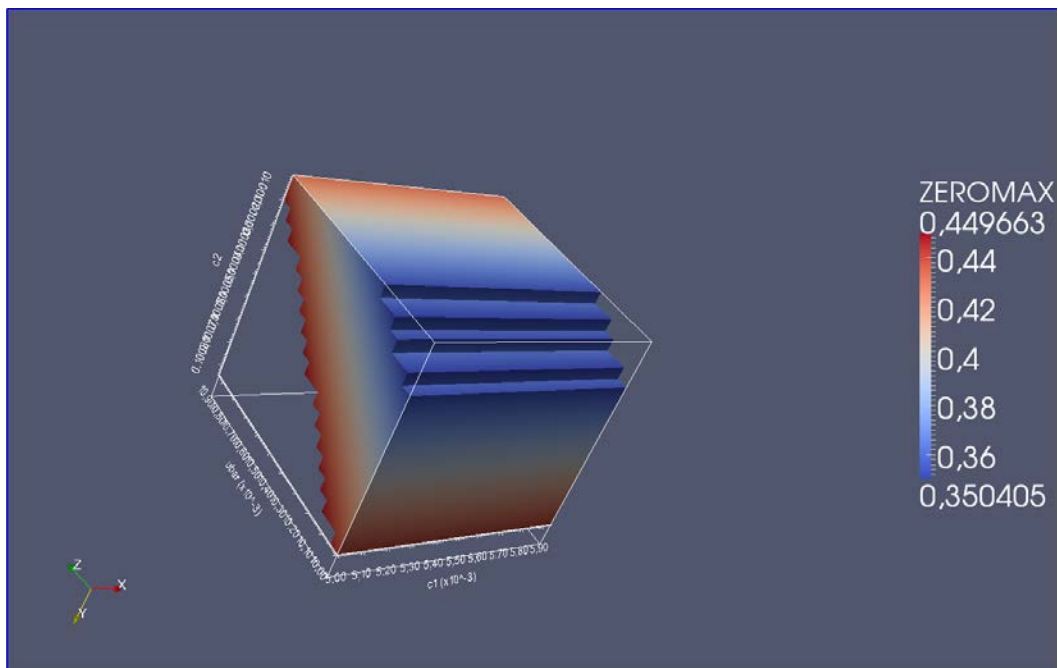


FIGURE 5.16 – Cube de visualisation des racines de la fonction  $h_f$ . En abscisse  $c_1$ , en ordonnée  $c_2$ , en altitude  $\bar{u}$ . Tout triplet  $(c_1, c_2, \bar{u})$  définit une racine positive de  $h_f$ , notée  $\gamma_0$ . La surface de réponse représentée est une portion de cube, donnant la valeur des zéros de  $h$  dans un intervalle  $[\gamma_1, \gamma_2]$  prescrit (dans cet exemple,  $\gamma_1 = 0.35$  et  $\gamma_2 = 0.45$ ). Ainsi, tout triplet sélectionné dans cette hypersurface vérifie la contrainte  $\gamma_0 \in [\gamma_1, \gamma_2]$ .

## Chapitre 5. Calibration

exemple régler l'écart entre ces deux valeurs), et au final sélectionner un triplet de valeurs admissibles.

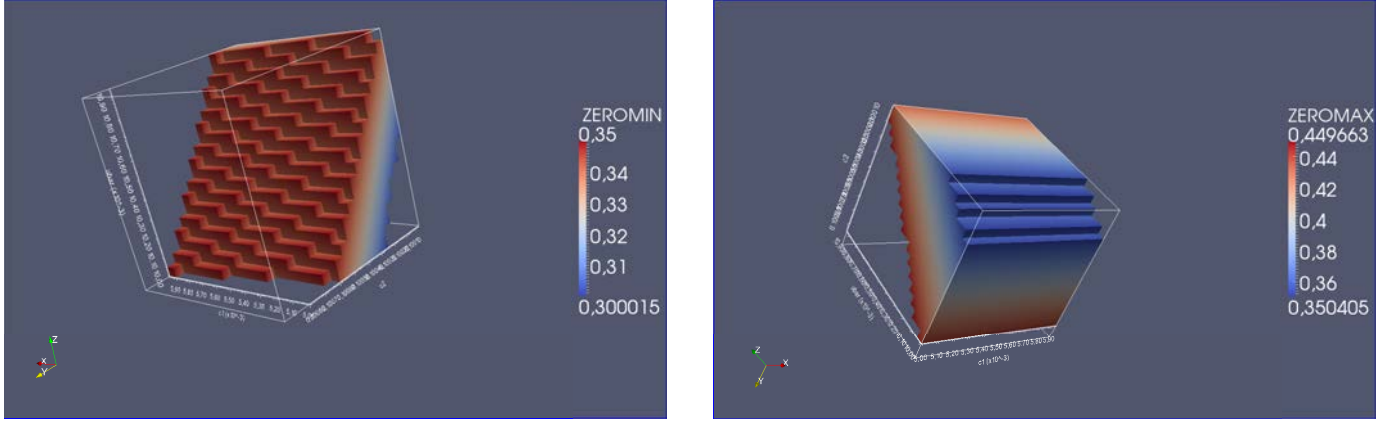


FIGURE 5.17 – Cubes de visualisation des racines de la fonction  $h_f$ . Pour déterminer un triplet  $(c_1, c_2, \bar{u})$  admissible, on trace sur le cube de gauche une carte de couleur avec les zéros de  $h_f$  à contrôle minimal (typiquement, pour un follicule atréitique  $\gamma_0 \in [\gamma_s, \gamma_s^+]$ , pour  $u_f = \bar{u}_{\min}$ ), à droite une carte de couleur avec les zéros de  $h$  pour un contrôle maximal (typiquement, un follicule ovulatoire  $\gamma_0 > \gamma_s^+$ , pour  $u_f = U_{\min}$ ). On peut alors choisir un triplet à l'intersection des deux sous ensembles. Dans cet exemple,  $\gamma_s = 0.3$  et  $\gamma_s^+ = 0.35$ .

### 5.4.2 Détermination des paramètres affectant la dynamique

Dans le paragraphe précédent, on s'est attaché à définir une maturité asymptotique. Une fois celle-ci fixée, il reste à définir la dynamique pour l'atteindre. Dans un premier temps, nous cherchons à contrôler les sorties, et plus précisément la masse cellulaire au temps final, à contrôle macroscopique constant (i.e. en boucle ouverte). Cette étape est une étape préliminaire à l'étude du système en boucle fermée. Pour cela, nous définissons un sous problème.

**Sous problème.** Supposons que  $U(t) = U_{\max}$ . Comment prescrire la masse finale d'un follicule? Supposons que l'on fixe une masse en sortie de cycle cellulaire. Si l'on fait varier une composante de vitesse, comment faire varier l'autre composante pour arriver à la masse finale prescrite? Il s'agit de trouver un compromis entre vitesse de vieillissement  $g_f$  et vitesse de maturation  $h_f$ . Empiriquement, on trouve une relation linéaire entre les fonctions

$$g_f(a, \gamma, u_f) = \begin{cases} g_1 u_f + g_2 & \text{en } G1, \\ 1 & \text{en } SM \cup D \end{cases}$$

et

$$h_f(a, \gamma, u_f) = \tau_f \begin{cases} -\gamma^2 + (c_1 \gamma + c_2)(1 - e^{-\frac{u_f}{a}}) & \text{en } G1 \cup D, \\ 0 & \text{en } SM \end{cases}$$



## Chapitre 5. Calibration

---

dans le cas simplifié où les paramètres  $c_1, c_2, \bar{u}$  restent inchangés. Ainsi, pour obtenir la même masse finale, lorsque l'on divise les paramètres  $g_1$  et  $g_2$  par une même quantité  $p$ , il suffit de diviser  $\tau_f$  par cette même quantité.

**Reparamétrisation.** On a vu que les paramètres du sous problème étaient liés de façon linéaire à contrôle constant. On peut alors ré-exprimer les paramètres des fonctions  $g_f$  et  $h_f$ . Nous posons

$$\begin{cases} g_1 = p, \\ g_2 = p\alpha, \\ \tau_f = C_p \times p, \\ \alpha = \frac{g_2}{g_1}. \end{cases}$$

avec  $p, C_p$  et  $\alpha$  trois constantes réelles, que nous désignerons par le terme de "paramètres de travail". Nous allons à présent voir l'utilité de cette reparamétrisation.

**Paramètre de capital prolifératif à contrôle constant  $C_p$ .** Sur la Figure 5.18, on trace les sorties mésoscopiques et macroscopiques pour différentes valeurs du paramètre de travail  $C_p$  dans la plage de paramètres : 0.56, 0.57, 0.58, 0.59, 0.6, 0.61, 0.62, 0.63, les autres paramètres étant les paramètres par défaut, pour une cohorte de 8 follicules. Le paramètre  $C_p$  semble contrôler la masse finale de la solution à contrôle constant. La plage de valeurs dans laquelle on fait varier le paramètre de travail  $C_p$  a été choisie de manière à obtenir un accroissement de masse compris entre 20 et 30.

**Paramètre de dynamique à contrôle constant  $p$ .** Sur la Figure 5.19, on trace les solutions pour différentes valeurs du paramètre de travail  $p$  0.75, 1, 1.25, 1.5, 1.75, 2, 2.25, 2.5, 2.75, 3, à paramètre de travail  $C_p = 0.6$  fixé. On remarque que les courbes de masse et de maturité n'ont pas suivi la même trajectoire au cours du temps mais aboutissent à la même valeur finale. Le paramètre de travail  $p$  permet ainsi de régler la cinétique d'accroissement cellulaire.

### 5.4.3 Exemple de situation de référence à 1 follicule

Nous concluons ce paragraphe en combinant les deux méthodes précédentes (détermination de la maturité asymptotique, détermination des paramètres affectant la dynamique) pour définir un exemple de situation de référence. La Figure 5.20 présente le panel de diagnostic associé à cette situation. On note que le follicule satisfait tous les critères d'une trajectoire ovulatoire.

- La valeur du contrôle global chute de moitié entre le début et la fin de la simulation.
- La fraction de croissance chute de 1 à près de 0.
- Le taux de mort cumulée est inférieur à 10%.
- L'accroissement d'effectif cellulaire est compris entre 20 et 30 fois l'effectif initial.

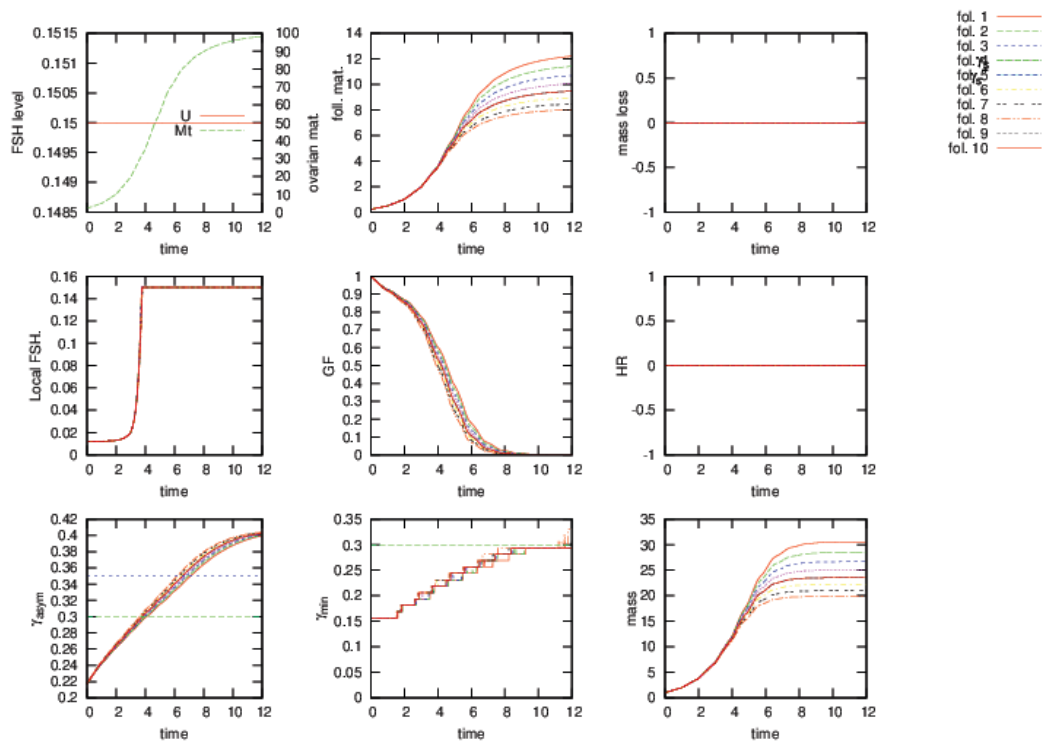


FIGURE 5.18 – *Diagnostic de simulation numérique à contrôle macroscopique à constant  $U = U_{\max}$ , pour une plage de valeurs du paramètre de travail  $C_p$  (les autres valeurs étant prises par défaut). Ce paramètre permet de piloter la masse finale à contrôle constant. Ici, on a représenté des courbes de masse atteignant des valeurs finales comprises entre 20 et 30, de façon à respecter les spécifications d’une trajectoire ovulatoire.*

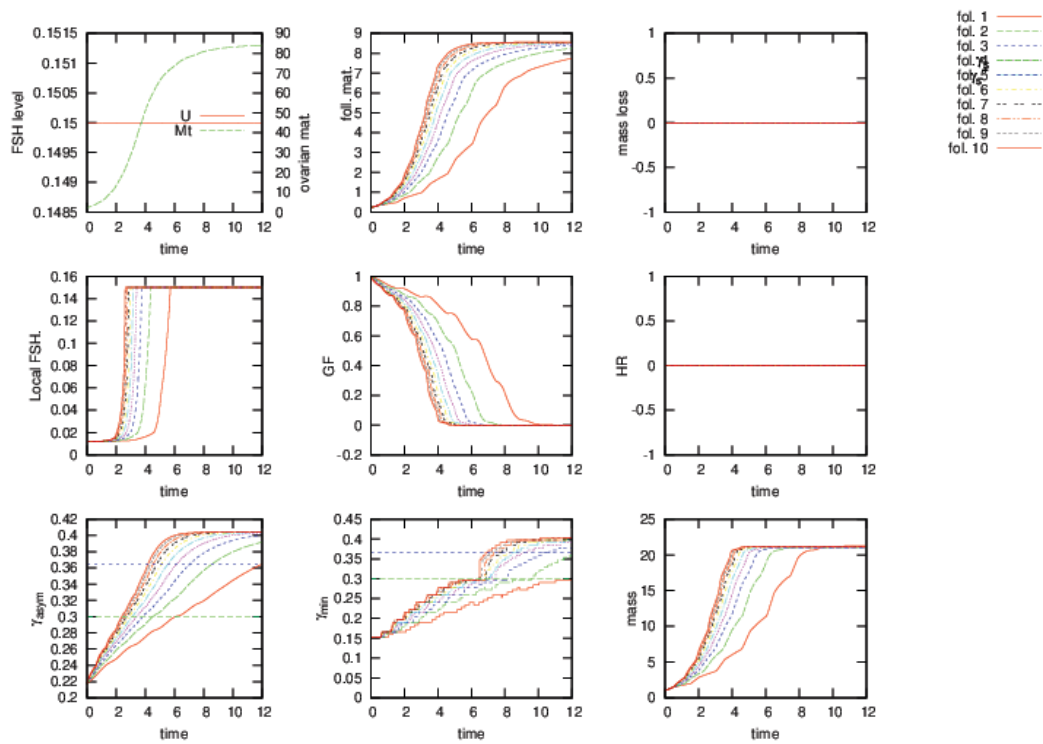


FIGURE 5.19 – Diagnostic de simulation numérique à contrôle constant, pour une plage de valeurs du paramètre de travail  $p$ . Ce paramètre permet de piloter la cinétique d'accroissement cellulaire, tout en gardant les mêmes masses et maturités finales.

## Chapitre 5. Calibration

- La maturité asymptotique atteinte en fin de simulation est en dehors de la zone d’apoptose.
- L’ovulation survient autour de 8 cycles cellulaires après le début du processus de sélection (ce qui correspond à environ 10 jours).

Le tableau 5.3 présente les valeurs des paramètres pour ce cas. Par la suite, nous utiliserons ces valeurs par défaut.

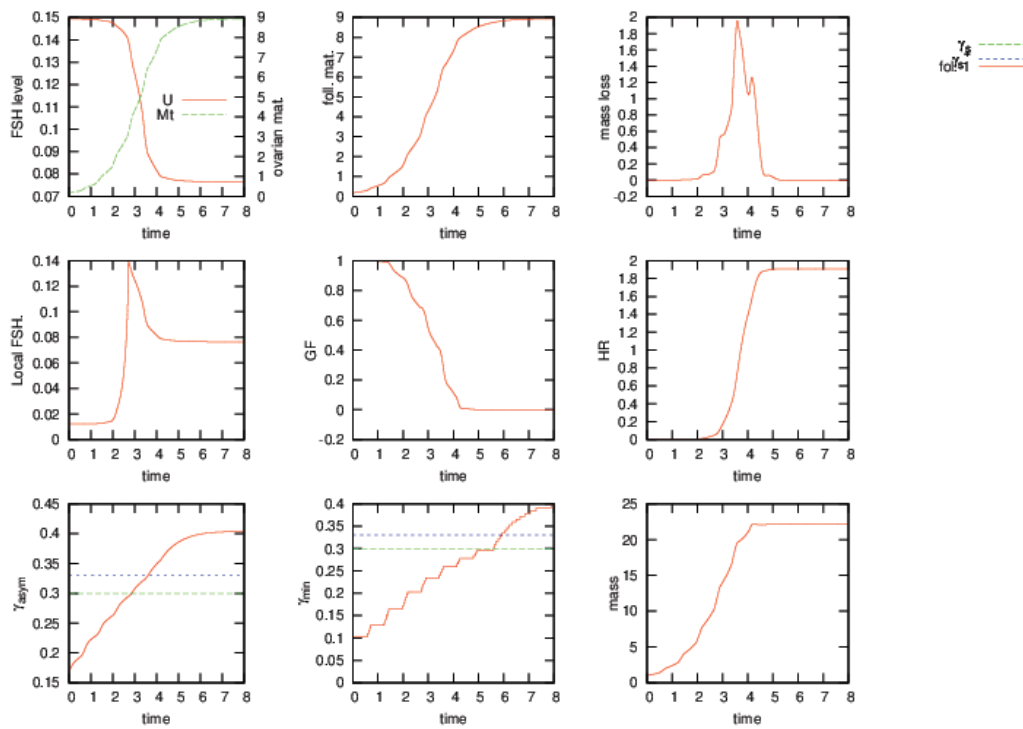


FIGURE 5.20 – *Panel de diagnostic de simulation numérique associé à la situation de référence à un follicule. Ce follicule satisfait tous les critères d’un follicule ovulatoire.*

### 5.5 Méthode de calibration pour une cohorte de follicules

Dans cette section, nous allons voir comment passer du cas à un follicule au cas d’une cohorte de follicules. Deux notions importantes doivent être prises en compte :

- la pression sélective ;
- l’hétérogénéité.

Paramètre	Valeur par défaut
$\mu_1$	8
$\mu_2$	7
$\mu_3$	5
$g_1$	1
$g_2$	1
$c_1$	0.032
$c_2$	0.148
$\bar{u}$	0.005
$p$	1
$C_p$	0.6
$\alpha$	1
$U_{\max}$	0.15
$U_{\min}$	0.075
$c$	2
$\bar{M}$	4.5
$M_f$	9
$M_o$	15
$b_1$	0.08
$b_2$	2.2482
$b_3$	1450
$K$	0.6
$\bar{\gamma}$	0.1
$\gamma_s^-$	0.25
$\gamma_s^+$	0.35

TABLE 5.3 – Paramètres par défaut (utilisés pour les simulations représentées dans la figures 20, et les suivantes).

### 5.5.1 Réglage de la pression sélective

Régler la pression sélective consiste à régler les paramètres de la fonction déterminant le contrôle global. Pour cela, on fait varier les paramètres  $c$  et  $\bar{M}$ . Tout d'abord, le paramètre  $\bar{M}$  doit être réglé de façon à ce que la chute de FSH ait bien lieu lorsque la maturité globale est suffisante, ni trop tôt, ni trop tard. Ensuite, le paramètre  $c$  doit être choisi de façon à ce que la pente du profil de chute de FSH ne soit pas trop abrupte en temps. Enfin, on peut régler le paramètre  $K$  déterminant l'intensité de l'apoptose, et ainsi respecter la perte maximale de 10% pour les follicules ovulatoires. Sur la Figure 5.21, on reprend les paramètres utilisés pour la Figure 5.19, mais cette fois-ci on introduit le contrôle en boucle fermée. On règle empiriquement les valeurs de  $c$ ,  $\bar{M}$  et  $K$ , dans cet ordre. La chute de FSH a pour effet de créer un phénomène de sélection. Sans sélection (en boucle ouverte), tous les follicules atteignaient la même masse finale (voir Figure 5.19). A présent, les masses finales s'étalent, et on

voit même apparaître un follicule atrétique. Il est important de noter que ce même follicule est ovulatoire à contrôle maximal (il n'est pas prédestiné).

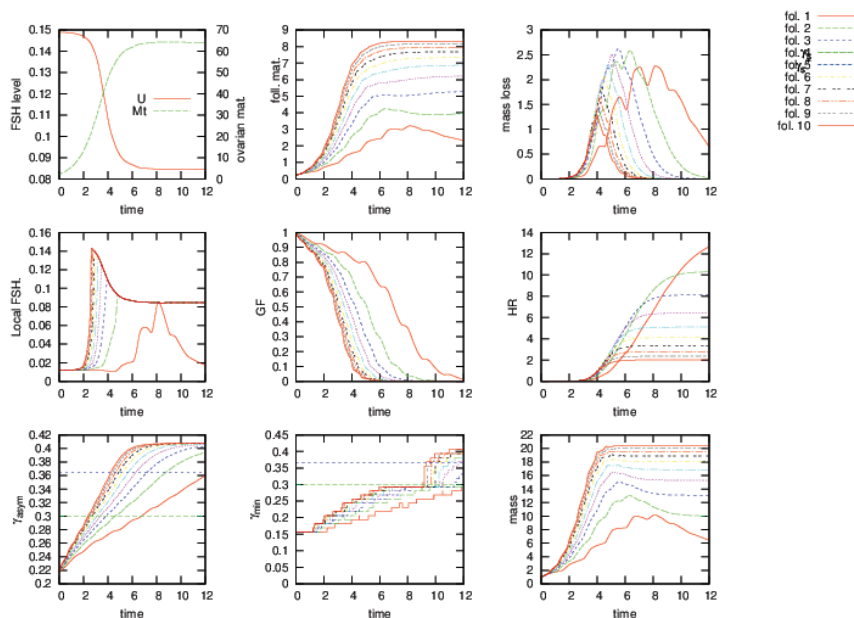


FIGURE 5.21 – Diagnostic de simulation numérique avec un contrôle en boucle fermée, pour les mêmes paramètres que la figure 5.19. On note que le contrôle introduit une sélection, car sans lui chacun aurait la même masse finale.

### 5.5.2 Abaques multi follicules

Dans ce chapitre, on définit des abaques de visualisation, afin d'aider à régler les paramètres a priori, et en particulier à s'assurer d'une certaine hétérogénéité.

**Abaque à deux follicules.** Dans un premier temps, nous traitons le cas de deux follicules. On peut s'inspirer de la Figure 5.3. A chaque follicule, on associe une famille de courbes d'une couleur différente, indexée par différentes valeurs du contrôle qu'on fait varier entre les valeurs atteintes en temps long pour les deux comportements extrêmes, ovulatoire et atrétique. L'idée est de prédire, a priori, le comportement asymptotique des racines de  $h$ . Dans le cas d'un follicule ovulatoire, on a  $\lim_{t \rightarrow \infty} b_f(t) = 1$ , donc  $\lim_{t \rightarrow \infty} u_f(t) = U(t)$  (et en pratique  $U(t) = U_{\min}$ ). Dans le cas d'un follicule atrétique, on a  $\lim_{t \rightarrow \infty} b_f(t) = (b_1 + \frac{1}{b_3})$ , donc  $\lim_{t \rightarrow \infty} u_f(t) = (b_1 + \frac{1}{b_3})U_{\min}$  (défini précédemment comme  $\bar{u}_{\min}$ ). On voit apparaître une notion nouvelle, celle de chevauchement, portion commune entre deux follicules. Un exemple est donné sur la Figure 5.22.

**Abaque multi follicule (1ère espèce).** En pratique nous voulons simuler des situations à environ 20 follicules, pour lesquelles une abaque du style représenté

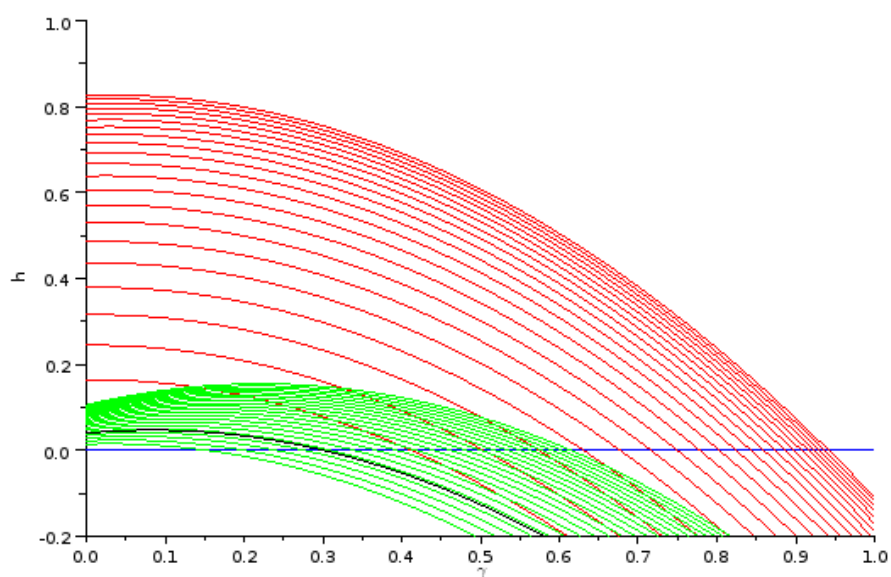


FIGURE 5.22 – Abaque de fonction  $h_f$  à deux follicules. En abscisse, la maturité cellulaire, en ordonnée, la vitesse de maturation pour différentes valeurs du contrôle. La fonction  $b_f$  est identique dans les deux cas. Les valeurs extrêmes du contrôle sont  $\bar{u}_{\min} = 0.0060517$  et  $U_{\min} = 0.075$ . Les courbes rouges correspondent à  $c_1 = 0.07$ ,  $c_2 = 0.9$  et  $\bar{u} = 0.03$ . Les courbes vertes correspondent à  $c_1 = 0.62$ ,  $c_2 = 0.168$  et  $\bar{u} = 0.06$ . Pour ce jeu de paramètres les valeurs de contrôle  $u_f$  telles que  $\bar{u}_{\min} \leq u_f \leq u_a = 0.0188384$  correspondent à une racine  $r^+(u_f) < \gamma_s$  donc à une situation d'équilibre dans la zone de prolifération. On a matérialisé en noir la courbe correspondant au contrôle  $u_a$ .

sur la Figure 5.22 devient difficile à interpréter. Nous allons alors en définir un autre type, plus agrégé, permettant de déterminer les comportements par rapport au contrôle, ainsi que les chevauchement observés dans le cas de deux follicules. Sur la Figure 5.25, nous représentons quatre sous-panels.

- Le premier panel représente le maximum de la fonction  $h_f$  pour différentes valeurs de  $\gamma$  à contrôle maximal. Cela permet d'évaluer les différences de vitesses de maturation.
- Sur le second panel, on représente le terme source, sur une coupe  $a \in G1$  du plan âge maturité, pour un contrôle minimal. Cela permet de visualiser l'intensité et l'étendue du terme source.
- Enfin, les deux panels restant permettent d'évaluer l'hétérogénéité inter folliculaire, pour la fonction  $g_f$  et pour la fonction  $h_f$ .

Leur construction repose sur le même principe, c'est pourquoi nous les présenterons conjointement. En abscisse, on indique le numéro de follicule, et en ordonnée, la valeur de la fonction (vitesse de maturation cellulaire à gauche, vitesse de vieillissement à droite). A chaque follicule on associe un bloc, délimité en abscisse par le numéro de follicule, et en ordonnée par des indicateurs correspondants à différentes valeurs de contrôle. Précisément, il s'agit des racines de la fonction  $h_f$  à gauche, et des valeurs de la fonction  $g_f$  en zone  $G1$ , à droite. On sépare ces blocs en deux parties, la partie rouge correspond à des valeurs de contrôle défavorables ( $\bar{u}_{\min} \leq u_f \leq U_{\min}$ ) et la partie verte à des valeurs de contrôle favorables ( $U_{\min} \leq u_f \leq U_{\max}$ ). Une lecture verticale permet de connaître des informations locales sur un follicule, et une lecture horizontale permet de quantifier l'hétérogénéité.

**Abaque multi follicule (2ème espèce).** Une autre option consiste à généraliser l'abaque à deux follicules en fixant la valeur du contrôle. On a alors une famille de courbes, indexées non plus par la valeur de contrôle, mais par le numéro de follicule. Cette représentation est très pratique pour visualiser les différences d'amplitude dans les fonctions  $h_f$ . On donne un exemple dans la Figure 5.26.

### 5.5.3 Exemple de situation de référence à 10 follicules

En combinant les résultats précédents, on définit une méthode en deux étapes pour calibrer une simulation multi follicules.

1. Contrôle en boucle ouverte
  - Détermination de la maturité asymptotique ;
  - Détermination des paramètres de cinétique cellulaire ;
2. Contrôle en boucle fermée
  - Réglage de la pression sélective ;
  - Réglage de l'hétérogénéité.

A titre d'exemple nous allons utiliser cet algorithme pour identifier une situation de référence à 10 follicules.



## Chapitre 5. Calibration

---

La Figure 5.23 représente le résultat d'une simulation d'une cohorte de follicules ayant tous les mêmes paramètres (paramètres par défaut), excepté le paramètre de travail  $p$ . Plus précisément :  $p = 0.48, 0.485, 0.49, 0.495, 0.5, 0.51, 0.52, 0.53, 0.54, 2.75$ . D'autre part, on prend le paramètre de travail  $C_p = 0.6$  pour tous les follicules, et  $K = 0.6, \bar{M} = 25, c = 0.5, \gamma_s^+ = 0.365$ .

On note que seul l'un des follicules est ovulatoire alors que les autres sont atrétiques. Ce sont des résultats préliminaires illustrant l'effet de l'hétérogénéité sur un seul paramètre. Il est intéressant de comparer ces résultats avec ceux de la Figure 5.24. En effet, cette dernière est le résultat de la même simulation, mais sans pression sélective. On note que dans ce cas, tous les follicules sont ovulatoires.

La Figure 5.25 présente l'abaque multi follicule associée. On note que le  $h$  maximum est bien différencié dans le cas du follicule ovulatoire, comparé à ceux qui sont atrétiques. Les racines sont pourtant positionnées au même endroit (ce qui assure la non prédestination dans le cas à contrôle maximal constant).

La Figure 5.26, qui représente les valeurs de toutes les fonctions  $h_f$  pour  $f = 1, \dots, N_f$  en fonction de  $\gamma$  pour le contrôle  $u_f = U_{\min}$  apporte un autre éclairage sur la situation. Pour la valeur de contrôle sélectionnée ( $u_f = U_{\min}$ , et il en serait de même pour toute valeur de  $u_f$ ), les racines positives des fonctions  $h_f$  sont toutes identiques. En revanche, les amplitudes des fonctions  $h_f$  pour une valeur de  $\gamma$  donnée diffèrent. Les follicules atrétiques ont une vitesse de maturation faible par rapport à celle du follicule ovulatoire. Le rapport entre les maxima des vitesses de maturation varie entre 4 et 5 suivant la valeur du contrôle.

**Conclusion** Cette étude est amenée à être poursuivie dans de futurs travaux, et notamment la recherche de diversification des sources d'hétérogénéité pour obtenir des situations de sélection avec des valeurs de  $\tau_f$  moins contrastées.

## Chapitre 5. Calibration

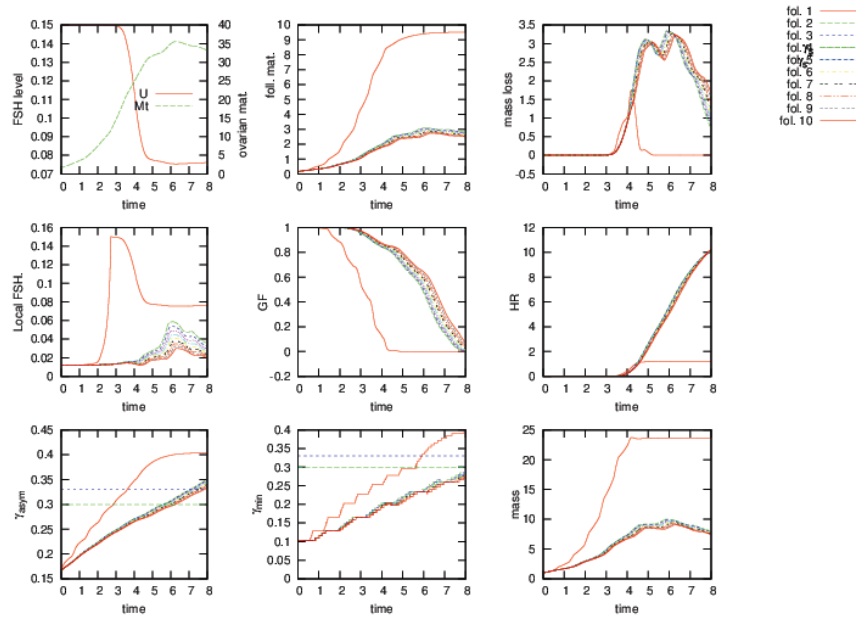


FIGURE 5.23 – Panel de diagnostic de simulation numérique pour la situation de référence à 10 follicules. Tous les follicules ont les mêmes paramètres excepté le paramètre de cinétique cellulaire  $p$ . Avec le contrôle en boucle fermée (autrement dit avec la sélection), seul l'un d'entre eux est ovulatoire, les autres devenant atrétiques.

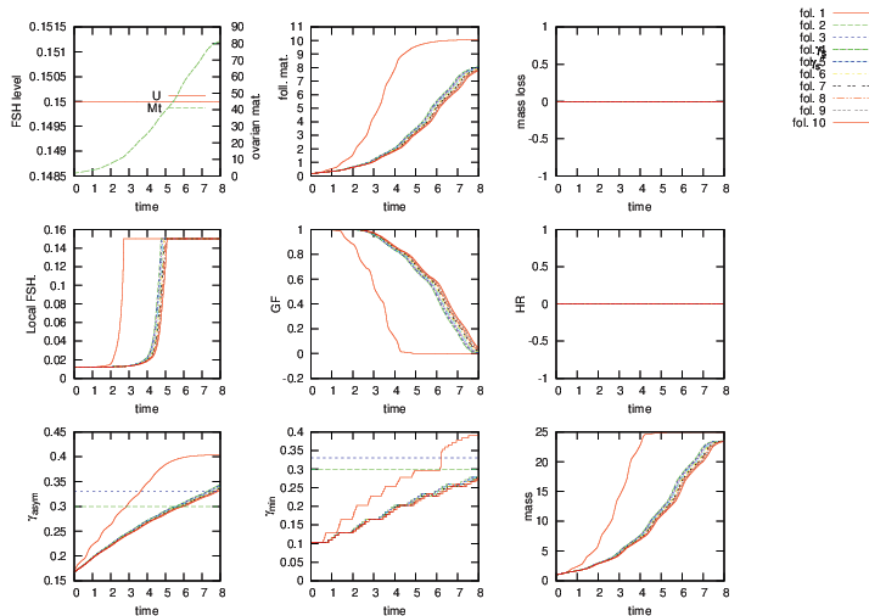


FIGURE 5.24 – Panel de diagnostic de simulation numérique pour le même jeu de paramètres que pour la Figure 5.23, mais à contrôle maximal constant. Contrairement à la Figure 5.23, il n'y a pas de sélection, et l'on observe que tous les follicules sont ovulatoires.

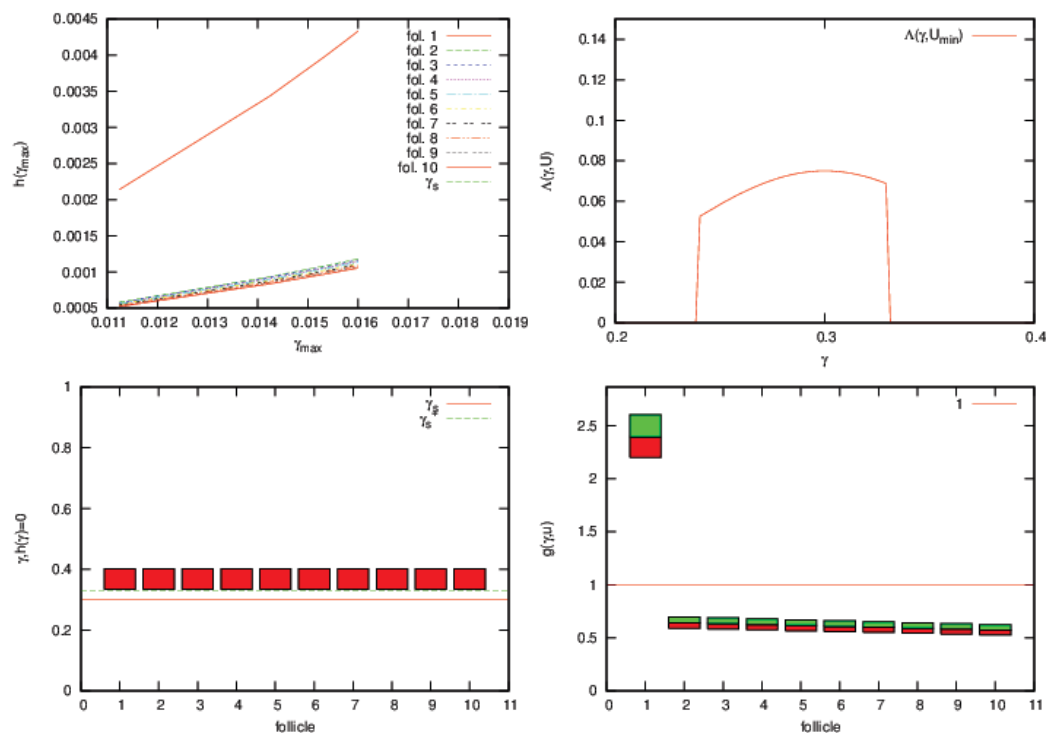


FIGURE 5.25 – Abaque multi follicules associée aux Figures 5.23, 5.24. Dans le premier panel, on représente les courbes d’amplitude maximales de  $h_f$  en fonction de  $\gamma_{\max}$ , une par follicule. C’est un indicateur d’hétérogénéité entre les follicules. Dans le second, en haut à droite, on trace une coupe verticale (i.e. à variable  $a$  fixée en zone G1), de la fonction déterminant le terme source. On note qu’elle est maximale à l’interface positionnée en  $\gamma = \gamma_s$  (ici  $\gamma_s = 0.3$ ). Dans le troisième et le quatrième, on représente des cubes, chacun associé à un follicule, avec en abscisse le numéro de follicule, et en ordonnée la valeur de la fonction observée (panel en bas à gauche, vitesse de maturation cellulaire, en bas à droite, vitesse de vieillissement). Chacun de ces blocs est délimité verticalement par deux valeurs, correspondant à deux valeurs distinctes de contrôle : un contrôle favorable, et un contrôle défavorable. Ces figures sont deux autres indicateurs d’hétérogénéité entre les follicules.

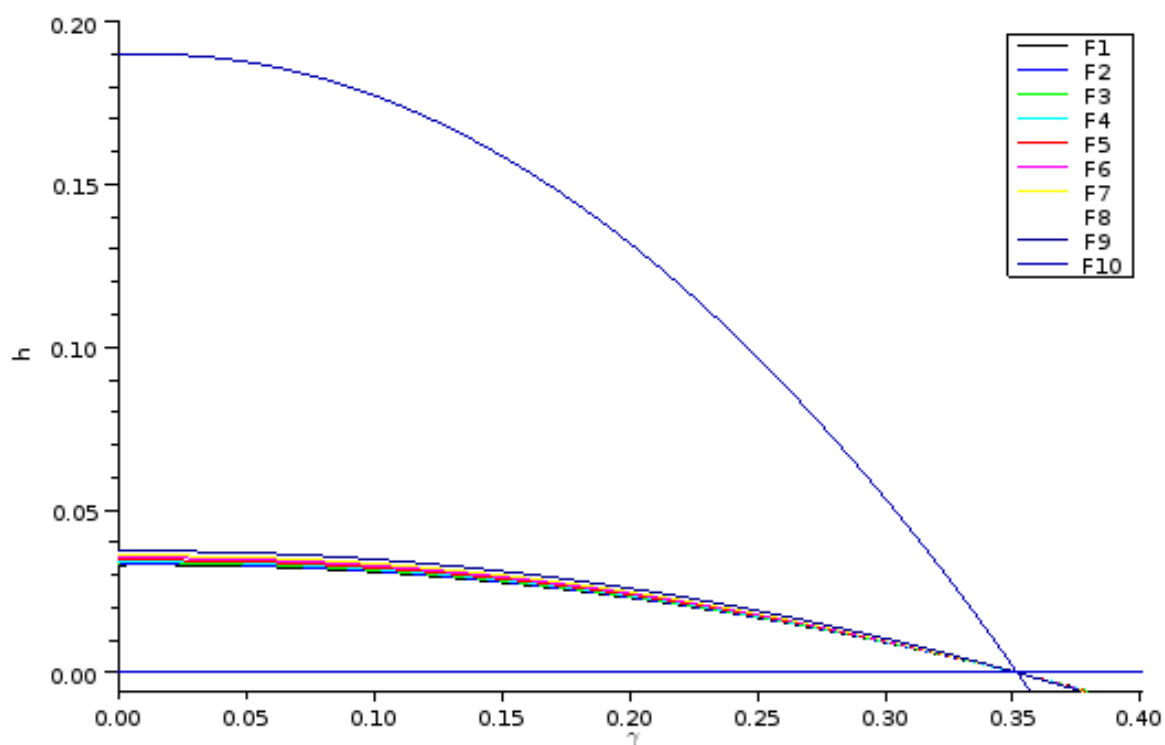


FIGURE 5.26 – Abaque de chevauchement à 10 follicules. Graphes de la fonction de maturation  $h_f$  pour  $f = 1, \dots, 10$ . Les zéros de  $h_f$  sont les mêmes pour tous les follicules, mais les amplitudes de la fonction sont différentes (en effet, seul le paramètre  $\tau_h$  varie). Sans pression sélective, cela n'a pas d'effet sur la trajectoire folliculaire, mais en introduisant de la sélection, on observe que seul le follicule avec une grande vitesse de maturation devient ovulatoire, les autres devenant atrétiques.



# Design and performance analysis of a multiscale scheme for controlled population dynamics

---

## Sommaire

---

<b>6.1 Multiscale model</b> . . . . .	<b>164</b>
6.1.1 Example : selection process of ovarian follicles . . . . .	166
<b>6.2 Multiscale numerical method</b> . . . . .	<b>169</b>
6.2.1 Microscopic level : adaptive finite volume scheme . . . . .	170
6.2.2 Mesoscopic and macroscopic scale : Numerical moments and control. . . . .	181
6.2.3 Parallel computing . . . . .	182
<b>6.3 Performance analysis</b> . . . . .	<b>184</b>
6.3.1 Method . . . . .	186
6.3.2 Mean load balance test . . . . .	187
6.3.3 Mean weak scalability test . . . . .	191

---

**Abstract.** In this chapter, we discuss two main subjects. First, we introduce a numerical scheme for a general multiscale model of controlled population dynamics. Originally designed for the selection process of ovarian follicles, the method is presented in a general frame that may be afterward adapted to several situations. Second, we introduce a new method for analyzing the performance of a parallel code.

**Keywords.** Multiscale models, population dynamics, non-local control, finite volume schemes, adaptive mesh, multiresolution, high order scheme, parallel computing, performances analysis, weak scalability, load balance, Monte Carlo method, uncertainty propagation.

**Introduction.** In this chapter, we are concerned with multiscale controlled population dynamics, and present two main results. First, we introduce a new multiscale method, and second we present a new method for performance analysis of a parallel code.

Originally, the multiscale numerical method was designed for the model of the selection process of ovarian follicles (see [41] and [40] ). The mathematical model consists

## Chapitre 6. Design and performance analysis of a multiscale scheme for controlled population dynamics

---

in a system of transport equations in 2D, with a linear source term. This model embeds a close loop control acting on the velocities and source term and defined from one of the first order moment of the solutions, so that the equations are weakly coupled, see Figure 6.1. Non conservative flux conditions at the internal boundaries between sub regions of the domain of interest are another original feature..

From the very beginning, the numerical simulation of this model was a problem (see the perspectives of the PhD thesis [39]).

A performant code was needed in order to run intensive simulations for calibration and uncertainty quantification on one hand, and to enable one to do real time computation within a "biologist friendly environment" on the other hand. From this point, it was necessary to find methods that would speed up the computations. To do so, we designed a dedicated scheme, involving parallel computing and adaptive meshing. We also enhanced the precision with a third order scheme. It takes now around 2 minutes on a cluster to obtain the solution of a simulation corresponding to a biologically realistic configuration (see Figure 6.14). This is a tremendous improvement with respect to the initial simulations ([39]) which would have required several days for the same set-up.

In order to evaluate the performances of this parallel code, the weak scalability test is the most adapted to our set-up. However, its results are relative since, for reasons that we will develop later, the CPU requirement are very sensitive to the parameters of the simulation. We then proposed a generalization, combining the classical method and Monte Carlo methods, inspired from uncertainty quantification. It consists in perturbing a reference situation, and evaluating the impact on the performances.

The outline of this article is the following. In the first section, we present the class of models considered in this work, namely the multiscale controlled population dynamics. We note that such models may be written under the form of a block diagram (see Figure 6.1). In the second section, we present the multiscale numerical method, together with the general SIMD algorithm in the last part of the section. Finally, we discuss the tools that we introduced for performance analysis, and apply them to the model of the selection process of ovarian follicles.

### 6.1 Multiscale model

**Conservation law.** The evolution of the sub-population densities is given by the following conservation law with associated initial condition

$$\begin{cases} \partial_t \phi_f(x, y, t) + \partial_x g_f(x, y, u_f) \phi_f(x, y, t) + \partial_y h_f(x, y, u_f) \phi_f(x, y, t) \\ \qquad \qquad \qquad = -\Lambda(x, y, U) \phi_f(x, y, t), \\ \phi_f(x, y, 0) = \phi_f^0(x, y), \end{cases}$$

where  $g_f$  and  $h_f$  are velocity functions,  $\Lambda$  is the source term,  $U$  is the global control and  $u_f$  the control at the mesoscopic level.

The velocity functions depend on the values of the non-local control. In the biological model, both controls depend on time, through the first order moment of the solution

**Chapitre 6. Design and performance analysis of a multiscale scheme for controlled population dynamics**

---

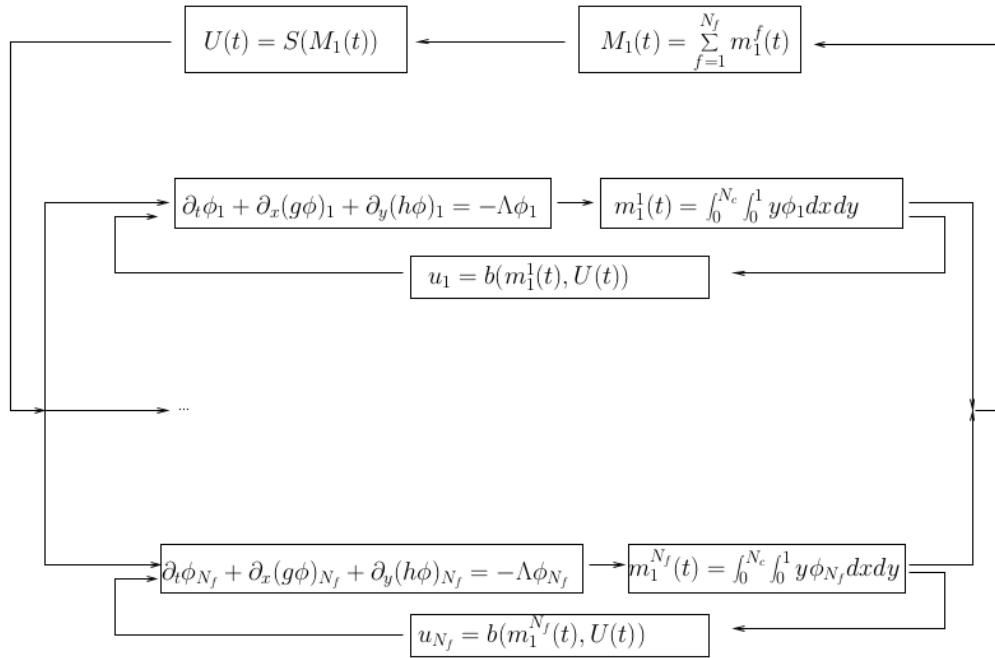


FIGURE 6.1 – Block diagram representation of the multiscale model for controlled population dynamics. The evolution of the density  $\phi_f$  each subpopulation  $f \in \{1, \dots, N_f\}$  is governed by a conservation law. For each box of conservation law, the inputs are the macroscopic control  $U$ , and a mesoscopic control  $u_f$ , that is the macroscopic control  $U$  filtered by a function  $b_f$  depending on the first order moment  $m_1^f$  of the density. The input of the macroscopic control  $U$  is the sum  $M_1$  of the first order moments of all subpopulation densities. For the sake of readability, we have represented only the boxes for two subpopulation,  $f = 1$  and  $f = N_f$ .



## Chapitre 6. Design and performance analysis of a multiscale scheme for controlled population dynamics

---

in the  $y$ -direction. They can also be defined as explicit functions of time in an open loop control version.

The source term models a birth term when it is positive, or a loss term when it is negative. It can also be both, for example the sum of a birth term and a loss term.

**Moments.** The sub-populations moments are the link between the micro scale and the mesoscopic scale. We compute the first order moments of each sub-population with the following formula

$$m_1^f(t) = \iint_{[0, N_c] \times [0, 1]} y \phi_f(x, y, t) dx dy. \quad (6.1)$$

The link between the mesoscopic scale and the macroscopic scale is defined with the global moment :

$$M_1(t) = \sum_{f=1}^{N_f} m_1^f(t). \quad (6.2)$$

**Control.** The control functions are non-local functions in the sense that they are defined with respect to the moments of the densities. The global control function uses information from the sum of the first order moments only. Each sub-population is controlled at the mesoscopic level by a closed loop control, where the global control also plays a part.

### 6.1.1 Example : selection process of ovarian follicles

An example of such models is the selection process of ovarian follicles introduced in [40] and [41], which we briefly describe in this paragraph. The unknowns in this case are the granulosa cell densities of each follicle, denoted  $\phi_f(x, y, t)$ , with  $f \in \{1, \dots, N_f\}$ .  $N_f$  has a value around 20. The computational domain is  $\Omega = [0, N_c] \times [0, 1]$ , with  $N_c$  the number of cell cycles. The space variables  $x$  and  $y$  are functional variables denoting respectively the age of the cells within the cell cycle and their maturity. An horizontal boundary at  $y = \gamma_s$  splits the domain into a proliferating part (bottom), and a differentiated part (top). The corresponding sub-domains are defined as follows :

$$\begin{cases} G1 = \{(x, y) \in \Omega, \quad k \leq x \leq (k + 1/2), \quad k = 0, \dots, N_c - 1, \quad 0 \leq y \leq \gamma_s\}, \\ SM = \{(x, y) \in \Omega, \quad (k + 1/2) \leq x \leq (k + 1), \quad k = 0, \dots, N_c - 1, \quad 0 \leq y \leq \gamma_s\}, \\ D = \{(x, y) \in \Omega, \quad \gamma_s \leq y\}. \end{cases}$$

## Chapitre 6. Design and performance analysis of a multiscale scheme for controlled population dynamics

---

Here, the speed functions and the source term are defined by

$$\left\{ \begin{array}{l} g_f(x, y, u) = \begin{cases} p(u + \alpha) & \text{for } (x, y) \in G1, \\ 1 & \text{for } (x, y) \in SM \cup D, \end{cases} \\ h_f(x, y, u) = \begin{cases} C_p \times p \times (-y^2 + (c_1 y + c_2)(1 - \exp(\frac{-u}{\bar{u}}))) & \text{for } (x, y) \in G1 \cup D, \\ 0 & \text{for } (x, y) \in SM, \end{cases} \\ \Lambda(x, y, U) = \begin{cases} -K \times \exp(-(\frac{y - \gamma_s}{\bar{\gamma}})^2) \times \frac{(U_{\max} - U)}{U_{\max}} & \text{for } (x, y) \in G1 \cup D, \\ 0 & \text{for } (x, y) \in SM, \end{cases} \end{array} \right. \quad (6.3)$$

where  $p, \alpha, C_p, c_1, c_2, \bar{u}, U_{\max}, K$  and  $\bar{\gamma}$  are positive real constants.

The functions  $S$  and  $b_f$  defining the macroscopic and mesoscopic controls respectively in the functional diagram (see Figure 6.1) are given by

$$\left\{ \begin{array}{l} S(M_1) = U_{\min} + \frac{U_{\max} - U_{\min}}{1 + \exp(c(M_1 - \bar{M}))}, \\ b_f(m_1^f) = \min \left( b_1 + \frac{\exp(b_2 m_1^f)}{b_3}, 1 \right), \end{array} \right. \quad (6.4)$$

where  $U_{\min}, c, \bar{M}, b_1, b_2$  and  $b_3$  are positive real constants.

The initial conditions are given by

$$\phi_f^0(x, y) = \begin{cases} \mu_1 \frac{c_k}{1 - \exp^{-c_k}} \exp(-c_k x) & \text{if } \gamma_1 < y < \gamma_2 \quad \text{and} \quad 0 < x < 1, \\ \mu_2 \frac{c_k}{1 - \exp^{-c_k}} \exp(-c_k x) & \text{if } \gamma_2 < y < \gamma_3 \quad \text{and} \quad 0 < x < 1, \\ \mu_3 \frac{c_k}{1 - \exp^{-c_k}} \exp(-c_k x) & \text{if } \gamma_3 < y < \gamma_4 \quad \text{and} \quad 0 < x < 1, \\ 0 & \text{otherwise,} \end{cases} \quad (6.5)$$

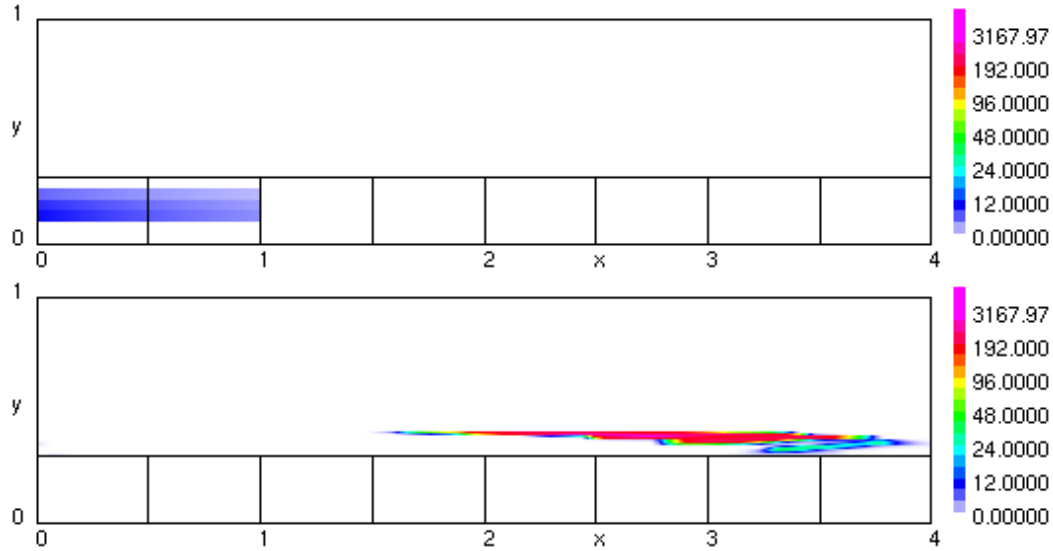
with  $\gamma_1 < \gamma_2 < \gamma_3 < \gamma_4 < \gamma_s$ ,  $\mu_1, \mu_2, \mu_3$  and  $c_k$  positive real constants. The population index for velocities  $g_f, h_f$ , control  $u_f$  and initial condition  $\phi_f^0$  indicates the possible heterogeneity among the follicles which will be introduced by the parameters  $p, \alpha, C_p, c_1, c_2, \bar{u}, b_1, b_2, b_3, (\gamma_i)_{i=1,2,3}, (\mu_i)_{i=1,2,3}, c_k$ , which can vary for the different follicles, although we do not index them by  $f$  to alleviate notations. We further associate transmission conditions between the sub-domains, as follows :

$$\left\{ \begin{array}{l} \phi_f(t, x^+, y) = p(u_f + \alpha)\phi_f(t, x^-, y), \quad x = k - 1/2, \quad 0 \leq y \leq \gamma_s, \\ p(u_f + \alpha)\phi_f(t, x^+, y) = 2\phi_f(t, x^-, y), \quad x = k, \quad 0 \leq y \leq \gamma_s, \\ \phi_f(t, x, \gamma_s^+) = 0, \quad k - 1/2 \leq x \leq k. \end{array} \right.$$

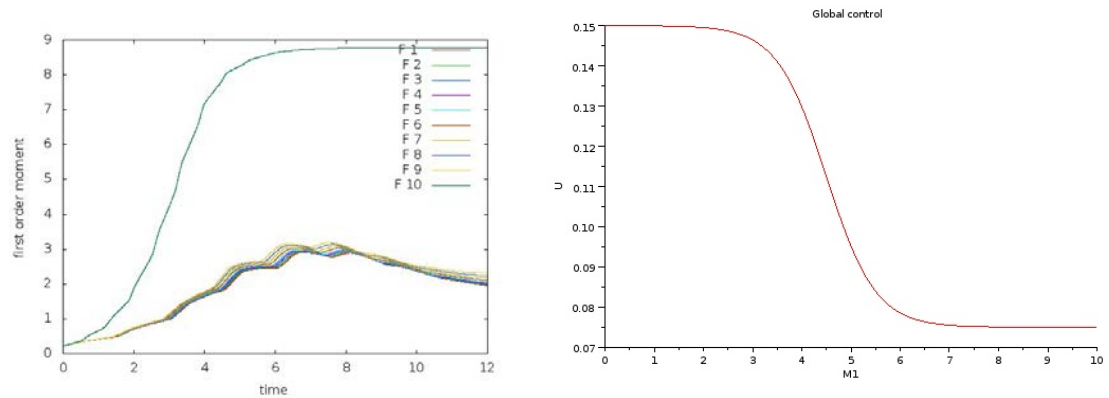
The second condition is a doubling flux, modeling cell mitosis.

As an example, on Figure 6.2, we present a result of a numerical simulation of this model. In panel a), we represent with a color code the cell density of one of the follicle on the microscopic level at initial time (top) and at a later time  $t = 4$  (bottom),

Chapitre 6. Design and performance analysis of a multiscale scheme for controlled population dynamics



a) Microscopic level : snapshots of the cell density of one follicle. At initial time (top panel) all the cells are in the first proliferation cycle. At a later time ( $t = 5$ , bottom panel) most of the population has exited the proliferation stage ( $y < \gamma_s$ ). The same color code is used at both times in order to highlight the increase in the size of the population due to the successive mitoses.



b) Mesoscopic level : evolution of the first order moments of the densities

c) Macroscopic level : Evolution of the global closed loop function.

FIGURE 6.2 – Example of the model of the follicle selection process. We note that as the global control decreases (right panel), all the follicle maturities decrease, except for one of them, that is selected (left panel).

## Chapitre 6. Design and performance analysis of a multiscale scheme for controlled population dynamics

---

with the age  $x$  in abscissa and cell maturity  $y$  in ordinate. The sub-domains (cells cycles and differentiation phase) are bounded with straight lines. In this model the mesoscopic moments (eq 6.1) are indicators of the follicle maturities and the macroscopic moment (eq 6.2) represents the ovarian maturity. In panel b) of Figure 6.2, we display the follicle maturities as a function of time for all ten follicles of the numerical experiment. In panel c) we display the global control function  $U = S(M_1)$  as a function of the maturity.

For these specific speed functions, an initial density localized in the bottom left of the computational domain (cycle 1) is transported to the top right of the domain (see Figure 6.2, panel a)). During this process, the mesoscopic moments begin by increasing, because the density crosses, several times, a mitosis boundary, where the density is locally doubled (see Figure 6.2, panel b)). Once the density has reached the upper part of the domain ( $y > \gamma_s$ ), the cells stop proliferating, the maturities either stabilize or decrease due to cell apoptosis. At the beginning, the global control is high, but around  $t = 4.5$ , it becomes sensitive to the increase in the global maturity and decreases steadily until reaching its lowest level (see Figure 6.2, bottom right panel). The minimum and maximum values of the global control  $U$  in equation 6.4 are for this simulation  $U_{\min} = 0.075$  and  $U_{\max} = 0.15$ . They are asymptotically reached for values of the global moment  $M_1$  well within the range of the final and initial values of this quantity  $M_1(t)$  as a function of time. Therefore for this choice of parameters  $U(t)$ , represented in panel c) of Figure 6.2 has the same shape as  $S(M)$ . We will not go here into the details of the underlying biological model, which is the topic addressed in the Chapter "Calibration". However, it is interesting to note that after the control has dropped to half its initial value only one follicle keep a stationary maturity while all other follicular maturities decrease. This is a good instance of a selection process triggered by hormonal control.

**Conclusion.** In this section, we have presented a class of models for controlled multiscale population dynamics. This model may be adapted to many other scenarios, either by changing the conservation law, or by changing the controls.

### 6.2 Multiscale numerical method

In this section, we present the numerical method that we have designed in order to simulate the multiscale model presented in the first section.

**Size of the problem.** In order to fix ideas, let us precise the size of the problem in a typical realistic situation in the biological model. We have  $N_f$  coupled PDEs in a 2D computational domain that is a compact subset of  $\mathbb{R}^2$ , namely  $[0, 1] \times [0, N_c]$ , with  $N_c$  of order 10. If we take  $100 \times 100$  grid cells to discretize the domain  $[0, 1] \times [0, 1]$ , then we have  $100 \times 100 \times N_c$  mesh cells for each sub-population hence  $100 \times 100 \times N_c \times N_f$  grid cells for the whole population. Setting  $N_c = 10$  and  $N_f = 20$ , we

## Chapitre 6. Design and performance analysis of a multiscale scheme for controlled population dynamics

---

then have  $2 \times 10^6$  grid cells, to be updated at each time step. It is necessary to use speedup techniques in order to simulate the problem in a reasonable time.

Note here that the size of the computational domain in  $x$ , which denote the cells age within the successive cell cycles, can be any integer value (we have conventionally set the size of one cell cycle to 1). If we want to keep track of the cells' history, at the microscopic scale, we can use the same number of cell cycles as that needed to follow a whole simulation. One may be only interested in the mesoscopic and macroscopic outputs (size of the population, maturities, growth fraction, etc) as a function of time. In that case we may use only one cell cycle that repeats with periodic boundary conditions. In practice, we often use an intermediate solution, with sufficient successive cell cycles to avoid any ambiguity on the age of the cells (which would occur if densities belonging to different cycles could be superimposed due to the periodicity).

### 6.2.1 Microscopic level : adaptive finite volume scheme

At a microscopic level, the evolution of the densities is given by a system of transport equations. A classical method used for this type of systems is a finite volume scheme. In this section we present the principles of adaptive finite volume schemes using multiresolution. For the sake of simplicity, we only present the case of two levels of grid, with a simple first order scheme. The generalization will come in the next paragraph. The reader can refer to [36] and [11] for bases on finite volume schemes and to [45], [62] and [30] for adaptive mesh refinement in the context of finite volume schemes.

**Hierarchy of nested grids.** We define a sequence of nested grids (see Figure 6.3), for  $\ell = 0, \dots, L - 1$ ,

$$\begin{cases} x_i^\ell = i\Delta x^\ell \\ y_j^\ell = j\Delta x^\ell \\ \Omega_{i,j}^\ell = [x_i^\ell, x_{i+1}^\ell] \times [y_j^\ell, y_{j+1}^\ell], \end{cases}$$

with  $i \in \{0, \dots, N_x\}$ ,  $j \in \{0, \dots, N_y\}$  and  $\Delta x^\ell = \Delta x 2^{(L-1)-\ell}$ . We have the following property :

$$\Omega_{i,j}^\ell = \Omega_{2i,2j}^{\ell+1} \cup \Omega_{2i+1,2j}^{\ell+1} \cup \Omega_{2i,2j+1}^{\ell+1} \cup \Omega_{2i+1,2j+1}^{\ell+1}$$

We say that  $\Omega_{i,j}^\ell$  is a mother grid cell, and  $\{\Omega_{2i,2j}^{\ell+1}, \Omega_{2i+1,2j}^{\ell+1}, \Omega_{2i,2j+1}^{\ell+1}, \Omega_{2i+1,2j+1}^{\ell+1}\}$  is the set of corresponding daughter grid cells.

**Finite volume scheme.** The FV approximation of the solution is defined on level  $\ell$  by

$$\phi_{i,j}^{n,\ell} \approx \frac{1}{(\Delta x^\ell)^2} \int_{\Omega_{i,j}^\ell} \phi(x, y, n\Delta t) dx dy,$$

**Chapitre 6. Design and performance analysis of a multiscale scheme for controlled population dynamics**

---

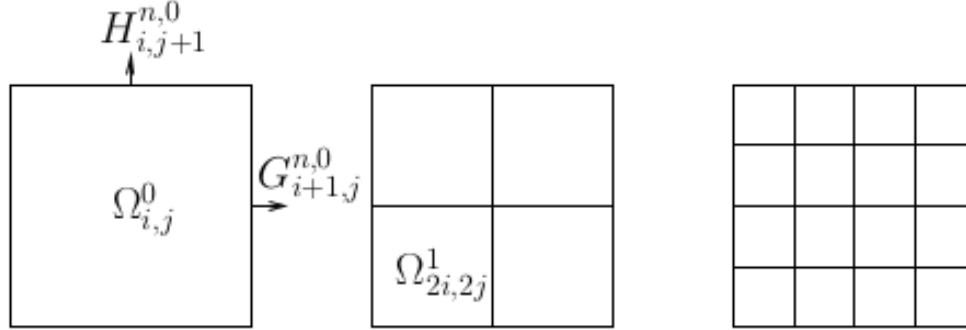


FIGURE 6.3 – 2d nested grids for multiresolution. From left to right : level 0, level 1 and level 2.

The finite volume scheme at level  $\ell$  is defined as

$$\phi_{i,j}^{n+1,\ell} = \phi_{i,j}^{n,\ell} - \frac{\Delta t^n}{\Delta x^\ell} (G_{i+1,j}^{n,\ell} - G_{i,j}^{n,\ell} + H_{i,j+1}^{n,\ell} - H_{i,j}^{n,\ell}) - \Delta t^n \Lambda_{i,j}^{n,\ell} \phi_{i,j}^{n,\ell} \quad (6.6)$$

with

$$\begin{cases} G_{i,j}^{n,\ell} \approx \frac{1}{\Delta t^n} \frac{1}{\Delta x^\ell} \iint_{[t^n, t^n + \Delta t] \times [y_j^\ell, y_{j+1}^\ell]} g(x_i^\ell, y, u) \phi(x_i^\ell, y, t) dy dt, \\ H_{i,j}^{n,\ell} \approx \frac{1}{\Delta t^n} \frac{1}{\Delta x^\ell} \iint_{[t^n, t^n + \Delta t] \times [x_i^\ell, x_{i+1}^\ell]} h(x, y_j^\ell, u) \phi(x, y_j^\ell, t) dx dt, \end{cases} \quad (6.7)$$

the numerical fluxes at the vertical boundaries and horizontal boundaries respectively. We would like to insist on the fact that this notation departs from the standard notations in Finite Volume on uniform grids. It is however commonly used in the multiresolution framework where the dyadic embedding of grid levels renders the use of half indices clumsy. Mean values over meshes and meshes themselves are indexed with the indices of the lower left corner. Mean values over vertical (respectively horizontal) edges are indexed with the indices of the bottom (respectively left) end of the edge. The choice of the approximation for the numerical fluxes in 6.7 determines the precision of the scheme. For instance, a first order upwind flux gives

$$\begin{cases} G_{i,j}^{n,\ell} = (g_{i-1,j}^{n,\ell})^+ \phi_{i-1,j}^{n,\ell} + (g_{i,j}^{n,\ell})^- \phi_{i,j}^{n,\ell} \\ H_{i,j}^{n,\ell} = (h_{i,j-1}^{n,\ell})^+ \phi_{i,j-1}^{n,\ell} + (h_{i,j}^{n,\ell})^- \phi_{i,j}^{n,\ell} \end{cases}$$

with the notations  $(f)^+ = \sup(f, 0)$ ,  $(f)^- = \inf(f, 0)$ . The time step is computed at each time step with the following formula

$$\Delta t^n = \lambda \min_{i,j,\ell} \left\{ \frac{\Delta x^\ell}{(|g_{i,j}^{n,\ell}|)}, \frac{\Delta x^\ell}{(|h_{i,j}^{n,\ell}|)} \right\}, \quad (6.8)$$

where  $\lambda$  must satisfy  $0 < \lambda \leq 1$  in order to respect stability condition. An additional restriction is imposed to the time step to ensure the stability of the source term action (see [6] for details).

## Chapitre 6. Design and performance analysis of a multiscale scheme for controlled population dynamics

---

**Multiresolution.** In this paragraph, we present the basis of the multiresolution analysis. Since we are interested in adaptive finite volume schemes, we use a multiresolution approach based on a mean value representation of the solution, by contrast to point values decomposition scheme (see [10] for more details).

For the sake of simplicity, we will consider only two levels of multiresolution in order to introduce the main concepts.

The solution in a mother cell can be computed from the daughter cells with the following formula

$$\phi_{i,j}^{n,\ell} = \frac{1}{4} \left( \phi_{2i,2j}^{n,\ell+1} + \phi_{2i+1,2j}^{n,\ell+1} + \phi_{2i,2j+1}^{n,\ell+1} + \phi_{2i+1,2j+1}^{n,\ell+1} \right). \quad (6.9)$$

This formula defines a 2D projection operator.

Knowing the mean values of the solution on the coarser level, approximated mean values on the finer level (daughter cells) may be obtained by second order reconstruction :

$$\begin{aligned} \hat{\phi}_{2i+1,2j}^{\ell+1} &= -\frac{1}{64}\phi_{i-1,j-1}^{\ell} + \frac{1}{8}\phi_{i,j-1}^{\ell} + \frac{1}{64}\phi_{i+1,j-1}^{\ell} \\ &\quad -\frac{1}{8}\phi_{i-1,j}^{\ell} + \phi_{i,j}^{\ell} + \frac{1}{8}\phi_{i+1,j}^{\ell} \\ &\quad +\frac{1}{64}\phi_{i-1,j+1}^{\ell} - \frac{1}{8}\phi_{i,j+1}^{\ell} - \frac{1}{64}\phi_{i+1,j+1}^{\ell} \\ \hat{\phi}_{2i,2j+1}^{\ell+1} &= -\frac{1}{64}\phi_{i-1,j-1}^{\ell} - \frac{1}{8}\phi_{i,j-1}^{\ell} + \frac{1}{64}\phi_{i+1,j-1}^{\ell} \\ &\quad +\frac{1}{8}\phi_{i-1,j}^{\ell} + \phi_{i,j}^{\ell} - \frac{1}{8}\phi_{i+1,j}^{\ell} \\ &\quad +\frac{1}{64}\phi_{i-1,j+1}^{\ell} + \frac{1}{8}\phi_{i,j+1}^{\ell} - \frac{1}{64}\phi_{i+1,j+1}^{\ell} \\ \hat{\phi}_{2i+1,2j+1}^{\ell+1} &= \frac{1}{64}\phi_{i-1,j-1}^{\ell} - \frac{1}{8}\phi_{i,j-1}^{\ell} - \frac{1}{64}\phi_{i+1,j-1}^{\ell} \\ &\quad -\frac{1}{8}\phi_{i-1,j}^{\ell} + \phi_{i,j}^{\ell} + \frac{1}{8}\phi_{i+1,j}^{\ell} \\ &\quad -\frac{1}{64}\phi_{i-1,j+1}^{\ell} + \frac{1}{8}\phi_{i,j+1}^{\ell} + \frac{1}{64}\phi_{i+1,j+1}^{\ell} \\ \hat{\phi}_{2i,2j}^{\ell+1} &= \phi_{i,j}^{\ell} - \hat{\phi}_{2i+1,2j}^{\ell+1} - \hat{\phi}_{2i,2j+1}^{\ell+1} - \hat{\phi}_{2i+1,2j+1}^{\ell+1} \end{aligned} \quad (6.10)$$

This defines a 2D prediction operator.

The error between real values and predicted values are called details (the terminology comes from image processing), and may be computed with the formula

$$d_{i,j}^{\ell} = \phi_{i,j}^{\ell} - \hat{\phi}_{i,j}^{\ell} \quad (6.11)$$

For each mother-daughters pair we define the change of basis, from mean values to details,

$$(\phi_{i,j}^{\ell+1}, \phi_{i+1,j}^{\ell+1}, \phi_{i,j+1}^{\ell+1}, \phi_{i+1,j+1}^{\ell+1}) \leftrightarrow (\phi_{i,j}^{\ell}, d_{i+1,j}^{\ell+1}, d_{i,j+1}^{\ell+1}, d_{i+1,j+1}^{\ell+1})$$

Note that due to property (6.9), it is sufficient to know only three details and the mean value of a mother grid cell to compute the value of the last detail. The corresponding algorithm (called "Encoding") in the general multilevel case is defined by Algorithm (3). This operation is invertible, and the corresponding algorithm (called "Decoding") in the general multilevel case is given by Algorithm (4).

## Chapitre 6. Design and performance analysis of a multiscale scheme for controlled population dynamics

---

We can define a hybrid grid based on a strategy consisting in replacing locally the solution by its second order polynomial approximation, when it is smooth enough. Indeed, approximation theory provides the relation between the details and the local smoothness ([29]). In the multiresolution framework the details are used as local regularity sensors to trigger or not refinement to the finer grid level. Given a threshold value  $\epsilon$ , for each mother-daughters pair, if all the details are under the threshold value

$$|d_{i+k,j+m}^{\ell+1}| \leq \epsilon, \quad \forall k, m \in \{0, 1\}, \quad (6.12)$$

then we can replace the details by 0 (local second order approximation), and delete the corresponding daughter cells.

In the general case of  $L$  levels, the threshold value is level dependent

$$\epsilon_\ell = \frac{\epsilon}{2^{(L-1)-\ell}}$$

In other words, each time we decrease the level, we divide the threshold value by 2. This amounts to a more cautious threshold at coarsest levels than at finest ones.

**Data:** Solution on hybrid grid  
**Result:** Solution coded on hybrid tree  
**for**  $\ell = (L - 1)$  **to** 1 **do**  
    **for** cells in level  $\ell$  **do**  
        Compute the mean values with the Projection operator (6.9)  
             $(\phi_{i,j}^{\ell+1}, \phi_{i+1,j}^{\ell+1}, \phi_{i,j+1}^{\ell+1}, \phi_{i+1,j+1}^{\ell+1}) \rightarrow (\phi_{i,j}^\ell, \phi_{i+1,j}^{\ell+1}, \phi_{i,j+1}^{\ell+1}, \phi_{i+1,j+1}^{\ell+1})$   
    **end**  
    **for** cells in level 1 **do**  
        Compute details with the Prediction operator (6.10) and detail formula (6.11)  
             $(\phi_{i,j}^\ell, \phi_{i+1,j}^{\ell+1}, \phi_{i,j+1}^{\ell+1}, \phi_{i+1,j+1}^{\ell+1}) \rightarrow (\phi_{i,j}^\ell, d_{i+1,j}^{\ell+1}, d_{i,j+1}^{\ell+1}, d_{i+1,j+1}^{\ell+1})$   
    **end**  
**end**

**Algorithm 3:** Solution encoding : from the physical basis (FVS) representation into multiresolution basis (wavelet).



```

Data: Solution coded on hybrid tree
Result: Solution on hybrid grid
for  $\ell = 1$  to  $(L - 1)$  do
  for cells in level  $\ell$  do
    Reconstruct three daughter cell values with the Prediction operator
    (6.10) and the detail formula (6.11)

     $(\phi_{i,j}^\ell, d_{i+1,j}^{\ell+1}, d_{i,j+1}^{\ell+1}, d_{i+1,j+1}^{\ell+1}) \rightarrow (\phi_{i,j}^\ell, \phi_{i+1,j}^{\ell+1}, \phi_{i,j+1}^{\ell+1}, \phi_{i+1,j+1}^{\ell+1})$ 

  end
  for cells in level  $\ell$  do
    Reconstruct last value with the Projection formula (6.9)

     $(\phi_{i,j}^\ell, \phi_{i+1,j}^{\ell+1}, \phi_{i,j+1}^{\ell+1}, \phi_{i+1,j+1}^{\ell+1}) \rightarrow (\phi_{i,j}^{\ell+1}, \phi_{i+1,j}^{\ell+1}, \phi_{i,j+1}^{\ell+1}, \phi_{i+1,j+1}^{\ell+1})$ 

  end
end

```

**Algorithm 4:** Solution decoding : from the multiresolution (wavelet) representation into the physical basis (FVS).

**Tree structure.** In order to preserve the linear complexity of Algorithms 3 and 4 it is crucial to ensure that the adaptive grid corresponds to the hierarchy of details organized in a tree structure. To make this concept more precise we introduce the notion of stencil of prediction  $SP_{i,j}^\ell$ , which is the set of mesh indices entering in the prediction of the solution values on the subdivisions of mesh  $\Omega_{i,j}^\ell$ . For the quadratic prediction that we use

$$SP_{i,j}^\ell = \{(i+k, j+m), k = -1, \dots, 1, m = -1, \dots, 1\}. \quad (6.13)$$

After identifying the meshes that cannot be coarsened using rule 6.12 we must also include at this level all the meshes belonging to their prediction stencils (6.13) even if those satisfy rule 6.12. To do so, it is practical to introduce the notion of "border cells" to denote cells that are added at a given level only because they belong to the stencil of prediction of "inner" cells. It is important to distinguish these border cells from the others because they should not be tested for subdivision at the finer level.

**Finite volume scheme on an hybrid grid.** The key point in adaptive finite volume schemes using multiresolution is the following : if the solution is smooth enough (in other words, if the details are under a threshold value) we can compute the scheme on the coarser level.

In fact, instead of computing the details, we compute regularity sensors, according to the article of Bihari and Harten [10]. The linear relation between details and regularity sensors is given by the following formula

## Chapitre 6. Design and performance analysis of a multiscale scheme for controlled population dynamics

---

$$\begin{cases} e_3 = \frac{1}{4}(d_{2i,2j}^{\ell+1} - d_{2i+1,2j}^{\ell+1} - d_{2i,2j+1}^{\ell+1} + d_{2i+1,2j+1}^{\ell+1}) \\ e_0 = \frac{1}{4}(-d_{2i,2j}^{\ell+1} + d_{2i+1,2j}^{\ell+1} - d_{2i,2j+1}^{\ell+1} + d_{2i+1,2j+1}^{\ell+1}) \\ e_1 = \frac{1}{4}(-d_{2i,2j}^{\ell+1} - d_{2i+1,2j}^{\ell+1} + d_{2i,2j+1}^{\ell+1} + d_{2i+1,2j+1}^{\ell+1}) \\ e_2 = -e_1 - e_2 - e_3 \end{cases}$$

These formulas lead to different threshold tests, that we call the Harten rules (see Algorithm 5 and 6).

<p><b>Data:</b> Regularity sensors <math>e_0, e_1, e_2, e_3</math>, threshold value <math>\epsilon_l</math></p> <p><b>Result:</b> Boolean test1</p> <p>test1 = 0 ;</p> <p>If <math>( e_3  &gt; \frac{1}{3}\epsilon_l)</math> and <math>( e_1  &gt; \epsilon_l)</math> and <math>( e_0  &gt; \epsilon_l)</math></p> <p>test1 = 1 ;</p>
---------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------

**Algorithm 5:** First Harten rule - Accounts for possible displacement of the solution.

<p><b>Data:</b> Regularity sensors <math>e_0, e_1, e_2, e_3</math>, threshold value <math>\epsilon_l</math></p> <p><b>Result:</b> Boolean test2</p> <p>test2 = 0 ;</p> <p>If <math>( e_3  &gt; 32\epsilon_l)</math> and <math>( e_1  &gt; 16\epsilon_l)</math> and <math>( e_0  &gt; 16\epsilon_l)</math></p> <p>test2 = 1 ;</p>
----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------

**Algorithm 6:** Second Harten rule - Accounts for possible formation of shocks.

We will show on a simple example, without any source term and involving only two multiresolution levels, how to update a solution defined on a multiresolution grid. If two neighboring grid cells do not have the same size (see Figure 6.4), we cannot apply directly the classical finite volume scheme (6.6), which is defined for uniform grid at level  $\ell$ . First, we add ghost cells (the coarse cell in the center is divided), in order to compute the fluxes entering the cells at the finer level (fine grid cells on the right on Figure 6.4). Then, we delete these ghost cells, and compute the fluxes entering the cell on the coarser level, at the center. We can do that using the fluxes of the finer level, by computing the sum of the fine fluxes. Indeed, using additivity theorem, the flux across a reference vertical edge  $[0, \Delta x]$  during a reference time step  $[0, \Delta t]$  satisfies

$$\begin{cases} \frac{1}{\Delta t} \frac{1}{\Delta x^\ell} \iint_{[0, \Delta t] \times [0, \Delta x]} g\phi dy dt = \frac{1}{\Delta t} \frac{1}{\Delta x^\ell} \iint_{[0, \Delta t] \times [0, \frac{\Delta x}{2}]} g\phi dy dt \\ \quad + \frac{1}{\Delta t} \frac{1}{\Delta x^\ell} \iint_{[0, \Delta t] \times [\frac{\Delta x}{2}, \Delta x]} g\phi dy dt, \end{cases}$$

which leads at the discrete level to

$$G_{i,j}^{n,\ell} = \frac{1}{2}(G_{2i,2j}^{n,\ell+1} + G_{2i,2j+1}^{n,\ell+1}),$$

and a similar formula for fluxes across horizontal edges. At the end, we compute the fluxes at the coarse level.

## Chapitre 6. Design and performance analysis of a multiscale scheme for controlled population dynamics

---

Let us now give the details of the computations of the center grid cell. We may write in this example

$$\begin{cases} \phi_{i,j}^{n+1,0} = \\ \phi_{i,j}^{n,0} - \frac{\Delta t^n}{(\Delta x^0)^2} (\Delta x^0 G_{i,j}^{n,0} - \frac{\Delta x^0}{2} G_{i+1,2j}^{n,1} - \frac{\Delta x^0}{2} G_{i+1,2j+1,0}^{n,1} + \Delta x^0 H_{i,j}^{n,0} - \Delta x^0 H_{i,j}^{n,0}) \end{cases}$$

We note that it is useful then to use the following formula

$$\phi_{i,j}^{n+1,\ell} = \phi_{i,j}^{n,\ell} + \lambda^\ell D_{i,j}^{n,\ell} \quad (6.14)$$

with  $\lambda^\ell = \frac{\Delta t^n}{(\Delta x^\ell)^2}$  and  $D_{i,j}^{n,\ell}$  the numerical divergence and source term of cell  $\Omega_{i,j}^{n,\ell}$ . This term may be computed as follows. At the beginning of each time step, we have  $D_{i,j}^{n,\ell} = 0$ . Using a loop on the hybrid grid cells, we compute for each of them the numerical fluxes (see 6.7) on the left vertical edge  $G_{i,j}^{n,\ell}$  and on the bottom horizontal edge  $H_{i,j}^{n,\ell}$ , then we modify these fluxes as follows

$$\begin{cases} \tilde{G}_{i,j}^{n,\ell} = \Delta x^\ell G_{i,j}^{n,\ell}, \\ \tilde{H}_{i,j}^{n,\ell} = \Delta x^\ell H_{i,j}^{n,\ell}, \end{cases} \quad (6.15)$$

and we update the values of the load balances of the neighbors

$$\begin{cases} D_{i-1,j}^{n,\ell} - = \tilde{G}_{i,j}^{n,\ell}, \\ D_{i,j}^{n,\ell} + = \tilde{G}_{i,j}^{n,\ell}, \\ D_{i,j-1}^{n,\ell} - = \tilde{H}_{i,j}^{n,\ell}, \\ D_{i,j}^{n,\ell} + = \tilde{H}_{i,j}^{n,\ell}. \end{cases} \quad (6.16)$$

At the end we add the source term

$$D_{i,j}^{n,\ell} + = \Delta t^n \Lambda_{i,j}^{n,\ell} \phi_{i,j}^{n,\ell}.$$

**Data:** Solution on hybrid grid at time  $t^n$   
**Result:** Flux balance  $D_{i,j}^{n,\ell}$  at time  $t^n$   
Initialize load balances  $D_{i,j}^{n,\ell} = 0$ ; **for**  $\ell = (L - 1)$  **to**  $1$  **do**  
    **for** cells in level  $\ell$  **do**  
        | If (not ghost)  
        | Compute fluxes (6.15)  
        | Update load balances (6.16)  
    **end**  
    **for** cells in level  $\ell$  **do**  
        | If (ghost)  
        | Delete three ghost cells with Projection (6.9)  
    **end**  
**end**  
**for** cells in level  $0$  **do**  
    | Compute fluxes  
**end**

**Algorithm 7:** Flux computation

## Chapitre 6. Design and performance analysis of a multiscale scheme for controlled population dynamics

---

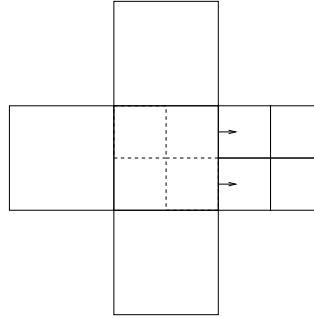


FIGURE 6.4 – *Finite volume scheme on a hybrid grid : the flux computation between two grid cells of different size (for example the center right fine cells, marked with an horizontal arrow, and the center coarse cell) have to be done in two steps. First, we have to add ghost cells in order to compute the fluxes on the fine level, then we delete these cells doing a projection (6.9), and compute the fluxes on the coarse level.*

**Update the hybrid grid.** The hybrid grid at time  $t^n$  is generated by a graded tree, noted  $T^n$ , defined by recursion

$$\forall n \geq 1, \quad T^{n+1} = P(T^n)$$

with  $P$  the prediction operator. This operation, consisting in evaluating the grid cells that have to be either added or kept, is done in three steps :

- Adding of cells using the heuristic Harten rules (Algorithms 5 and 6) ;
- Adding of ghost cells : in order to be able to compute the fluxes on the hybrid mesh, we add all the necessary ghost cells (see Figure 6.4) ;
- Deletion of unused cells ;

Thus, updating the hybrid grid consists in :

1. Encoding the solution (see Algorithm 8) ;
2. Computing the prediction operator ;
3. Decoding the solution (see Algorithm 4) ;

## Chapitre 6. Design and performance analysis of a multiscale scheme for controlled population dynamics

---

In Algorithm 8 we gather these operations.

```

Data: Solution encoded on hybrid tree at time  $t^n$ 
Result: Solution encoded on hybrid tree at time  $t^{n+1}$ 
for  $\ell = (L - 1)$  to  $1$  do
  | for cells in level  $\ell$  do
  | | Add cells by Harten rules (Algorithm 5) and (Algorithm 6).
  | end
end
for  $\ell = (L - 1)$  to  $1$  do
  | for cells in level  $\ell$  do
  | | Add ghost cells (see Figure 6.4).
  | end
end
for  $\ell = (L - 1)$  to  $1$  do
  | for cells in level  $\ell$  do
  | | Delete unused cells.
  | end
end

```

**Algorithm 8:** Grid adaptation

**Initialize the hybrid grid.** The initial hybrid grid  $T^0$  may be defined by using two different methods.

The first method consists in computing the grid from the finer level. One defines the solution on all the fine grids, and then apply the algorithmic sequence : coding, thresholding, decoding. In other words, we compute all the details, and delete all the grid cells where the details are below the threshold. The problem with this method is that it uses a lot of memory at the beginning.

Another approach may be used, without defining the entire fine level. First, we compute the solution on the coarser level. Then we do a two level recursion. At level  $\ell$  we compute the predictions and the details at level  $\ell + 1$  (we can do it because we know the exact solution at time  $t = 0$ ). We split all the grid cells where the details are significant. Yet, as we can see on Figure 6.5, it is not enough. The grid mesh has to be graded. In order to ensure this property, we do another loop on the level, and split all the grid cells that are in the prediction stencil defined by (6.13) of the new grid cells created just before. This can be a recursive operation if the prediction stencils of these new cells are not already in the grid.

We have illustrated the differences between both methods on Figure 6.6. The parameters used are :  $\epsilon = 0.01$ , 10 grid cells on the coarsest level, 5 levels of multiresolution. Initialization starting from the coarsest grid is of course much more economical in terms of memory. However, this method should be used with caution when the initial condition is not smooth enough. Indeed the coarsest discretization must be able to capture all the features of the initial condition in order to trigger its refinement to the adequate level of discretization. In our case, we use piecewise

## Chapitre 6. Design and performance analysis of a multiscale scheme for controlled population dynamics

---

constant or slowly varying initial condition and we safely adopt this strategy.

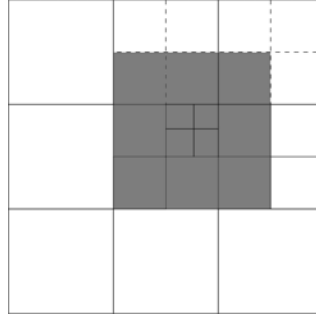


FIGURE 6.5 – Initialization of the hybrid grid from the coarsest level. In this special case, we see that the prediction stencil of the small grid cell in the center (grey part of the domain) is not full (there are cut lines at the places where grid cells are needed). In order to get a graded mesh tree when initializing from the coarsest level, it is necessary to do two loops on each level. The first one identifies the grid cells that have to be created because the details are significant, and the second one is needed to grade the hybrid tree.

```

Data: Initial condition  $\phi_0$ 
Result: Solution on hybrid grid at time  $t^0$ 
Create and initialize cells of level 0 with initial condition.
for  $\ell = 0$  to  $L - 1$  do
    for inner cells in level  $\ell$  do
        | Subdividing according to local smoothness (Algorithm 10)
    end
    for inner cells in level  $\ell + 1$  do
        | Complete stencil of prediction (Algorithm 11)
    end
end

```

**Algorithm 9:** Initialization from the coarsest level and graduation of a hybrid grid.

```

Data: Mother cell, level  $\ell$ , threshold value  $\epsilon_\ell$ 
Result: Additional cells if refinement is necessary
Compute initial condition values on the 4 daughter cells
Compute predicted values on the 4 daughter cells
Compute 4 daughter cells details  $d_1, d_2, d_3, d_4$ 
If  $(|d_1| > \epsilon_\ell)$  or  $(|d_2| > \epsilon_\ell)$  or  $(|d_3| > \epsilon_\ell)$  or  $(|d_4| > \epsilon_\ell)$ 
    Replace mother cell by its 4 daughters
    Initialize 4 daughter cells' values with initial condition

```

**Algorithm 10:** Subdividing according to local smoothness : Compare initial condition values with predicted values computed on a coarser level, for each level.

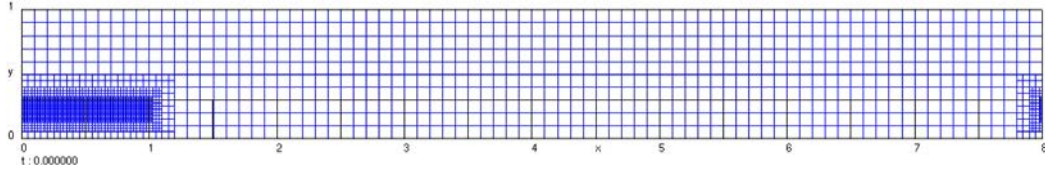


FIGURE 6.6 – Initialization method from the coarsest level. We clearly see that this method use less memory than in the case where we initialize from the finest grid.

**Data:** Inner cell, level  $\ell + 1$   
**Result:** Additional cells  
 For cells in stencil prediction of mother cell  
 If(level <  $\ell$ )  
 Divide to create border cells (Algorithm 12)

**Algorithm 11:** Complete stencil of prediction : if needed, subdivision of the mother cells of the neighbors of the new divided cells, in order to complete their prediction stencils.

**Data:** Border cell, level  $\ell$   
**Result:** Additional cells  
 Create cells of stencil of prediction if needed (**possible recursivity**)  
 Initialize 4 daughters cells values

**Algorithm 12:** Divide to create border cell

**Precision enhancement.** In order to enhance the precision of the scheme, we can improve the numerical scheme. On one hand, the scheme has to be precise in space. It is ensured thanks to a good choice of the numerical flux (6.7).

For more details about the original scheme developed in the framework of the model of the selection process of ovarian follicles, with specific transmission conditions (in particular concerning mitosis), reader can refer to [7] and [8]. In the first reference, one may find details about the numerical scheme on uniform grids, and in the second, one may find details on how to adapt the previous scheme in a framework of adaptive meshing using multiresolution analysis.

On the other hand, we have to get better precision in time, which can be obtained with a Runge-Kutta scheme. Yet, we have to adapt this method if we want to use it on an adaptive grid. Runge-Kutta schemes consist in computing intermediate steps to compute each time step. Since the Finite Volume scheme designed in [7] achieves at best third order precision in space, we need to use a third order scheme in time : RK3 (see [38]). For the first step, we compute

$$\phi_{i,j}^{*,\ell} = \phi_{i,j}^{n,\ell} + \lambda^\ell D_{i,j}^{n,\ell}, \quad (6.17)$$

for the second step,

$$\phi_{i,j}^{**,\ell} = \phi_{i,j}^{*,\ell} + \lambda^\ell \left( \frac{1}{4} D_{i,j}^{n,\ell} + \frac{1}{4} D_{i,j}^{*,\ell} \right), \quad (6.18)$$

## Chapitre 6. Design and performance analysis of a multiscale scheme for controlled population dynamics

---

and for the last step,

$$\phi_{i,j}^{n+1,\ell} = \phi_{i,j}^{n,\ell} + \lambda^\ell \left( \frac{1}{6} D_{i,j}^{n,\ell} + \frac{1}{6} D_{i,j}^{*,\ell} + \frac{2}{3} D_{i,j}^{**,\ell} \right). \quad (6.19)$$

with  $D_{i,j}^{*,\ell}$  the flux balance (including the source term) computed from the intermediate solution  $\phi_{i,j}^{*,\ell}$ , and  $D_{i,j}^{**,\ell}$  the flux balance (including the source term) computed from the intermediate solution  $\phi_{i,j}^{**,\ell}$ .

At each intermediate step, instead of deleting ghost cells after the flux computation, we keep them in the hybrid tree. Then after updating the solution, we compute the new values of the ghost cells, and the hybrid grid is ready for a new intermediate step. We free the memory corresponding to the ghost cells only at the final intermediate step, when the cells are no longer necessary.

**Data:** Solution coded on hybrid tree  
**Result:** Solution on hybrid grid

```

for  $\ell = L - 1$  to 1 do
  for cells in level  $\ell$  do
    If(ghost cell)
    Reconstruct three daughter cell values with Prediction operator (6.10)
    and detail formula (6.11)
       $(\phi_{i,j}^\ell, d_{i+1,j}^{\ell+1}, d_{i,j+1}^{\ell+1}, d_{i+1,j+1}^{\ell+1}) \rightarrow (\phi_{i,j}^\ell, \phi_{i+1,j}^{\ell+1}, \phi_{i,j+1}^{\ell+1}, \phi_{i+1,j+1}^{\ell+1})$ 
    end
  for cells in level  $\ell$  do
    If(ghost cell)
    Reconstruct mother cell value with projection formula (6.9)
       $(\phi_{i,j}^\ell, \phi_{i+1,j}^{\ell+1}, \phi_{i,j+1}^{\ell+1}, \phi_{i+1,j+1}^{\ell+1}) \rightarrow (\phi_{i,j}^{\ell+1}, \phi_{i+1,j}^{\ell+1}, \phi_{i,j+1}^{\ell+1}, \phi_{i+1,j+1}^{\ell+1})$ 
    end
  end
end

```

**Algorithm 13:** ghostReconstruction

### 6.2.2 Mesoscopic and macroscopic scale : Numerical moments and control.

Once we have defined a discretization of time and space, and after having computed the densities at the microscopic level, we can compute the numerical first order moments at the mesoscopic scale

$$m_{1,f}^n = \sum_{(i,j,\ell) \in T} \phi_{i,j}^{n,f} (\Delta x^\ell)^2 y_{i,j,\ell}, \quad (6.20)$$



## Chapitre 6. Design and performance analysis of a multiscale scheme for controlled population dynamics

---

the numerical global first order moment, where  $y_{i,j}^\ell$  is the ordinate of the center of the corresponding grid cell,

$$M_1^n = \sum_{i=1}^{N_f} m_{1,f}^n, \quad (6.21)$$

the numerical global control,

$$U^n = S(M_1^n), \quad (6.22)$$

and the numerical controls at the mesoscopic scale,

$$u_f^n = b(m_{1,f}^n, U^n). \quad (6.23)$$

### 6.2.3 Parallel computing

In order to speedup the code, we use parallel computing techniques. Let us first recall some basics about parallel computing (the reader can find an introduction to parallel computing in the lecture [68]).

Schematically, a computer has three main components : the CPU (Control Process Unit), the disk cache memory and the global memory. The disk cache is a small memory with fast data access, as opposed to the main memory, that is larger, but with slow data access. In parallel computing, computations are done on a computer with several processors, each of them endowed with a disk cache and a global memory. Tasks that can be performed independently and simultaneously will be splitted between the different processors. In the framework of distributed memory computation, which is the set-up that we have selected, there is no global memory, but small local memories, associated to each CPU. If the processes need to communicate, this communication is done via a network. There are several well-known problems that may arise in parallel computing, such as those described below.

- If the amount of computation to be done is not the same for each process then naturally some of the processes are forced to wait for the others. (unbalanced load). The load needs to be distributed as equally as possible among the processes.
- As opposed to classical computing, communications between processes are needed to share data. Therefore, a certain amount of work is added. When designing a parallel code, we should take care of not spending too much time on communications. Indeed, if there are too many communications, the parallel method may take more time than the sequential one.
- Since the disk cache is a fast data access memory, in the best scenario, the whole data is in the disk cache. If this is not the case, the process has to access the small local memory and this action penalizes the user time.

We will now propose a parallel scheme for our multiscale model. Trying to solve our specific problem in this parallel framework will lead us to choose between two options.

**Option 1 : domain decomposition method.** The first option involves using a classical domain decomposition method (see Figure 6.8). The idea is to split the

## Chapitre 6. Design and performance analysis of a multiscale scheme for controlled population dynamics

---

computational domain into several parts, and send each of them on a process. To do so, we have to add several communications, at each time step, as follows :

- FVS evolution : Each process has to send the table of the boundary cells whose mean value enters the flux computation on the boundary of the grid mesh of its neighbors. It also has to receive from the other processes the corresponding tables on the other part of the boundary. The bottom line is that each process has to send two tables and receive two tables (except for those on the outside boundaries, which send and receive only one table), of the size of the mesh. This communication is mesh size dependent.
- Moments computation : each process has to compute its own part of the first order moment, for each sub-population, and share the table of local first order moments. This can be done with a reduction operation. Each process has to make  $N_f$  reduction operations, that will compute the  $N_f$  local first order moments.
- When all the involved processes are aware of all local maturities, the global first order moment is computed as the sum of the local maturities (reduction operation (6.21)).
- When the global and local control values are computed, the local time steps can also be computed. Afterward, the global time step can be computed as the minimum of the local time steps (reduction operation)

$$\Delta t = \min_{f \in \{1, \dots, N_f\}} \Delta t_f \quad (6.24)$$

**Option 2 : biology inspired method.** A second option is to try to mimic biology. As we know, one of the problems arising in parallel computing is to deal correctly with the available data, as well as to keep a maximal granularity, avoid communications and try to localize data in space and time in order to optimize disk cache accesses. The idea behind this option is to send each sub-population on a process (see Figure 6.7). In some way, it can be viewed as an efficient domain decomposition method. Indeed, if we replicate the computational domain (one for each sub-population), and represent them as a stack (see Figure 6.8 for an example with 2 sub-populations), this method can be considered as a domain decomposition method, which avoids a large number of communications. The only communications to be done are the following :

- Computation of the global first order moment as the sum of the local maturities (reduction operation).
- Computation of the global time as the minimum of the local time steps (reduction operation).

Moreover, these communications are the fastest as possible, as they use only reduction operations, which are moreover mesh size-independant.

**Conclusion.** The second option was chosen, for the following reasons :

- First, as we have seen, it involves fewer and more efficient communications.
- Second, the load balance is better with this option. Indeed, the densities are quite localized in the computational domain, and they start from the same area in their

## Chapitre 6. Design and performance analysis of a multiscale scheme for controlled population dynamics

---

respective domains at initial time. If we look at Figure 6.8, with the first option, process 1 will be very busy at the beginning, because it contains all the active grid cells, while process 2 is not busy. Then, when the densities move to the second part of the domain, the roles are inverted. Using option 2 ensures that the load balance remains good with time.

We now summarize the methods described in the previous sections and define the general SIMD (Single Instruction Multiple Data) algorithm for our problem.

```
Initialize hybrid grid from coarse level (Algorithm 9).
while  $t < T$  do
  Update hybrid grid :
  - Coding (Algorithm 3)
  - Prediction (Algorithm 8)
  - Decoding (Algorithm 4)
  Numerical moments :
  - Compute local moment (6.20)
  - Communication(reduction operation - sum) : global moment (6.21)
  Numerical controls :
  - Compute global control (6.22)
  - Compute local control (6.23)
  Stable time step
  - Compute local time step (6.8)
  - Communication (reduction operation - min) : global time step. (6.24)
  Evolution
  - Compute fluxes (Algorithm 7)
  - Update values : first step of RK3 (6.17)
  - Reconstruct ghost cells (Algorithm 13)
  - Compute fluxes (Algorithm 7)
  - Update values : second step of RK3 (6.18)
  - Reconstruct ghost cells (Algorithm 13)
  - Compute fluxes (Algorithm 7)
  - Update values : last step of RK3 (6.19)
end
```

**Algorithm 14:** SIMD algorithm

### 6.3 Performance analysis

In this section, we will present a new method dedicated to the performance analysis of a parallel numerical scheme, based on uncertainty quantification. We apply this method to the example of follicular development introduced in the first section. For an introduction to the bases of probability and uncertainty quantification, the interested reader is referred to [52] and [74].

Chapitre 6. Design and performance analysis of a multiscale scheme for controlled population dynamics

---

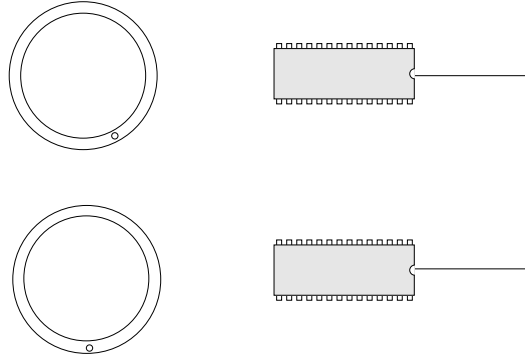


FIGURE 6.7 – Strategy of parallel computing using distributed memory. The strategy mimics biology : at each sub-population we associate a process. This strategy is easy to implement (SIMD algorithm), and involves few and efficient communications (reduction operations).

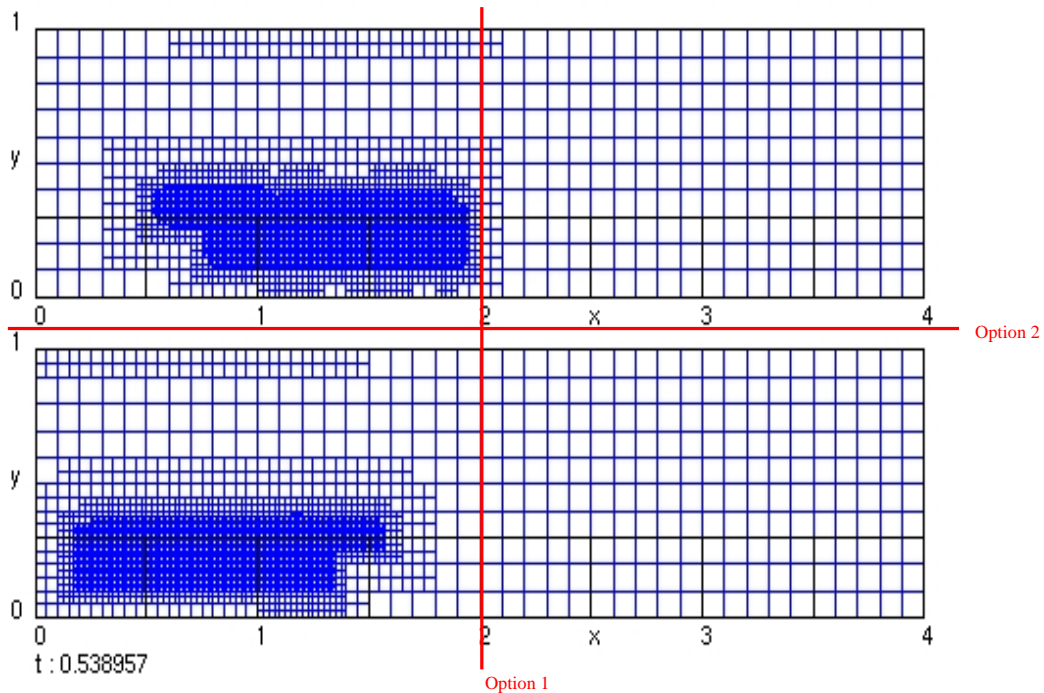


FIGURE 6.8 – Parallel computing strategy. There are 2 options for data management. Option 1 : domain decomposition method. We cut the domain into several parts and send them to different processes. Option 2 : split the sub-populations and send them to different processes. There are less communications using option 2, and the communications that have to be done are efficient (reduction operations) and mesh size independent.

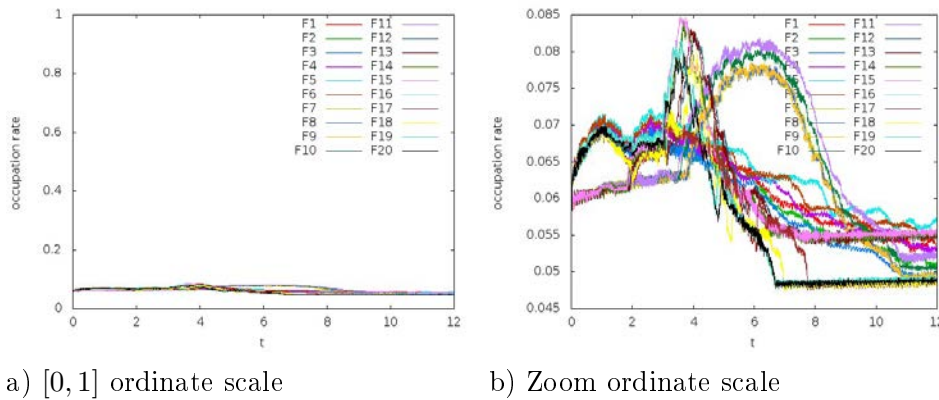


FIGURE 6.9 – Load balance in a simulation of a selection process among 20 sub-populations. In abscissa, the time, in ordinate, the occupation rate (6.26) of each sub-population (the amount of computations done by the process supporting the sub-population is proportional to this quantity). In this case, the computational domain is long ( $N_c = 8$ ), and we set different speed parameters for each sub-population. In panel a) we display the occupation rate in the scale  $[0,1]$ , to highlight the overall very good performance of the multiresolution compared with the uniform finest mesh. In panel b) we zoom on the range of interest to emphasize the differences among follicles.

### 6.3.1 Method

**The importance of using a stochastic method.** There exist several classical methods for evaluating the performance of a parallel code. In problems such as ours, where the size of the problem grows with the number of processes, the classical test is a weak scalability test.

The theoretical set-up for the evaluation of the weak scalability would consist in increasing the number of processors, and letting the parameters unchanged. This method was successfully applied in [6], in the framework of uniform meshes.

Since we have different parameters on each processor, we face two problems.

Due to the stability condition associated with the explicit numerical scheme, the parameters defining the control and velocities determine the local time step, hence the number of time steps necessary to reach a fixed final time. In our problem, the processors have to be synchronized, and to use the same global time step, computed as the minimum of the times steps computed on the mesoscopic scale. If the increase in the number of processes corresponds to a change in the time step, then the notion of weak scalability is modified because the size of the problem does not depend only on the number of processes.

Another source of variability between processors is the time adaptive meshing. Since the solutions are distinct on different processors, the meshes are not the same, so that the number of grid cells are also different.

In practice, the user who modifies or creates his own set of parameters is not ensured

## Chapitre 6. Design and performance analysis of a multiscale scheme for controlled population dynamics

---

to get the same performances as in the reference situation. In order to get the real performances, the user wants to know the average performance of the code. For example, the average user time may be of interest.

In this regard, our idea was to use a Monte Carlo method. We introduce perturbations on the parameters, and evaluate the resulting average performances. We can then define a mean weak scalability method, generalizing the weak scalability method, by plotting the mean user time and its variance with respect to the number of processes.

**Principle of the Monte Carlo method.** Based on the law of large numbers, the Monte Carlo method is a numerical method that evaluates the mean value of a random distribution of real numbers. The principle is to simulate many realizations (say  $N$  realizations) of a random variable  $X$ , and then compute the empirical average. The law of large numbers ensures that

$$\lim_{N \rightarrow \infty} \frac{1}{N} \sum_{i=1}^N X_i = \langle X \rangle, \quad (6.25)$$

where  $X_i$  is the  $i$ -th realization of  $X$ , and where by  $\langle \cdot \rangle$  we denote the average function.  $N$  will be set later for each numerical test. It is possible to visualize the results by using histograms, to represent the tabulated frequency of the data over discrete intervals (bins), with an area proportional to the frequency of the observations in the interval.

In practice, we simulated uniform laws using the Mersenn twister algorithm [59]. The seeds used for Mersenn twister method were

$$\text{time(NULL)} \times (p + 1)$$

with  $p$  the number of process (it starts from 0), in order to get different sequences on each process.

### 6.3.2 Mean load balance test

**Occupation rate.** As reminded previously, the key point in parallel computing is the data management. One of the classical concepts used is the load balance, which in our case is directly linked to the size of the FVS grid and to the time step. Each process has a certain amount of work to do. We say that the load balance is good when the work is well splitted between the processes, and none of them is working more than the others, constraining them to wait.

The occupation rate, denoted  $R(t)$ , is defined by

$$R(t) = \frac{m(t)}{M}, \quad (6.26)$$

where  $m(t)$  is the number of hybrid grid cells at time  $t$  and  $M$  is the total number of grid cells on the finest uniform grid. We note that  $4^{1-L} < R \leq 1$ . Indeed, there is

## Chapitre 6. Design and performance analysis of a multiscale scheme for controlled population dynamics

---

always at least the coarsest grid cells if no grid cell is refined, and in the worst case the whole finest grid is active.

The number of arithmetic operations in a simulation, in our case, is proportional to the number of active grid cells. Then, we can say that the occupation rate is a good indicator of the amount of computation, and we can use it in order to evaluate the load balance of the code.

On Figure 6.9, we have plotted the occupation rate curves in the case of a selection process among 20 sub-populations, which differ by their speed parameter. In this case,  $N_c = 8$ , the domain of computation is very large compared to the size of the support of the solution, which amounts to a very good overall occupation rate. At most 8.5% of the size of the uniform finest grid is required at any time of the simulation. We note that at the beginning, the occupation rates are the same, because all the follicles start from the same initial condition, and then, due to the differences between the velocity parameters, they become different from one another.

In the right panel, we see the influence of the heterogeneity among the follicles on the occupation rate which varies from 4.8% to 8.5%. Moreover its time dependance is also very different from one follicle to the next. Some curves exhibit a clear maximum around  $t = 4$ , others around  $t = 6$ , and others have a more regular behaviour throughout the simulation. Indeed we have noticed that the support of the density may become larger during the time span where some cells exit the proliferation cycle, if this event happens over several cycles. This behaviour is of course completely dependant on the individual velocity parameters of each follicle and its heterogeneity of the dynamics reflects on the occupation rates.

We next try to answer the question of the sensitivity of the computational performances with respect to the parameters of the simulation.

**Method.** The mean load balance test consists in performing many realizations of the numerical simulation of our model with perturbed parameters for one sub-population (sequential case). The aim is to evaluate the mean load balance. We add uniformly distributed perturbations to four distinct parameters :  $p$  and  $C_p$  appearing in the dynamics,  $\mu_1$  in the initial condition and  $\bar{M}$  in the global control term. The range of variation is the largest one ensuring that the parameters satisfy the biological specifications established in the chapter on the calibration of the model. Table 6.1 and Table 6.2 indicate the biological and numerical parameters. For the mean load balance test, we have used these parameters and we considered the perturbations of Table 6.3. Furthermore, in the Monte Carlo method (6.25), we have set  $N = 200$ .

**Hardware.** This test was done on a laptop Dell precision M4400 with an Intel Core 2 Duo processor of 2.26 G-Hz, 3Mo of disk cache.

**Results.** In order to interpret the Monte Carlo simulations we have plotted on Figure 6.11 the outputs (occupation rate and user time) obtained with 200 realisa-

**Chapitre 6. Design and performance analysis of a multiscale scheme for controlled population dynamics**

---

Parameter	Description	Value
$U_{\min}$	Global control minimum level	0.075
$U_{\max}$	minimum level	0.15
$c$	slope parameter	2.0
$\bar{M}$	abscissa of the inflection point	25
$b_1$	Local control basal level	0.08
$b_2$	exponential rate	2.2482
$b_3$	scaling factor	1450.
$\bar{\Lambda}$	Source term intensity factor	0.6
$\Lambda_w$	half width	0.05
$\bar{\gamma}$	scaling factor	0.1
$\gamma_s$	cellular maturity threshold	0.3
$C_p$	Dynamics capacity	0.47
$p$	speed	2.2
$\alpha$	contrast	1
$c_1$	Maturation function	0.032
$c_2$		0.148
$\bar{u}$		0.05

TABLE 6.1 – *Biological parameters*

tions of the same reference situation, with the parameters equal to the mean values of Table 6.4. As expected, the occupation rate is the same for all these simulations and equal to 0.1717. The user time is not constant because it can be perturbed by the computer load. It is distributed with a beta- like distribution around the mean value 27.27.

The results of the random simulations are exposed in Figure 6.10. There are four rows of figures, each of them corresponding to a set of Monte Carlo simulations where we let one parameter vary randomly with a uniform distribution in the range specified in Table 6.3, while the three other parameters remain fixed and equal to their mean value (in the left column of Table 6.4). On each row, we represent the histogram of a parameter in input (left column), the histogram of the occupation rate in output (center column) and the histogram on the user time in output (right column).

The first two lines correspond to variations of the parameters appearing in the velocity functions ( $p$  and  $C_p$ ). The distribution of the occupation rate looks uniform in the  $p$  case and more skewed towards the lower values in the  $C_p$  case. In both cases the mean value of the occupation rate (the number in parenthesis in Table 6.4) is very close to the occupation rate for the reference case. The distribution of the user



## Chapitre 6. Design and performance analysis of a multiscale scheme for controlled population dynamics

---

$N_c$	1
$N_m$	5
$L$	5
$\epsilon$	0.01
$\lambda$	0.4
$T$	12

TABLE 6.2 – *Numerical parameters.*

$p$	$U([0.75, 1.5])$
$C_p$	$U([0.56, 0.63])$
$\mu_1$	$U([5.7, 7.7])$
$\bar{M}$	$U([1.5, 3.5])$

TABLE 6.3 – *Laws of probability used to generate the random parameter perturbations for the simulations displayed in Figure 6.10. The ranges of variation are chosen in order to respect the biological constraints.*

time is clearly non uniform and has a beta-like shape in both the  $p$  and  $C_p$  cases. In the  $p$  case the mean value ( $\approx 30$ ) is slightly larger than the mean user time in the reference case ( $\approx 27$ ).

On Table 6.4 we have gathered the median values and the maximum deviation in percentage for the results displayed in Figure 6.10, computed with the following formula

$$\text{median value} = \frac{X_{\max} + X_{\min}}{2}$$

and

$$\text{max deviation in percentage} = \frac{X_{\max} - X_{\min}}{X_{\max} + X_{\min}}$$

where  $X_{\min}$  and  $X_{\max}$  are the extremal values reached by the random input parameter or by the output measurement. Because of the constraints on the parameters imposed by the biological specifications the random variations have different ranges for the four selected parameters. It is therefore difficult to appreciate their relative sensitivity. However the comparison of the maximum deviations in percentage between the occupation rate and the input parameters seems to indicate that variations in  $\mu_1$  and  $\bar{M}$  have little influence and that  $C_p$  is relatively more influential than  $p$ .

As opposed to Figure 6.10, which treats separately the impact of perturbing each of the four parameters, Figure 6.12 depicts the global situation where all four parameters are perturbed at the same time. With the perturbations of Table 6.3 on parameters  $p$ ,  $C_p$ ,  $\mu_1$  and  $\bar{M}$  in input, the occupation rate varies around a median value of 0.17 with a maximal deviation of  $\pm 18\%$  of the median value. There has obviously been a compensation phenomenon. The user time on the other hand varies between  $\approx 30$  and  $\approx 45$ . This interval is altogether above the average user time for

## Chapitre 6. Design and performance analysis of a multiscale scheme for controlled population dynamics

---

Input : perturbed parameter	Output : Occupation rate	Output : User time
$p = 1.125 \pm 33\%$	$0.175 \pm 14\%$ (0.1764)	$37.5 \pm 33\%$ (30.07)
$C_p = 0.595 \pm 5.9\%$	$0.1725 \pm 4\%$ (0.1720)	$29 \pm 10\%$ (26.83)
$\mu_1 = 6.7 \pm 15\%$	$0.171 \pm 1.2\%$ (0.1714)	$32 \pm 25\%$ (27.80)
$\bar{M} = 2.5 \pm 40\%$	$0.17 \pm 1.4\%$ (0.1704)	$31.5 \pm 17\%$ (27.72)

TABLE 6.4 – Results of the mean load balance test. Median values, maximum deviation in percentage and mean value of the output (in parentheses). Left column : perturbation on 4 different input parameters. Center column : occupation rate at final time. Right column : User time

the reference situation (see Figure 6.11), which indicates that variations on several parameters simultaneously combine into a systematic slowdown of the simulation. This test on one processor can be used to predict some features of the parallel case behavior. We expect that the load balance will remain good. However, it is not sufficient to ensure good weak scalability results. Indeed, even if the size of the meshes associated with the different populations are well balanced in size, the variability in the velocities can induce a variability in the time step. Since the population has to evolve in a synchronized manner, the smallest time step is used. If we progressively add sub-populations to our system, with randomly chosen velocity parameters, the time step can only decrease and therefore penalizes the overall CPU time.

These numerical simulations give us some statistic informations about the numerical performances of the scheme that we have designed, and furthermore have set a methodology, allowing us to run a systematic study of parameters' sensibility. In a close future, we aim at extend this method of uncertainty propagations to other outputs, such as biological outputs, describing the qualitative behavior of the model (ovulation rate, exit time, final mass, etc...).

### 6.3.3 Mean weak scalability test

**Method.** As explained in paragraph 6.3.1 in the class of models that we study the size of the problem is related not only to the number of processes but also to the heterogeneity in the parameters. We generalize the notion of weak scalability to define a "mean weak scalability". This test consists in performing many realizations of numerical simulations of the multiscale model for different sizes of the problem. We have tested four situations : the case of a compact domain (useful for visualizing the results on the mesoscopic and macroscopic scales, with fast computations), unrolled domain (useful for visualizing the results on the microscopic scale), precision enhancement on the whole domain (refine coarse level), precision enhancement locally (add one level of multiresolution). For this test we used the biological parameters listed in Table 6.1 and the numerical parameters from Table 6.2. For the perturbed parameters  $p$ ,  $C_p$ ,  $\mu_1$  and  $\bar{M}$ , we used Table 6.3. Moreover, in the Monte Carlo method (6.25), we set  $N = 20$ .

## Chapitre 6. Design and performance analysis of a multiscale scheme for controlled population dynamics

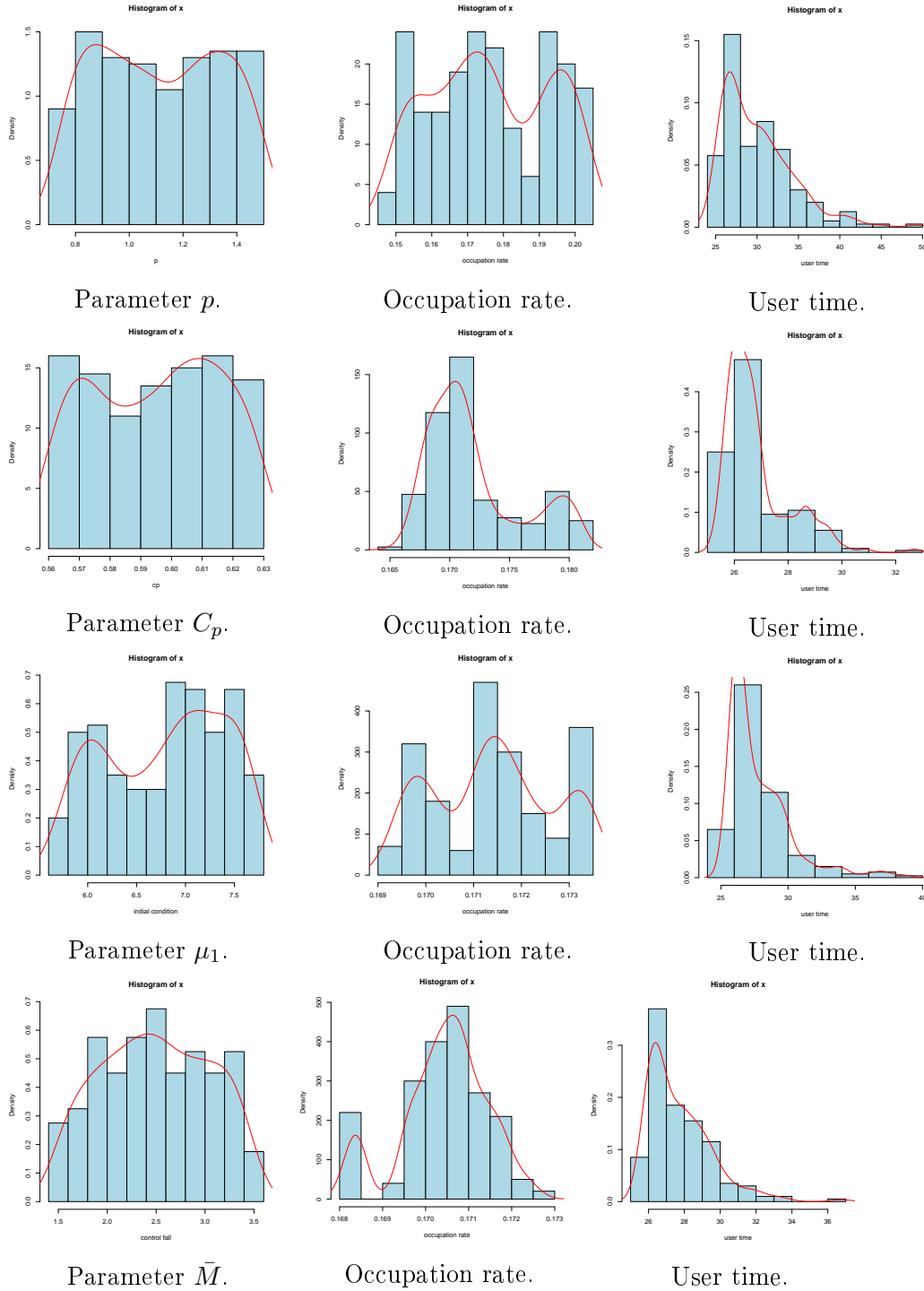


FIGURE 6.10 – Mean load balance test : perturbation of an input parameter (left column) and impact on the occupation rate at final time (center column) and user time (right column). Different parameters are tested :  $p$  and  $C_p$  appearing in the velocity functions,  $\mu_1$  appearing in the initial condition and  $\bar{M}$  appearing in the global control.

## Chapitre 6. Design and performance analysis of a multiscale scheme for controlled population dynamics

---

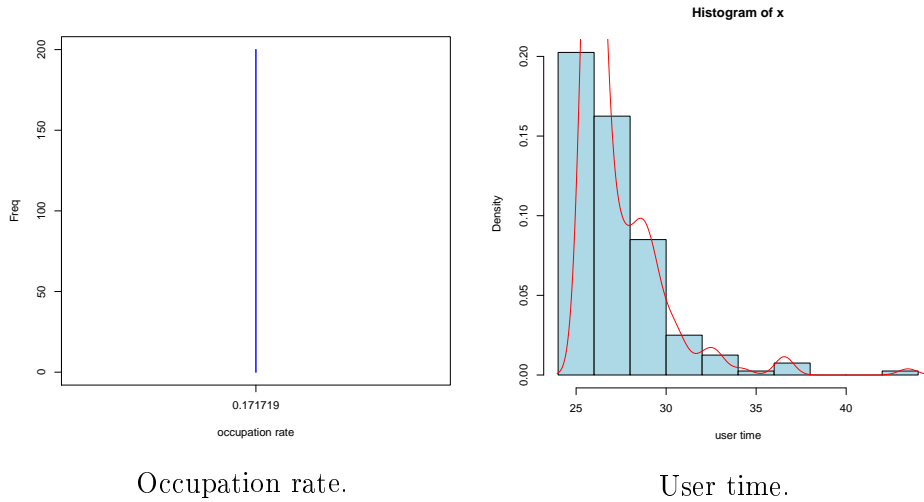


FIGURE 6.11 – Mean load balance test : Reference test without perturbations on the input parameters.

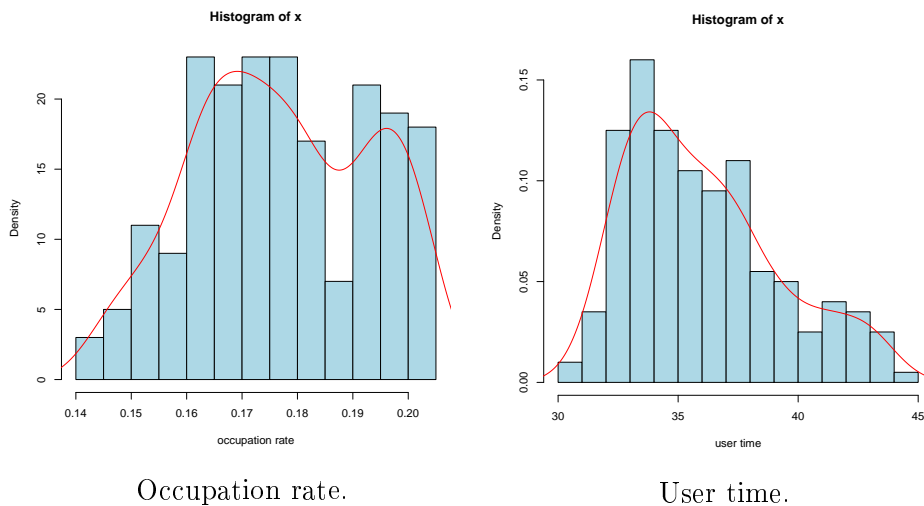


FIGURE 6.12 – Mean load balance test : perturbation of the reference situation and impact on the occupation rate and CPU time. The four parameters of Table 6.3 are perturbed at the same time.

## Chapitre 6. Design and performance analysis of a multiscale scheme for controlled population dynamics

---

**Hardware.** Cluster HPC1 of the Jacques-Louis Lions Laboratory is a SGI Altix UV 100, with 20 CPUs Intel Xeon 64 bits Nehalem EX 7550 of 8 cores of 2 G-Hz (160 cores), 640 Go of RAM, using Numalink interconnect system (15 GB/s).

**Results.** Figure 6.13 shows the results of this test for different cases. On each panel, the blue curve is the theoretical sequential time. We computed the mean user time for one process, denoted  $\langle t_{\text{seq}} \rangle$ , and we draw the curve

$$c(N_p) = \langle t_{\text{seq}} \rangle \times N_p,$$

with  $N_p$  the number of processes. The green curve is the theoretical limit. By using this type of parallelization strategy (distributed memory using MPI), the computational cost for several processes cannot be inferior to that corresponding to the behavior we could expect if the same computation was performed on all processors without any communications. It is important to note that this is not the case in a domain decomposition method, where the user time should decrease with the number of processes. In our case, the overall size of the problem grows with the number of processes. The black points correspond to realizations of numerical simulations using our method.

The parallel method that we use is more performant, in all cases, than the sequential method. Moreover, the user time is close to the theoretical limit.

On Figure 6.14, we represent zooms of the panels of Figure 6.13. The black dots are realizations of numerical simulations, and the red curve corresponds to the linear regression. On these panels, we see that there is a variability in the user time. This variability seems to decrease with the number of processes. This would be logical since the time step seems to be the major limiting factor on the CPU time. Indeed, adding processes can only force the local time step to decrease, and the processes are synchronized. Then, adding process can only slow all the processes.

In order to identify the source of this variability we first perform the mean weak scalability test on the reference situation with a uniform mesh, a constant time step, and the same parameters for all the populations and simulations. We then introduce variability through the four selected parameters  $C_p$ ,  $p$ ,  $\mu_1$  and  $\bar{M}$ , that vary simultaneously among the populations and also from one simulation to the next. We perform 20 independent simulations per number of processes. Since a change in the parameters influences the time step through the stability condition (6.8) and also the load balance through the adaptive mesh size, we study these two factors independently. For the uniform meshes, we have set  $N_m = 80$ ,  $L = 1$  and  $\epsilon = 0$ . For the constant time step we have set  $\Delta t = 0.001$ . For the other parameters we have used, when necessary, Tables 6.1, 6.2 and 6.3. In particular, the simulations with random parameters are performed with uniformly distributed random realisations of the four parameters  $C_p$ ,  $p$ ,  $\mu_1$  and  $\bar{M}$  simultaneously.

The results are displayed in Figures 6.15 and 6.16. They are composed of four panels : The two top panels correspond to simulations performed with the same discretisation parameters in time and space (uniform grid). In the left panel the parameters  $C_p$ ,

## Chapitre 6. Design and performance analysis of a multiscale scheme for controlled population dynamics

---

$p$ ,  $\mu_1$  and  $\bar{M}$  are constant for the 20 simulations (equal to the mean values in Table 6.3). In the right panel they vary randomly, but, as expected, without influencing the user time statistics. In the bottom left panel, we use the uniform mesh to see the influence of the random parameters on the time step only. In the bottom right panel the time step is kept fixed, but the random parameters influence the space discretization through adaptive mesh refinement.

As for the single processor reference test displayed in Figure 6.11 we note that even in the reference situation, there is some variability in the user time. This is related to the technology used, and it is known that the user time on a deterministic problem is subject to variability.

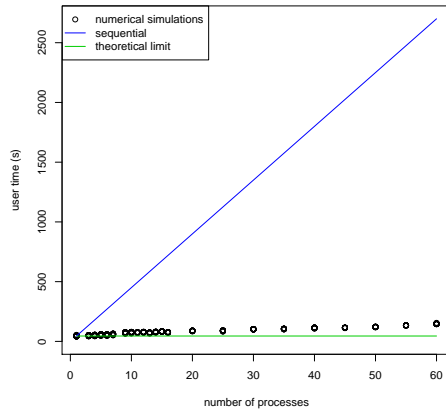
The range of variation of the user time due to the variability of the adaptive time stepping is roughly three times wider than the one due to variability of the adaptive mesh.

Also, we clearly see, on the panels of Figure 6.16, that the dots are gathered, and the groups of dots are separated by two gaps. First, a gap between 1 and 3 processes, which corresponds to the addition of communications, that are not necessary for one process only. Second, a gap between 7 and 9 processes, due to a change of technology. Indeed, processors are gathered in nodes of eight on the cluster that we have used, and the communications are faster within a node.

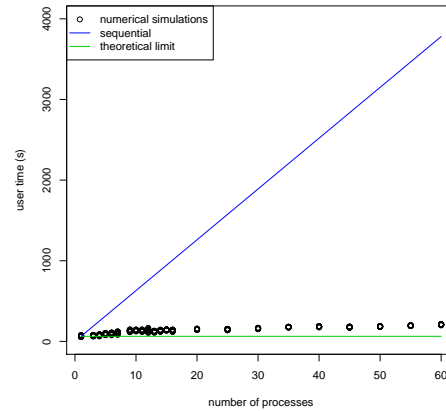
**Conclusion** In this chapter, we have designed a general multiscale scheme for controlled population dynamics, and we have proposed a method of performance analysis in the framework of parallel computing. As an example, we applied successfully the scheme and method to our biological problem, the selection process of ovarian follicles. In appendix A and B the reader can find two "getting started" chapters, on the computational code CodeFollicleMR and the visualization code dualMR, respectively. All the implementation details are detailed in Appendix C.

## Chapitre 6. Design and performance analysis of a multiscale scheme for controlled population dynamics

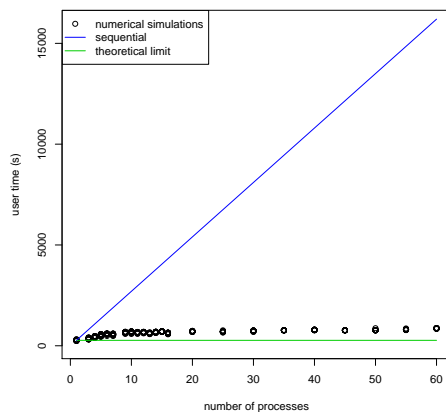
---



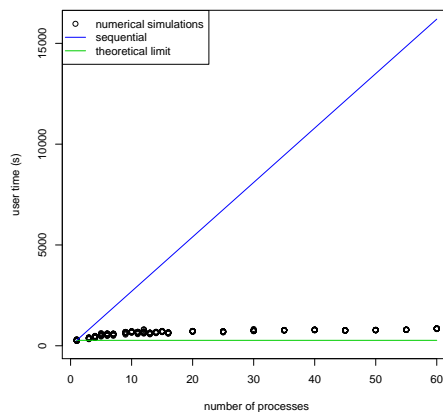
Compact domain  
(one cell cycle)



Unrolled domain  
(eight cell cycles)



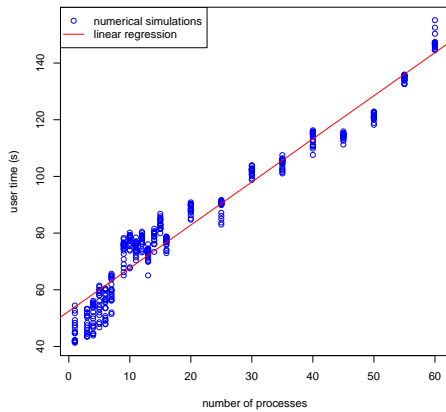
Precision enhancement  
(more meshes on the coarsest level)



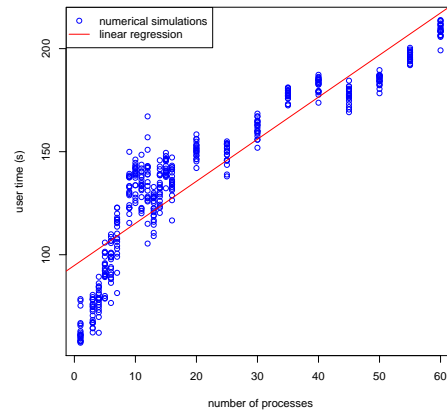
Precision enhancement  
(one more level of MR)

FIGURE 6.13 – Mean weak scalability test. In abscissa, the number of processes, in ordinate, the user time. The green line corresponds to the mean user time for one process. This is the theoretical limit. The blue line indicates the mean sequential behavior, product of the mean user time for one process by the number of processes. The black dots are random independent realizations of the numerical simulations of our model, with perturbed parameters. We note that the size of the problem grows with the number of processes, but the CPU time remains close to the optimal theoretical value.

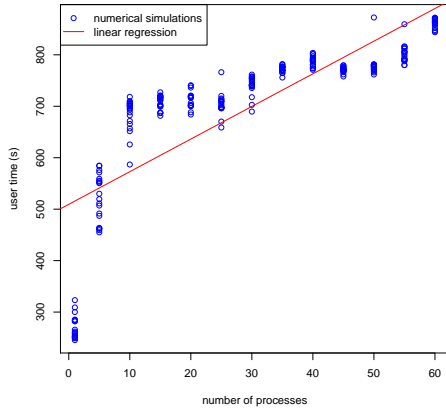
## Chapitre 6. Design and performance analysis of a multiscale scheme for controlled population dynamics



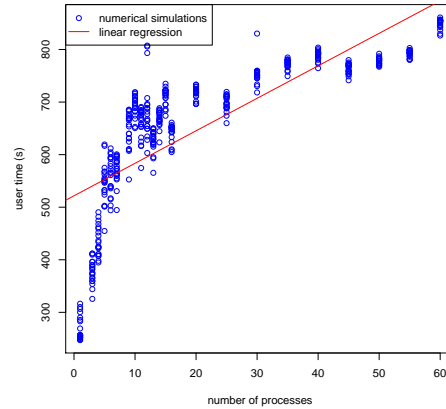
Compact domain  
(one cell cycle)



Unrolled domain  
(eight cell cycles)



Precision enhancement  
(more meshes on the coarsest level)



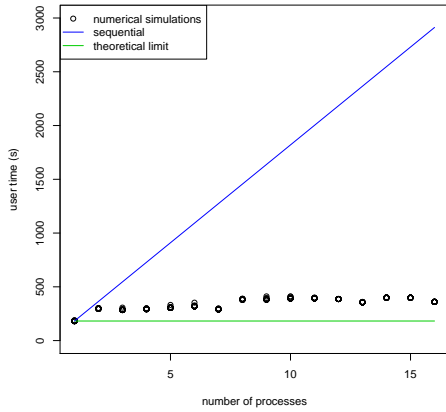
Precision enhancement  
(one more level of MR)

FIGURE 6.14 – Mean weak scalability test. Zoom of the panels of Figure 6.13. The blue dots correspond to the numerical simulations, and the red curve corresponds to the linear regression. The variability tends to decrease with respect to the number of processes. It is interesting to note that there is a gap between the values for a number of process smaller than 8 and the others, due to the technology that we used (the first eight nodes are gathered in a small and fast cluster).

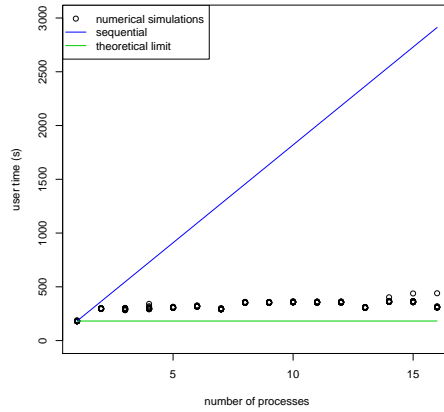


## Chapitre 6. Design and performance analysis of a multiscale scheme for controlled population dynamics

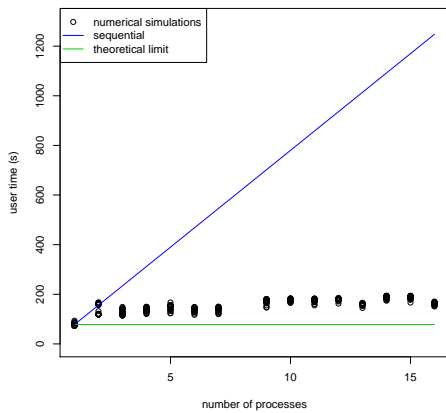
---



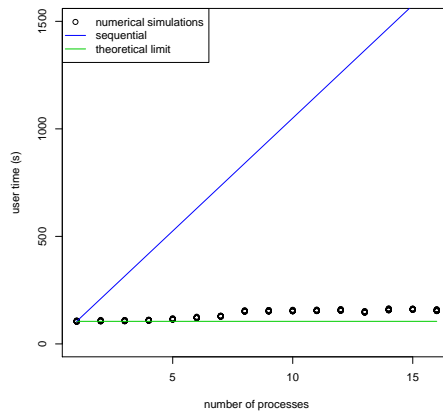
Reference  
(same parameters, uniform mesh,  
constant time step)



Random parameters  
(uniform mesh and constant time  
step)



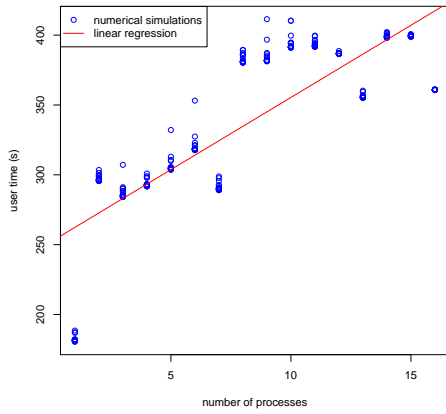
Random parameters and adaptive  
time step  
(uniform mesh)



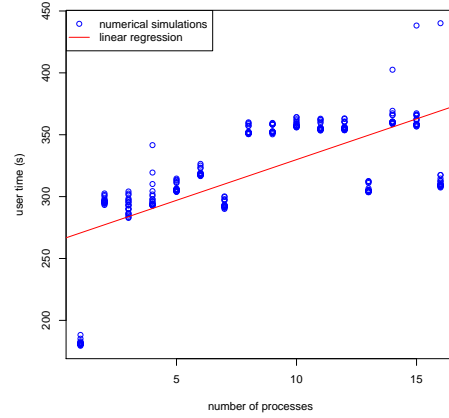
Random parameters and adaptive  
mesh  
(constant time step)

FIGURE 6.15 – Mean weak scalability test. In these cases, we identify the origin of the variability observed in the general case of an adaptive mesh, an adaptive time step, and random parameters (Figures 6.13 and 6.14). Top left panel : reference situation with uniform mesh, constant time step, constant parameters (classical weak scalability test). Top right panel : uniform mesh, constant time step, random parameters. Bottom left panel : uniform mesh, adaptive time step, random parameters. Bottom right panel : adaptive mesh, constant time step, random parameters.

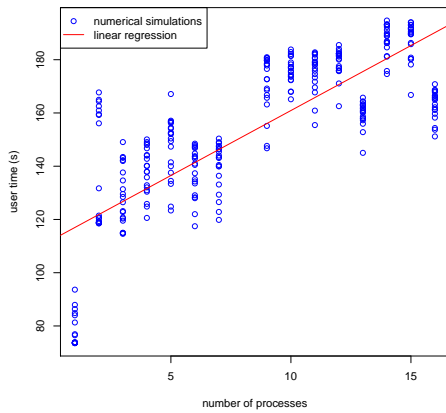
## Chapitre 6. Design and performance analysis of a multiscale scheme for controlled population dynamics



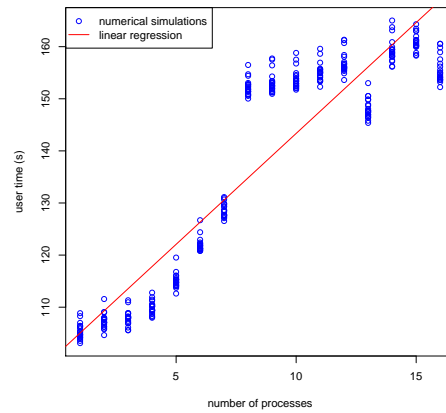
Reference  
(same parameters, uniform mesh,  
constant time step)



Random parameters  
(uniform mesh and constant time  
step)



Random parameters and adaptive  
time step  
(uniform mesh)



Random parameters and adaptive  
mesh  
(constant time step)

FIGURE 6.16 – Mean weak scalability test. Zooms on the panels of Figure 6.15. The bottom left panel corresponding to adaptive time step, uniform mesh and random coefficients exhibits the highest variability. We clearly see the gaps between one and three processors, and between seven and nine processors. The first gap corresponds to the communications that have to be done, and the second to a change of technology in the network that is used for communications (processors are gathered in nodes of eight).



# Conclusion

---

Dans cette thèse, nous avons développé une méthode numérique fiable et performante servant à simuler numériquement un système physiologique complexe modélisant l'action d'un contrôle hormonal sur une population de cellules. Cette méthode numérique est basée sur trois piliers : un schéma numérique prenant en compte à l'ordre élevé les conditions de transmission spécifiques du modèle (qui a fait l'objet d'une publication dans le journal *SIAM J. Sci. Comput.* [7]), une méthode d'adaptation de maillage par multirésolution compatible avec ces conditions (qui a fait l'objet d'une publication dans le journal *M2AN* [8]) et une stratégie de parallélisation basée sur du calcul à mémoire distribuée (qui a fait l'objet d'un *proceeding* [6], et un manuscrit est en préparation).

Ces résultats nous permettent maintenant d'effectuer des campagnes de simulations numériques intensives afin de prolonger les résultats obtenus au chapitre 5 où nous faisons de la calibration de paramètre par une compréhension fine du modèle. Les résultats de ces très nombreuses simulations pourraient être rassemblés dans une base de donnée en vue d'être exploités. Une idée intéressante serait d'effectuer des tests de propagation d'incertitudes, en identifiant par exemple l'impact des différents paramètres sur des sorties agrégées du modèle (ovulation ou non, temps de sortie du cycle de prolifération...). On pourrait aussi faire une exploration méthodique de l'espace des paramètres, en plusieurs étapes : effectuer un tirage aléatoire de paramètres en entrée (respectant les contraintes a priori définies au chapitre 5), effectuer la simulation numérique avec ces paramètres, vérifier a posteriori si la simulation est admissible ou non du point de vue biologique (selon les critères aussi établis au chapitre 5), et ainsi réduire l'espace des paramètres lorsque c'est nécessaire. Une démarche de ce type a été effectuée dans [63], sur une version simplifiée du modèle. Il est aujourd'hui possible d'accélérer encore les calculs, en utilisant des techniques de GPU. Même si, comme nous l'avons vu au chapitre 6, la stratégie de parallélisation développée est très performante, ces campagnes de simulation bénéficieraient d'une réduction supplémentaire du temps de calcul. Il s'agit de techniques de calcul parallèle basées sur une technologie de type carte graphique (GPU, pour *Graphic Process Unit*), par opposition au calcul parallèle exploité dans la thèse, basé sur une technologie de processeur classique (CPU, pour *Control Process Unit*). Ces techniques ont plusieurs avantages (voir [14]). Elles sont notamment beaucoup plus rapides que du calcul parallèle classique (certains services météo comme le *Met Office* [37], mais aussi certaines banques comme la *BNP Paribas*, l'ont testé avec succès) et plus accessibles (en effet, il est plus facile et moins onéreux d'investir dans une carte graphique pour un ordinateur personnel que dans un cluster, sans parler d'économie

## Chapitre 7. Conclusion

---

en terme de maintenance, et d'énergie électrique). Par ailleurs, les techniques de calcul sur GPU se prêtent très bien aux méthodes de type Monte Carlo utilisées au chapitre 6 (voir par exemple [4]).

Un code rapide est indispensable pour créer une interface graphique (GUI) destinée aux biologistes (voir par exemple [57]). Une telle interface pourrait combiner les codes de calcul et de visualisation. Dans un premier temps, cela permettrait d'entrer les paramètres et visualiser les résultats, dans un environnement ergonomique et simple d'utilisation. A plus long terme, on pourrait imaginer une console permettant de faire des simulations numériques en temps réel, avec possibilité d'interrompre la simulation, prendre la main sur les variables de contrôle pour les modifier, ou même revenir en arrière. Ceci est une motivation supplémentaire à l'adaptation du code à une technologie GPU. En effet, il est plus raisonnable en pratique de proposer aux biologistes une interface qu'ils peuvent utiliser sur leur machine, en local, que sur un cluster.

Enfin, disposer d'un code performant faciliterait le couplage du modèle avec d'autres modèles. Le fait que la simulation numérique ne soit plus un facteur limitant ouvre en effet de nouvelles perspectives. On pourrait par exemple affiner le modèle de dynamique cellulaire, et définir les valeurs des vitesses de maturation et de vieillissement non plus à partir de fonctions, comme c'est le cas dans la thèse, mais à partir de résultats de simulations numériques de modèles de dynamique biochimiques à un niveau encore plus fin de dynamique cellulaire, en modélisant la réponse au contrôle hormonal à l'échelle intracellulaire à l'aide d'un réseau de signalisation hormonale (par exemple similaire à celui développé dans [46]).

# CodeFollicleMR : Notice utilisateur (getting started)

---

## Sommaire

---

<b>A.1 Simulation avec paramètres par défaut . . . . .</b>	<b>203</b>
<b>A.2 Visualisation . . . . .</b>	<b>203</b>
<b>A.3 Simulation avec plusieurs follicules, en choisissant un jeu de paramètre . . . . .</b>	<b>204</b>
<b>A.4 Simulation en créant son propre jeu de paramètres . . . . .</b>	<b>205</b>
<b>A.5 Interface Utilisateur . . . . .</b>	<b>206</b>

---

**CodeFollicleMR** CodeFollicleMR est un code de calcul développé en C++/MPI. Ce logiciel permet de simuler numériquement le processus de sélection des follicules ovariens. Le modèle et les algorithmes de calculs ont été définis dans la thèse. Le logiciel nécessite d'être lancé sur un système d'exploitation Linux ou sur Mac. Dans cette section nous proposons à l'utilisateur de suivre pas à pas une suite d'instructions. Cela l'amènera à découvrir les fonctionnalités du logiciel, des plus basiques aux plus avancées.

### A.1 Simulation avec paramètres par défaut

Pour un premier contact, le plus simple est de lancer une simulation en utilisant des paramètres par défaut. On commence par ouvrir un terminal. On choisit un nombre  $np$  de follicule, et on entre en ligne de commande

```
mpirun -np 1 follicleMR
```

Le terminal affiche des détails sur la simulation (voir Figure A.1). On peut ensuite visualiser les résultats.

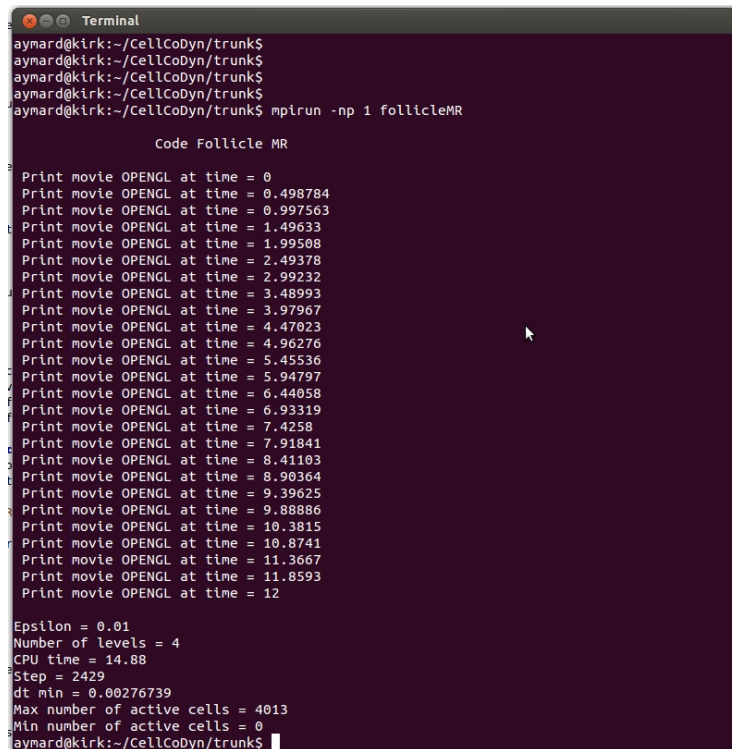
### A.2 Visualisation

On peut visualiser les résultats de plusieurs façons. Les sorties macro peuvent être visualisées avec le logiciel Open source gnuplot avec la commande

```
load "macro.plt"
```

## Annexe A. CodeFollicleMR : Notice utilisateur (getting started)

---



```
aymard@kirk:~/CellCoDyn/trunk$
aymard@kirk:~/CellCoDyn/trunk$
aymard@kirk:~/CellCoDyn/trunk$
aymard@kirk:~/CellCoDyn/trunk$
aymard@kirk:~/CellCoDyn/trunk$ mpirun -np 1 follicleMR

Code Follicle MR

Print movie OPENGL at time = 0
Print movie OPENGL at time = 0.498784
Print movie OPENGL at time = 0.997563
Print movie OPENGL at time = 1.49633
Print movie OPENGL at time = 1.99508
Print movie OPENGL at time = 2.49378
Print movie OPENGL at time = 2.99232
Print movie OPENGL at time = 3.48993
Print movie OPENGL at time = 3.97967
Print movie OPENGL at time = 4.47023
Print movie OPENGL at time = 4.96276
Print movie OPENGL at time = 5.45536
Print movie OPENGL at time = 5.94797
Print movie OPENGL at time = 6.44058
Print movie OPENGL at time = 6.93319
Print movie OPENGL at time = 7.4258
Print movie OPENGL at time = 7.91841
Print movie OPENGL at time = 8.41103
Print movie OPENGL at time = 8.90364
Print movie OPENGL at time = 9.39625
Print movie OPENGL at time = 9.88886
Print movie OPENGL at time = 10.3815
Print movie OPENGL at time = 10.8741
Print movie OPENGL at time = 11.3667
Print movie OPENGL at time = 11.8593
Print movie OPENGL at time = 12

Epsilon = 0.01
Number of levels = 4
CPU time = 14.88
Step = 2429
dt min = 0.00276739
Max number of active cells = 4013
Min number of active cells = 0
aymard@kirk:~/CellCoDyn/trunk$
```

FIGURE A.1 – Affichage du terminal à l'issue d'une simulation avec CodeFollicleMR.

pour afficher les valeurs macroscopiques (voir Figure A.2).

Pour la partie micro, on peut utiliser le logiciel open source Paraview.

Une autre solution est d'utiliser le logiciel dédié dualMR, qui permet de visualiser à la fois la partie micro et la partie macro (voir Annexe B).

### A.3 Simulation avec plusieurs follicules, en choisissant un jeu de paramètre

Dans un premier temps, on peut partir de jeux de paramètres prédéfinis, ceux là même qui furent utilisés pour la rédaction des articles constituant la thèse. Ces derniers se trouvent dans le dossier Input. On peut par exemple copier un jeu de paramètres pour 4 follicules, en entrant en ligne de commande

```
cp Input/4FBook.par 4FBook.par
```

Ensuite, la syntaxe à l'exécution est la suivante

```
mpirun -np 4 follicleMR @4FBook.par
```

On peut ensuite visualiser le résultat.

## Annexe A. CodeFollicleMR : Notice utilisateur (getting started)

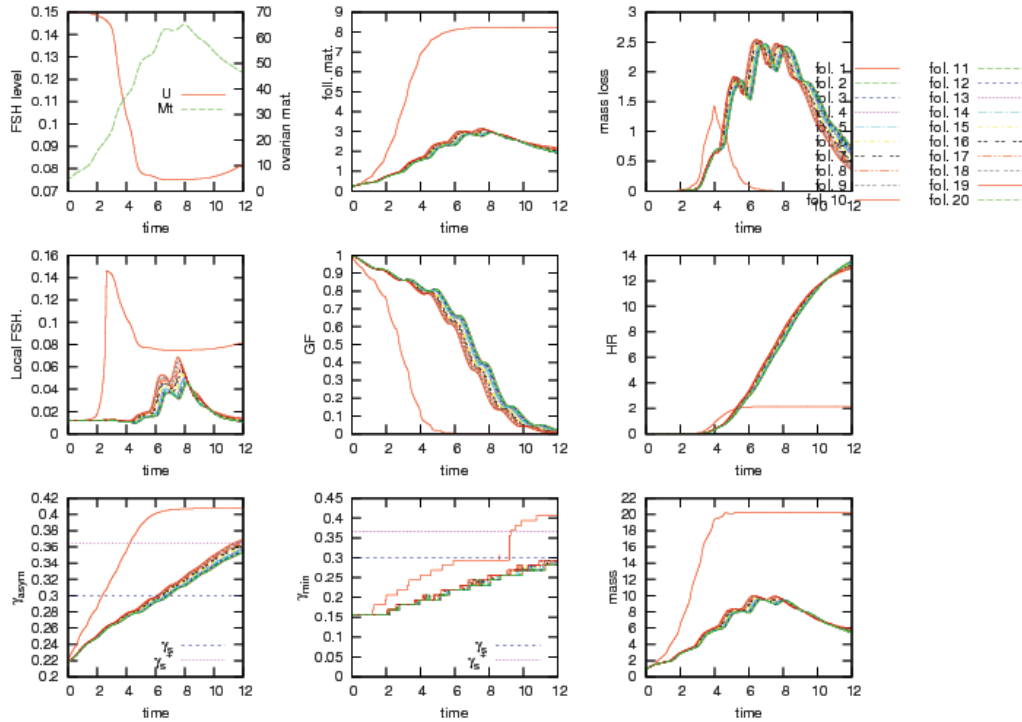


FIGURE A.2 – Exemple d’affichage de résultats macroscopiques, avec le logiciel libre *gnuplot*, à l’issue d’un calcul effectué avec *CodeFollicleMR*.

### A.4 Simulation en créant son propre jeu de paramètres

L’étape suivante est naturellement de créer son propre jeu de paramètres. Pour cela il suffit d’ouvrir un fichier, que l’on sauvegardera au format `.par`. Une seule déclaration de paramètre par ligne, sous la forme

`-monParametre=maValeur`

pour les paramètres globaux, et sous la forme

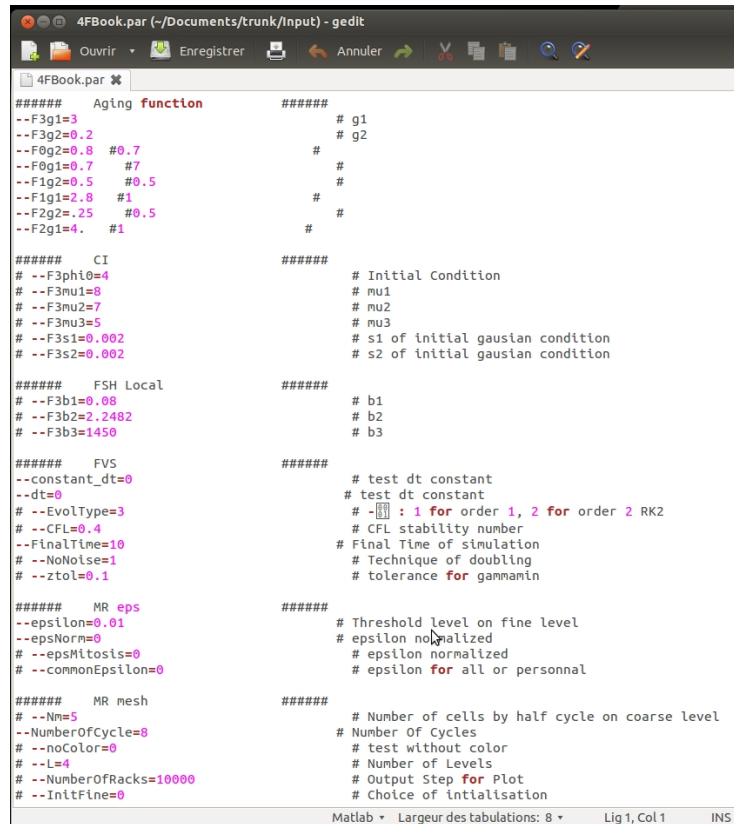
`-FpmonParametre=maValeur`

pour les paramètres locaux, en remplaçant  $p$  par un numéro de follicule (voir l’exemple de fichier de paramètre A.3). En pratique, on peut aussi se contenter de perturber un fichier de paramètres existant dans le dossier `Input`. La commande d’exécution est la même que pour un jeu prédéfini, il suffit d’indiquer le nom de son fichier de paramètres :

`mpirun -np 4 follicleMR @nomDuFichier.par`



## Annexe A. CodeFollicleMR : Notice utilisateur (getting started)



```
##### Aging fonction #####
--F3g1=3 # g1
--F3g2=0.2 # g2
--F0g2=0.8 #0.7 #
--F0g1=0.7 #7 #
--F1g2=0.5 #0.5 #
--F1g1=2.8 #1 #
--F2g2=-.25 #0.5 #
--F2g1=4. #1 #

##### CI #####
# --F3phi0=4 # Initial Condition
# --F3mu1=8 # mu1
# --F3mu2=7 # mu2
# --F3mu3=5 # mu3
# --F3s1=0.002 # s1 of initial gaussian condition
# --F3s2=0.002 # s2 of initial gaussian condition

##### FSH Local #####
# --F3b1=0.08 # b1
# --F3b2=2.2482 # b2
# --F3b3=1450 # b3

##### FVS #####
--constant_dt=0 # test dt constant
--dt=0 # test dt constant
# --EvolType=3 # -[1] : 1 for order 1, 2 for order 2 RK2
# --CFL=0.4 # CFL stability number
--FinalTime=10 # Final Time of simulation
# --NoNoise=1 # Technique of doubling
# --ztol=0.1 # tolerance for gammamin

##### MR eps #####
--epsilon=0.01 # Threshold level on fine level
--epsNorm=0 # epsilon normalized
# --epsMltosts=0 # epsilon normalized
# --commonEpsilon=0 # epsilon for all or personal

##### MR mesh #####
# --Nm=5 # Number of cells by half cycle on coarse level
--NumberOfCycle=8 # Number Of Cycles
# --noColor=0 # test without color
# --L=4 # Number of Levels
# --NumberOfRacks=10000 # Output Step for Plot
# --IntFine=0 # Choice of Initialisation
```

FIGURE A.3 – Exemple d’un fichier de paramètre utilisé par CodeFollicleMR. Les lignes précédées d’un dièse sont commentées, et ne sont donc pas prises en compte. Pour modifier un paramètre, il faut décommenter la ligne, et indiquer la valeur, sinon c’est la valeur par défaut définie dans le programme qui est utilisée.

### A.5 Interface Utilisateur

Au cours de la thèse, nous avons commencé, en collaboration avec Serge Steer (Inria Paris-Rocquencourt) la réalisation d’un interface utilisateur permettant d’entrer les paramètres, de faire une pré-visualisation (abaques du chapitre portant sur la calibration), puis de simuler et visualiser les résultats. Ces travaux sont amenés à être poursuivis, comme évoqué dans la conclusion. La Figure A.4 est une capture d’écran de cette console.

## Annexe A. CodeFollicleMR : Notice utilisateur (getting started)

---

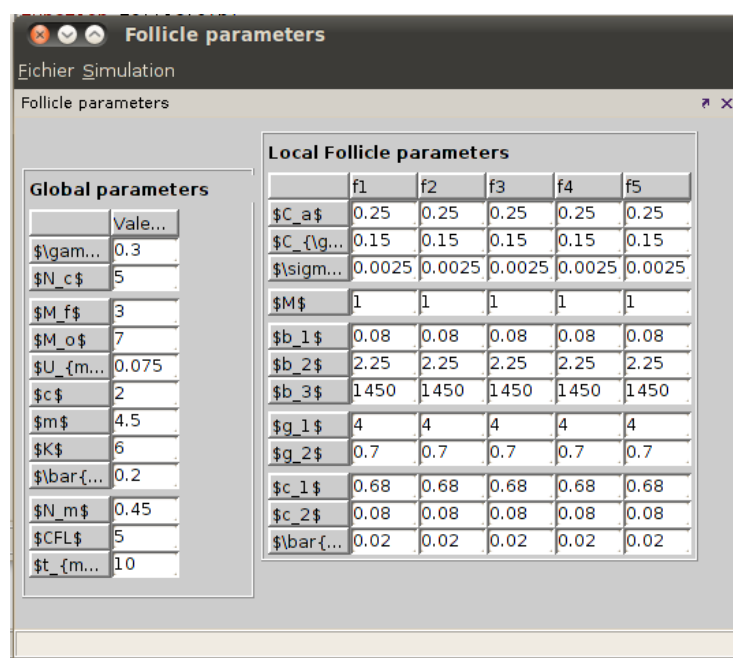


FIGURE A.4 – Exemple d'interface graphique, réalisée en collaboration avec Serge Steer (Inria Paris-Rocquencourt), dédiée à la simulation numérique du processus de sélection des follicules ovariens.



# dualMR : Notice utilisateur (getting started)

---

## Sommaire

---

<b>B.1</b>	<b>Comment utiliser dualMR . . . . .</b>	<b>209</b>
<b>B.2</b>	<b>Visualiser les sorties macroscopiques et mésoscopiques . . .</b>	<b>209</b>
<b>B.3</b>	<b>Visualiser les sorties microscopiques . . . . .</b>	<b>210</b>

---

Le logiciel **dualMR** est un code de visualisation développé en C++/OpenGL. Il permet de visualiser à la fois les sorties microscopiques et les sorties macroscopiques relatives au modèle de processus de sélection des follicules ovariens.

## B.1 Comment utiliser dualMR

A la suite d'un calcul avec CodeFollicleMR, on peut visualiser les résultats en entrant dans un terminal la commande

```
./dualMR
```

Le logiciel affiche deux fenêtre. Une première fenêtre concerne les valeurs macroscopiques B.1, et l'autre les valeurs microscopiques B.2.

## B.2 Visualiser les sorties macroscopiques et mésoscopiques

Les sorties macroscopiques et mésoscopiques se visualisent dans la fenêtre appelée **macro** (voir Figure B.1). Le clavier correspondant est donné dans le tableau B.1.

Les commandes + et - du clavier permettent de passer d'une valeur à une autre, dans la liste suivante

- $U(t)$
- $m_0^f(t)$
- $m_1^f(t)$
- $GF^f(t)$
- $R_f(t)$
- $\Delta t_f^n$
- $r_f(t)$
- $M_1(t)$
- $R_f(t)$

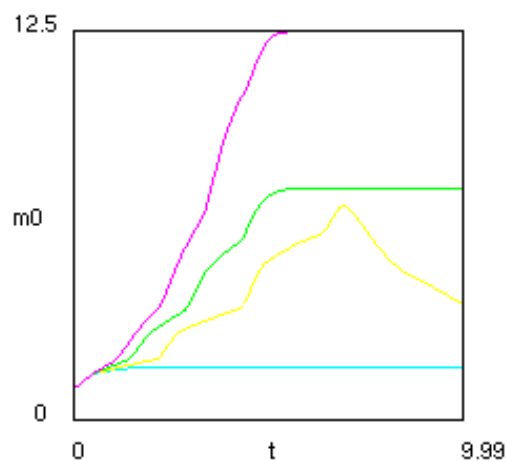


FIGURE B.1 – Exemple d’affichage de résultats macroscopiques, avec le logiciel dualMR, à l’issue d’un calcul effectué avec CodeFollicleMR.

- $\gamma_{\min}^f(t)$
- $\epsilon_f(t)$
- $I_m^f(t)$

On peut aussi faire une impression au format `.ppm` avec la touche `p`.

### B.3 Visualiser les sorties microscopiques

La seconde fenêtre concerne les sorties microscopiques B.2. Par défaut, s’il n’y a qu’un seul follicule, elle affiche le maillage et la densité. Dans le cas général, elle affiche les densités des follicules les unes au dessous des autres. Le Tableau B.2 présente les touches du clavier.

## Annexe B. dualMR : Notice utilisateur (getting started)

Touche	Effet
w	déplacer le cadre de visualisation à gauche
x	déplacer le cadre de visualisation en bas
c	déplacer le cadre de visualisation à droite
s	déplacer le cadre de visualisation en haut
i	zoom in
o	zoom out
+	passer à la sortie meso/macro suivante (voir liste)
-	passer à la sortie meso/macro précédente (voir liste)
q	quitter
p	imprimer une image (au format ppm)

TABLE B.1 – Clavier du logiciel dualMR concernant la fenêtre macro.

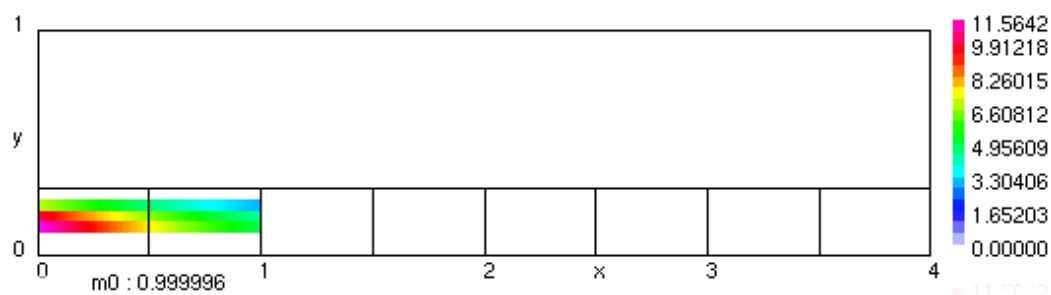


FIGURE B.2 – Exemple d'affichage de résultats microscopiques, avec le logiciel dualMR, à l'issue d'un calcul effectué avec CodeFollicleMR.

Touche	Effet
w	déplacer le cadre de visualisation à gauche
x	déplacer le cadre de visualisation en bas
c	déplacer le cadre de visualisation à droite
s	déplacer le cadre de visualisation en haut
i	zoom in
o	zoom out
+	avancer d'un pas de temps
-	reculer d'un pas de temps
q	quitter
p	imprimer une image (au format ppm)
m	créer une animation
M	créer et sauvegarder une animation (format ppm)
d	afficher/masquer les sous domaines
a	changer le code couleur

TABLE B.2 – Clavier du logiciel dualMR concernant la fenêtre micro.



# CodeFollicleMR : Notice programmeur

---

## Sommaire

---

<b>C.1 Structures de données . . . . .</b>	<b>213</b>
<b>C.2 Organisation du code source . . . . .</b>	<b>217</b>
<b>C.3 Entrées/sorties . . . . .</b>	<b>219</b>

---

Dans cette section sont regroupées les informations utiles à la modification du code, pour des questions de maintenance, ou d'améliorations futures. Les algorithmes utilisés ont été détaillés dans le corps du manuscrit. On détaille dans ce document les structures de données adéquates, l'organisation du code source en plusieurs modules, et la gestion des entrées/sorties de données.

## C.1 Structures de données

L'idée générale dans la définition de structures de données est d'optimiser au maximum les calculs. Cela passe par la limitation de la mémoire utilisée et l'accès rapide en mémoire cache.

**Convention code couleur** Les flux étant définis par sous domaines, nous utilisons une technique de coloriage du domaine. Pour cela, on se donne la convention suivante : 0 zone G1, 1 zone SM, 2 zone D, 3 mailles à droite de l'interface SM-G1, 4 mailles au nord de l'interface SM-D.

L'ajout des couleurs 3 et 4 permet d'identifier les zones ou les conditions de transmission doivent être prises en compte. Comme nous l'avons vu dans le chapitre condition de transmission, il n'est pas nécessaire de modifier les calculs à l'interface G1-SM, car nous avons pris soin de respecter la place des interfaces dans la définition du maillage, et que nous évaluons les vitesses au centre des mailles.

**Classe Cell** Commençons par définir une structure de données qui servira à recueillir l'information de chaque maille. Celle ci contient au moins la valeur, le bilan de flux et la couleur de chaque maille. Afin d'optimiser le temps de calcul, et plus précisément pour ne pas perdre de temps à rechercher les voisins en parcourant tout le domaine, nous définissons des stencils de projection C.1, prédiction C.2 et volumes



## Annexe C. CodeFollicleMR : Notice programmeur

---

finis C.3, que nous garderons en mémoire. En effet, plusieurs accès successifs sont faits sur ces stencils, notamment lors des opérations de codage et de décodage.

1	2
3	0

FIGURE C.1 – Convention utilisée pour la numérotation des cellules-filles du stencil de projection, que l'on range dans la cellule-fille 0. On note qu'un tel stencil n'existe pas au niveau zéro, car c'est inutile.

6	7	8
3		5
	4	
0	1	2

FIGURE C.2 – Convention utilisée pour la numérotation des cellules du stencil de prédiction.

Pour cela, on utilise une classe, que l'on nommera Cell. On notera que l'on a ajouté deux pointeurs dont nous verrons l'utilité par la suite.

- Valeur  $\phi_{i,j}^{n,l}$  (réel) ;
- Bilan de flux  $D_{i,j}^{n,l}$  (réel) ;
- Couleur (réel) ;
- StencilProjection (tableau de 4 pointeurs vers le type Cell) ;
- StencilPrediction (tableau de 9 pointeurs vers le type Cell) ;
- StencilFVS (tableau de 16 pointeurs vers le type Cell) ;
- NextInLevel (pointeur vers un type Cell) ;
- KeyAddress (pointeur vers un type Key).

**Listes chaînées de niveaux** Plusieurs algorithmes nécessitent de parcourir le maillage hybride niveau par niveau : codage, décodage, calcul des flux, mise à jour

## Annexe C. CodeFollicleMR : Notice programmeur

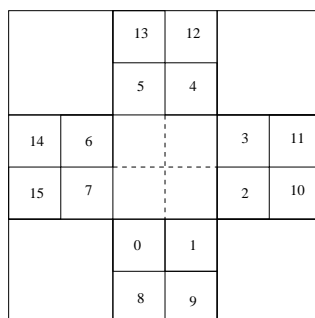
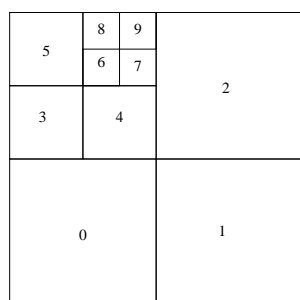


FIGURE C.3 – *Convention utilisée pour le stockage des cellules du stencil volume fini.*

des mailles. Pour faciliter cette opération, nous allons chaîner les mailles d'un même niveau, ce qui autorisera les boucles sur un niveau, et nous allons garder en mémoire dans un tableau les adresses mémoires des débuts de chaque liste de niveau, ce qui permettra de faire une boucle sur les niveaux. Pour cela nous définissons un tableau de pointeurs vers le type Cell

– StartOfLevel (array)

et chaque maille contiendra un pointeur vers la maille suivante dans le niveau, que l'on nommera NextInLevel. C'est l'un des deux pointeurs que l'on a ajouté à la classe Cell.



Hybrid Grid

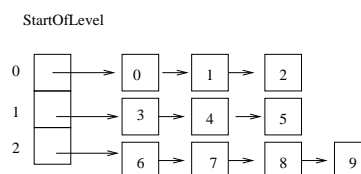


FIGURE C.4 – *Listes chaînées de niveaux. Cette structure de données est très utile pour passer en revue les mailles d'un niveau donné (par exemple dans les algorithmes de projection et de prédiction). Les cellules sont chaînées par groupes de 4 cellules-filles, sauf sur le premier niveau. Les pointeurs sont stockés dans les cellules-filles numéro 0. Le tableau StartOfLevel contient les adresses des premières cellules de chaque niveau, et permet ainsi d'initialiser les boucles sur les niveaux.*

**Table de hashage** A l'initialisation, ainsi qu'à l'étape de prédiction, il est nécessaire de parcourir le maillage hybride, et éventuellement d'ajouter ou supprimer des mailles. La structure de donnée idéale pour ce genre d'opérations est la table de hashage.

Rappelons les grands principes d'une table de hachage. Pour fixer les idées, nous utiliserons une analogie avec des chambres d'hôtels. Pour l'exemple, on considérera des chambres un peu particulières, où l'on peut installer plusieurs lits à la demande. On se donne une taille mémoire au départ ; cela correspond au nombre de lits disponibles.

Pour placer une nouvelle personne dans l'hôtel, dont le nombre de chambres est fixé, on dispose d'un outil, appelé fonction de hachage, qui permet d'attribuer de manière déterministe un numéro de chambre à un client en fonction de ses caractéristiques. Deux cas peuvent alors se produire. La chambre est vide, dans ce cas, on place un lit pour le client. La chambre est occupée, et dans ce cas, on place un lit à la suite des lits des autres clients.

Pour trouver l'emplacement d'une personne, on ré utilise notre outil, ce qui nous dit dans quelle chambre chercher, puis on parcourt les lits les uns après les autres, jusqu'à la trouver (nous sommes sûr de la trouver là, le numéro de chambre étant attribué de manière unique).

De même, lorsqu'une personne quitte l'hôtel, nous cherchons son emplacement avec la méthode précédente, puis nous enlevons son lit et le rangeons dans la réserve, et on déplace toutes les personnes suivantes pour combler le trou.

Les tables de hachage ne sont pas plus compliquées. On commence par se donner un identifiant par objet, que l'on appellera une clé (Figure C.1). Pour cela, on définit une classe, que nous appellerons Key. L'idée étant que l'on va chercher une position parmi les identifiants, et que sur chaque identifiant on aura un lien vers la personne souhaitée, il faut que cette structure soit la plus légère possible.

- CellAddress ;
- PreviousInRack ;
- NextInPool ;
- indices (i,j) (caractéristiques de l'objet).

Les paires mailles-identifiants (ou clients d'hôtel-identifiant dans notre exemple) étant connues, on alloue en mémoire de la place pour les lits.

Ensuite, nous allons ranger les clés de deux manières différentes, suivant leur utilisation. L'avantage est qu'en informatique, ces deux méthodes peuvent coexister.

On veut d'une part pouvoir trouver directement une clé libre, lorsqu'on cherche à ajouter une maille. Pour cela, on définit une liste chaînée de clés, chacune pointant vers la maille qui la précédé, via le pointeur PreviousInRack de la classe Key, et on se donne un pointeur vers la première clé libre.

- FirstFreeKey (int) ;
- Racks (array) ;

D'autre part, on veut pouvoir minimiser le temps passé à chercher une clé en mémoire. Pour cela on définit une fonction de hachage. Cette dernière doit être adaptée au maximum au problème considéré. Dans notre cas, nous avons choisit une fonction

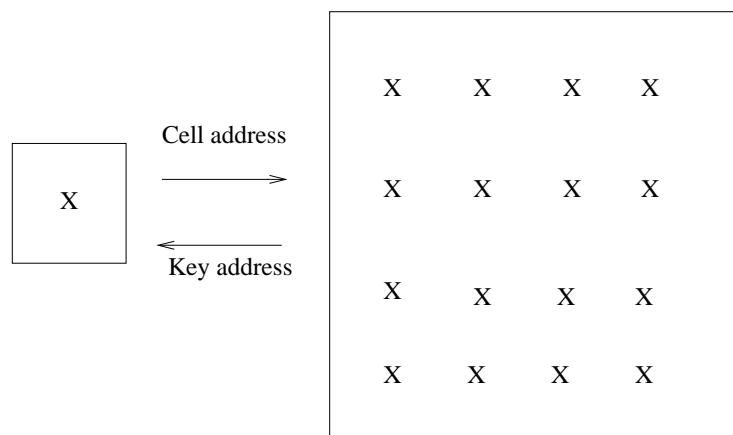


FIGURE C.5 – Couple clé-maille. Les clés sont des structures de données très légères, utilisées par la table de hachage, contrairement aux mailles, qui contiennent toutes les données nécessaires aux calculs. Des pointeurs permettent de passer de l'un à l'autre.

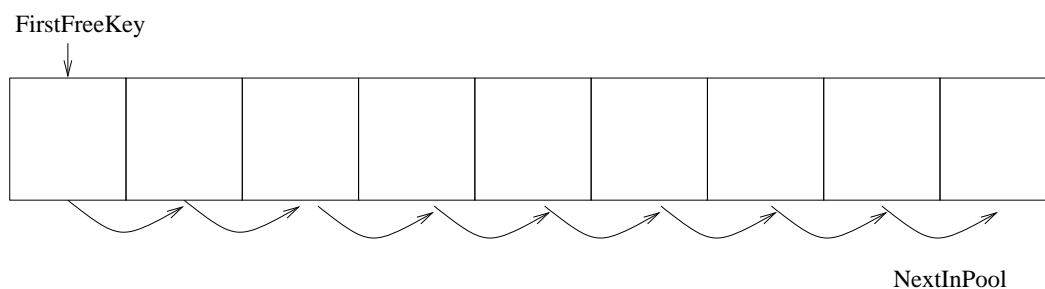


FIGURE C.6 – Réserve de clés pour la table de hachage. Toutes les clés sont créées et chaînées au départ, chacune pointant vers la suivante. Un pointeur, appelé *FirstFreeKey*, indique la première clé disponible.

simple, basée sur la représentation géométrique en 2D du maillage au niveau le plus fin.

$$hash(i, j) = (i + jNL_x) \bmod (NumberOfRacks).$$

Cela laisse un degré de liberté, dans le paramètre `NumberOfRacks`.

## C.2 Organisation du code source

**class.h** Ce fichier contient la définition des variables globales et des classes de données utilisées.

**hash.cpp** Ce fichier contient les fonctions relatives à l'utilisation des tables de hashages : définition, ajout d'un élément, recherche d'un élément, suppression d'un

## Annexe C. CodeFollicleMR : Notice programmeur

---

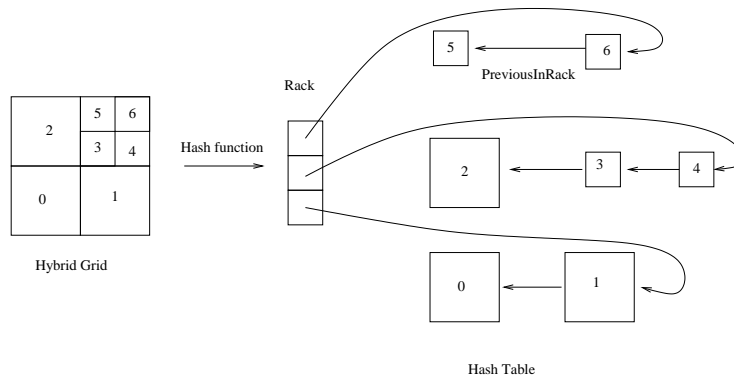


FIGURE C.7 – Table de haschage. Cette structure de données permet un accès rapide aux cellules, via leur clés. Elle est utilisée pour la création et la suppression de mailles (algorithme de prédiction, algorithme d’initialisation).

élément.

**MR.cpp** Ce fichier contient les fonctions relatives à la multirésolution. Initialisation par le niveau fin et le niveau grossier, codage, décodage, seuillage, reconstruction du niveau le plus fin (pour les calculs d’erreur).

**FVS.cpp** Ce fichier contient les fonctions relatives à la méthode des volumes finis. Tout d’abord les fonctions applicables à la méthode classique en maillage uniforme : calcul de flux horizontaux et verticaux (prend en compte les conditions de transmission), calcul de flux sur tout le maillage, méthode de Runge Kutta pour mettre à jour les mailles, calcul du pas de temps. Ensuite, les fonctions utiles à l’adaptation de maillage : prédiction du maillage, reconstruction des fantômes à l’ordre élevé.

**follicleMR.cpp** Ce fichier contient le programme principal codant l’algorithme général.

**input.cpp** Ce fichier contient la fonction de lecture des paramètres compatibles avec le parser.

**output.cpp** Ce fichier contient les fonctions de sorties du code : création de fichier de sorties mésoscopiques et macroscopiques, création de fichier de sorties microscopiques (format OpenGL ou Paraview), sorties dans le terminal.

**biologie.cpp** Ce fichier contient les fonctions liées à la biologie : différentes conditions initiales, vitesses de vieillissement et de maturation.

**errorMR.cpp** Ce fichier contient une fonction permettant d'évaluer l'erreur entre deux solutions. Elle est principalement utilisée via les scripts du Benchmark pour faire des courbes de convergence.

**dualMR.cpp** Ce fichier contient le code source du logiciel de visualisation.

### C.3 Entrées/sorties

**Paramètres en entrée** Le passage de paramètre dans le code de calcul se fait via un parser (ce parser a été développé par Marc Schoenauer, chef de l'équipe-projet TAO à l'Inria Saclay). Tous les paramètres ont une valeur par défaut définie dans `input.cpp`. On peut passer un fichier de paramètres à l'exécution. Note importante, il n'y a qu'un seul fichier de paramètres en entrée, contenant les paramètres de tous les processeurs, et non pas un fichier de paramètres par processeur. Chacun des noeuds de calcul lit une copie de ce fichier de paramètres et va récupérer les paramètres qui le concerne. Cela se fait en récupérant le numéro du noeud de calcul, et en créant dynamiquement les noms de paramètres sous le format

FpParametre

F (pour follicle) précède tous les paramètres en parallèle, p est le numéro du noeud de calcul, et Parametre est le nom du paramètre (par exemple F0g2 désigne le paramètre g2 du noeud de calcul 0). Un ensemble de jeux de paramètres est disponible dans le dossier Input.

**Output macroscopiques et mésoscopiques** A chaque itération, plusieurs valeurs macroscopiques et mésoscopiques sont mises à jour. On sauvegarde leur valeur dans un fichier au format

macrop.dat

Comme pour les fichiers de paramètres, les noms de fichier sortie sont créés dynamiquement en fonction du numéro p de processeur (par exemple macro2.dat contient les sorties macroscopiques et mésoscopiques relatives au follicule 2).

**Output microscopiques** Deux types de sorties microscopiques peuvent être faites. Dans les deux cas, ces dernières sont faites au format binaire, pour limiter les erreurs de conversion au format décimal.

Des sorties à intervalle régulier, espacées de `plotStep`, permettent de réaliser des animations. Ces sorties peuvent se faire en deux format : OpenGL pour être utilisés avec le code de visualisation dualMR, ou le logiciel open source Paraview.

On peut aussi faire une unique sortie à la fin de la simulation, pour faire des calculs d'erreur avec errorMR par exemple. Pour cela la solution, définie sur le maillage hybride, est calculée sur un maillage uniforme, par complétion (on ajoute des détails à zéro et on reconstruit sur le niveau fin par interpolation).

## **Annexe C. CodeFollicleMR : Notice programmeur**

---

Ici encore, les noms des fichiers de sorties sont créés de façon dynamique à l'exécution en fonction des numéros de noeuds de calcul. Par exemple le fichier

micro*p.i*

contient la solution du processeur *p* à l'itération *i* (par exemple micro1.32 contient la sortie microscopique relative au follicule 1 au pas de temps 32).

# Bibliographie

- [1] ADIMURTHI, S. MISHRA & G. D. V. GOWDA – « Optimal entropy solutions for conservation laws with discontinuous flux-functions », J. Hyperbolic Differ. Equ. **2** (2005), no. 4, p. 783–837. (Cité en pages 50 et 51.)
- [2] M. ADIMY, F. CRAUSTE & S. RUAN – « A mathematical study of the hematopoiesis process with applications to chronic myelogenous leukemia », SIAM J. Appl. Math. **65** (2005), no. 4, p. 1328–1352. (Cité en page 70.)
- [3] A. AMBROSO, C. CHALONS, F. COQUEL, E. GODLEWSKI, F. LAGOUTIÈRE, P.-A. RAVIART & N. SEGUIN – « Relaxation methods and coupling procedures », Internat. J. Numer. Methods Fluids **56** (2008), no. 8, p. 1123–1129. (Cité en page 51.)
- [4] G. ARAMPATZIS, M. A. KATSOULAKIS & P. PLECHAC – « Parallelization, processor communication and error analysis in lattice kinetic monte carlo », SIAM J. Numer. Anal. **52** (2014), p. 1156–1182. (Cité en page 202.)
- [5] E. AUDUSSE & B. PERTHAME – « Uniqueness for scalar conservation laws with discontinuous flux via adapted entropies », Proc. Roy. Soc. Edinburgh Sect. A **135** (2005), no. 2, p. 253–265. (Cité en page 51.)
- [6] B. AYMARD, F. CLÉMENT, F. COQUEL & M. POSTEL – « Numerical simulation of the selection process of the ovarian follicles. », ESAIM : Proc. (2012). (Cité en pages i, iii, 2, 13, 14, 25, 52, 70, 73, 83, 105, 112, 171, 186 et 201.)
- [7] — , « A numerical method for transport equations with discontinuous flux functions : Application to mathematical modeling of cell dynamics. », SIAM J. Sci. Comput. **35** (2013), p. A2442–A2468. (Cité en pages i, iii, 15, 16, 17, 49, 83, 94, 105, 112, 180 et 201.)
- [8] B. AYMARD, F. CLÉMENT & M. POSTEL – « Adaptive mesh refinement strategy for a non conservative transport problem », ESAIM : Math. model. Numer. Anal. (M2AN) **00** (2014), p. 00. (Cité en pages i, iii, 7, 18, 81, 180 et 201.)
- [9] D. BALE, R. LEVEQUE, S. MITRAN & J. ROSSMANITH – « A wave propagation method for conservation laws and balance laws with spatially varying flux functions », SIAM J. Sci. Comput. **24** (2002), p. 955–978. (Cité en page 13.)
- [10] B. BIHARI & A. HARTEN – « Multiresolution schemes for the numerical solution of 2-D conservation laws », SIAM J. Sci. Comput. **18** (1997), p. 315–354. (Cité en pages 15, 172 et 174.)
- [11] F. BOUCHUT – Nonlinear stability of finite volume methods for hyperbolic conservation laws, and well-balanced schemes for sources, Birkhauser, Basel, 2004. (Cité en page 170.)
- [12] F. BOUCHUT & F. JAMES – « One-dimensional transport equations with discontinuous coefficients », Nonlinear Anal. **32** (1998), no. 7, p. 891–933. (Cité en pages 31 et 51.)



## Bibliographie

---

- [13] B. BOUTIN, F. COQUEL & L. P. G. – « Coupling techniques for nonlinear hyperbolic equations. III. well-balanced approximation of thick interfaces », arXiv :1205.2437 (2012). (Cité en pages 50 et 51.)
- [14] A. R. BRODTKORB, T. R. HAGEN & M. L. SÆTRA – « Graphics processing unit (GPU) programming strategies and trends in {GPU} computing », Journal of Parallel and Distributed Computing **73** (2013), no. 1, p. 4–13, *Metaheuristics on {GPUs}*. (Cité en page 201.)
- [15] R. BÜRGER & K. H. KARLSEN – « Conservation laws with discontinuous flux : a short introduction », J. Engrg. Math. **60** (2008), no. 3-4, p. 241–247. (Cité en pages 50 et 51.)
- [16] R. BÜRGER, K. H. KARLSEN & N. H. RISEBRO – « A relaxation scheme for continuous sedimentation in ideal clarifier-thickener units », Comput. Math. Appl. **50** (2005), no. 7, p. 993–1009. (Cité en page 50.)
- [17] R. BÜRGER, R. RUIZ, K. SCHNEIDER & M. SEPÚLVEDA – « Fully adaptive multiresolution schemes for strongly degenerate parabolic equations with discontinuous flux. », J. Engrg. Math. **60** (2008), p. 365–385. (Cité en page 83.)
- [18] R. BÜRGER, K. H. KARLSEN & J. D. TOWERS – « A conservation law with discontinuous flux modeling traffic flow with abruptly changing road surface conditions », in Hyperbolic problems : theory, numerics and applications, Proc. Sympos. Appl. Math., vol. 67, Amer. Math. Soc., Providence, RI, 2009, p. 455–464. (Cité en page 50.)
- [19] G.-Q. CHEN, N. EVEN & C. KLINGENBERG – « Hyperbolic conservation laws with discontinuous fluxes and hydrodynamic limit for particle systems », J. Differ. Equations **245** (2008), no. 11, p. 3095 – 3126. (Cité en pages 50 et 51.)
- [20] G. CHIAVASSA, R. DONAT & A. MARTINEZ-GAVARA – « Cost-effective multi-resolutions schemes for shock computations », in Multiresolution and adaptive methods for convection-dominated problems, ESAIM Proc., vol. 29, EDP Sci., Les Ulis, 2009, p. 8–27. (Cité en page 42.)
- [21] F. CLÉMENT – « Optimal control of the cell dynamics in the granulosa of ovulatory follicles », Math. Bio. **152** (1998), p. 123–142. (Cité en pages 3 et 4.)
- [22] — , « Modélisation, simulation et commande de systèmes physiologiques : vers une approche multi-échelles », *Habilitation à diriger des recherches*, Université Pierre et Marie Curie, 2005. (Cité en page 3.)
- [23] — , « Multiscale modeling of endocrine systems : new insight on the gonadotrope axis », ESAIM : Proc. **27** (2009), p. 209–226. (Cité en pages 3 et 26.)
- [24] F. CLÉMENT, J.-M. CORON & P. SHANG – « Optimal control of cell mass and maturity in a model of follicular ovulation », SIAM J. Control Optim. **51** (2013), p. 824–847. (Cité en pages 4 et 26.)
- [25] F. CLÉMENT, N. ECHENIM & M. SORINE – « Multiscale modeling of follicular ovulation as a reachability problem », Multiscale Model. Simul. **6** (2007), p. 895–912. (Cité en page 102.)

## Bibliographie

---

- [26] F. CLÉMENT, M. GRUET, P. MONGET, M. TERQUI, E. JOLIVET & D. MONNIAUX – « Growth kinetics of the granulosa cell population in ovarian follicles : an approach by mathematical modeling », Cell Prolif. **30** (1997), p. 255–270. (Cité en pages 3 et 134.)
- [27] F. CLÉMENT & D. MONNIAUX – « Multiscale modeling of ovarian follicular selection. », Progress in Biophysics and Molecular Biology **113** (2013), p. 398–408. (Cité en pages 3 et 112.)
- [28] F. CLÉMENT, D. MONNIAUX, J. STARK, K. HARDY, J. THALABARD, S. FRANKS & D. CLAUDE – « A mathematical model of FSH-induced cAMP production in ovarian follicles. », Am J. Physiol. Endocrinol. Metab. **281** (2001), p. E35–E53. (Cité en pages 3 et 8.)
- [29] A. COHEN – Wavelet methods in Numerical Analysis, Handbook of Numerical Analysis, vol. VII, P.G. Ciarlet and J.L. Lions eds, Elsevier, Amsterdam, 2000. (Cité en page 173.)
- [30] A. COHEN, S. M. KABER, S. MÜLLER & M. POSTEL – « Fully adaptive multiresolution finite volume schemes for conservation laws », Math. Comp. **72** (2003), no. 241, p. 183–225. (Cité en pages 15, 18, 41, 82, 88, 93, 99 et 170.)
- [31] A. COHEN, S. KABER & M. POSTEL – « Adaptive multiresolution for finite volume solutions of gas dynamics », Computer and Fluids **32** (2003), p. 31–38. (Cité en page 82.)
- [32] P. COLELLA – « Multidimensional Upwind Methods for Hyperbolic Conservation Laws », J. Comput. Phys. **87** (1990), p. 171–200. (Cité en page 13.)
- [33] F. COQUEL, Y. MADAY, S. MÜLLER, M. POSTEL, & Q. TRAN – « Multiresolution and adaptive methods for convection-dominated problems », ESAIM : Proceedings **29** (2009). (Cité en page 82.)
- [34] F. COQUEL, Q. L. NGUYEN, M. POSTEL & Q. H. TRAN – « Entropy-satisfying relaxation method with large time-steps for Euler IBVPs », Math. Comp. **79** (2010), no. 271, p. 1493–1533. (Cité en pages 42 et 82.)
- [35] F. COQUEL, M. POSTEL & Q. TRAN – « Convergence of time-space adaptive algorithms for nonlinear conservation laws », IMA J. Numer. Anal. **32** (2012), p. 1440–1483. (Cité en page 93.)
- [36] F. COQUEL – « Theoretical and numerical analysis of hyperbolic systems of conservation laws », 2010. (Cité en page 170.)
- [37] T. DAVIES – « A new dynamical core for the Met Office’s global and regional modelling of the atmosphere », Quarterly Journal of the Royal Meteorological Society **131** (2005), p. 1759–1782. (Cité en page 201.)
- [38] J. DEMAILLY – Analyse Numérique et Equations Différentielles, Presses Universitaires de Grenoble (PUG), Grenoble, 1996. (Cité en page 180.)
- [39] N. ECHENIM – « Modélisation et contrôle multi-échelles du processus de sélection des follicules ovulatoires », Thèse, Université Paris-Sud XI, Faculté des Sciences d’Orsay, 2006. (Cité en pages 4, 13, 26, 36, 39, 52 et 164.)

## Bibliographie

---

- [40] N. ECHENIM, F. CLÉMENT & M. SORINE – « Multiscale modeling of follicular ovulation as a reachability problem », Multiscale Model. Simul. **6** (2007), no. 3, p. 895–912. (Cit  en pages 3, 26, 36, 37, 39, 163 et 166.)
- [41] N. ECHENIM, D. MONNIAUX, M. SORINE & F. CLÉMENT – « Multi-scale modeling of the follicle selection process in the ovary », Math. Biosci. **198** (2005), no. 1, p. 57–79. (Cit  en pages 3, 26, 51, 52, 69, 83, 84, 112, 163 et 166.)
- [42] Y. GAVRIELI, Y. SHERMAN & S. BEN-SASSON – « Identification of programmed cell death In Situ via specific labeling of nuclear DNA fragmentation », J. Cell Biol. **119** (1992), p. 493–501. (Cit  en page 135.)
- [43] E. GODLEWSKI & P.-A. RAVIART – « The numerical interface coupling of nonlinear hyperbolic systems of conservation laws. I. The scalar case », Numer. Math. **97** (2004), no. 1, p. 81–130. (Cit  en pages 31, 34 et 51.)
- [44] S. GOTTLIEB & C. W. SHU – « Total Variation Diminishing Runge Kutta schemes », Math. Comp. **67** (1998), no. 221, p. 73–85. (Cit  en page 55.)
- [45] A. HARTEN – « Multiresolution algorithms for the numerical solutions of hyperbolic conservation laws », Comm. Pure Appl. Math. **48** (1995), p. 1305–1342. (Cit  en pages 87, 88 et 170.)
- [46] D. HEITZLER, G. DURAND, N. GALLAY, A. RIZK, S. AHN, J. KIM, J. VIOLIN, L. DUPUY, C. GAUTHIER, V. PIKETTY, P. CRÉPIEUX, A. POUPON, F. CLÉMENT, F. FAGES, R. LEFKOWITZ & E. REITER – « Competing g protein-coupled receptor kinases balance g protein and beta arrestin signaling », Mol Syst Biol. **8** (2012), p. 590. (Cit  en page 202.)
- [47] C. HOMBOURGER – « Rapport de stage de 3<sup> me</sup> ann e. Mod elisation multi- chelle du d veloppement folliculaire », 2008. (Cit  en pages 4 et 13.)
- [48] N. HOVHANNISYAN & S. M LLER – « On the stability of fully adaptive multiscale schemes for conservation laws using approximate flux and source reconstruction strategies », IMA J. Numer. Anal. **30** (2010), no. 4, p. 1256–1295. (Cit  en page 42.)
- [49] P. JOLLY, P. SMITH, D. HEATH, N. HUDSON, S. LUN, L. STILL, C. WATTS & K. MCNATTY – « Morphological evidence of apoptosis and the prevalence of apoptotic versus mitotic cells in the membrana granulosa of ovarian follicles during spontaneous and induced atresia in ewes », Biol. Reprod. **56** (1997), p. 837–846. (Cit  en page 135.)
- [50] B. KOREN – « A robust upwind discretisation method for advection, diffusion and source terms », in Numerical Methods for Advection-Diffusion Problems, Braunschweig : Vieweg, 1993, p. 117. (Cit  en page 55.)
- [51] S. N. KRUŽKOV – « First order quasilinear equations with several independent variables. », Mat. Sb. (N.S.) **81 (123)** (1970), p. 228–255. (Cit  en page 50.)
- [52] T. LELI VRE – « M thodes num riques probabilistes », 2010. (Cit  en page 184.)

## Bibliographie

---

- [53] R. LEVEQUE – « Wave propagation algorithms for multidimensional hyperbolic systems », *J. Comput. Phys.* **131** (1997), no. 2, p. 327–353. (Cité en page 26.)
- [54] R. LEVEQUE & M. BERGER – « Adaptive mesh refinement using wave-propagation algorithms for hyperbolic systems », *SIAM J. Numer. Anal.* **35** (1998), no. 6, p. 2298–2316. (Cité en page 26.)
- [55] R. LEVEQUE & S. MITRAN – « Wave-propagation methods and software for complex applications », p. 93–104. (Cité en pages 4 et 13.)
- [56] V. LOUVET & M. MASSOT – « Summer school on multiresolution and adaptive mesh refinement methods », *ESAIM : Proceedings* **34** (2011). (Cité en page 82.)
- [57] M. LUNACEK, A. NAG, D. ALBER, K. GRUCHALLA, C. CHANG & P. GRAF – « Simulation, characterization, and optimization of metabolic models with the high performance systems biology toolkit », *SIAM J. Sci. Comput.* **33** (2011), p. 3402–3424. (Cité en page 202.)
- [58] C. MANSANET – « Contrôle génétique et physiologique de la prolificité en race ovine lacaune : caractérisation de la mutation causale et rôle fonctionnel du gène *FecL* », Thèse, Inra Tour Nouzilly, 2013. (Cité en page 136.)
- [59] M. MATSUMOTO & T. NISHIMURA – « Mersenne Twister : A 623-dimensionally equidistributed uniform pseudo-random number generator », *ACM Transactions on Modeling and Computer Simulation* **8** (1998), p. 3–30. (Cité en page 187.)
- [60] P. MICHEL – « Multiscale modeling of follicular ovulation as a mass and maturity dynamical system », *Multiscale Model. Simul.* **9** (2011), no. 1, p. 282–313. (Cité en pages 4, 26, 102, 127 et 143.)
- [61] D. MONNIAUX, A. CARATY, F. CLÉMENT, R. DALBIÈS-TRAN, J. DUPONT, S. FABRE, N. GÉRARD, P. MERMILLOD, P. MONGET & S. UZBEKOVA – « Développement folliculaire ovarien et ovulation chez les mammifères », *Inra Prod. Anim.* **22** (2009), p. 59–76. (Cité en page 3.)
- [62] S. MULLER – *Adaptive multiscale schemes for conservation laws*, Springer, Basel, 2003. (Cité en pages 82 et 170.)
- [63] S. PASSOT – « Rapport de stage de master. », 2013. (Cité en page 201.)
- [64] B. PERTHAME – *Transport equations in biology*, Birkhäuser Verlag, Basel, 2007. (Cité en pages 69, 73, 82, 84 et 110.)
- [65] C. PISSELET, F. CLÉMENT & D. MONNIAUX – « Fraction of proliferating cells in granulosa during terminal follicular development in high and low prolific sheep breeds », *Reprod. Nutr. Dev.* **40** (2000), p. 295–304. (Cité en page 135.)
- [66] J. P. RAVINDRA, N. C. RAWLINGS, A. C. O. EVANS & G. P. ADAMS – « Ultrasonographic study of ovarian follicular dynamics in ewes during the oestrous cycle », *J. Reprod. Fertil.* **101** (1994), no. 2, p. 501–509. (Cité en page 134.)
- [67] G. RODRIGUEZ – « Princeton University Lecture Notes », <http://data.princeton.edu/wws509/notes/c7s1.html>, 1994–2014. (Cité en page 20.)

## Bibliographie

---

- [68] F.-X. ROUX – « Analyse numérique matricielle avancée et calcul parallèle », 2010. (Cité en page 182.)
- [69] A. SAKAUE-SAWANO, H. KUROKAWA, T. MORIMURA, A. HANYU, H. HAMA, H. OSAWA, S. KASHIWAGI, K. FUKAMI, T. MIYATA, H. MIYOSHI, T. IMAMURA, M. OGAWA, H. MASAI & A. MIYAWAKI – « Visualizing spatiotemporal dynamics of multicellular cell-cycle progression », Cell **132** (2008), p. 487–498. (Cité en pages 105 et 141.)
- [70] N. SEGUIN & J. VOVELLE – « Analysis and approximation of a scalar conservation law with a flux function with discontinuous coefficients », Math. Models Methods Appl. Sci. **13** (2003), no. 02, p. 221–257. (Cité en page 52.)
- [71] P. SHANG – « Cauchy problem for multiscale conservation laws : Application to structured cell populations », J. Math. Anal. Appl. **401** (2013), p. 896–920. (Cité en pages 4, 7, 26, 32, 51, 52, 72, 73, 82 et 113.)
- [72] P. K. SWEBY – « High resolution schemes using flux limiters for hyperbolic conservation laws », SIAM J. Numer. Anal. **21** (1984), no. 5, p. 995–1011. (Cité en pages 35 et 55.)
- [73] M. TOMURA, A. SAKAUE-SAWANO, Y. MORI, M. TAKASE-UTSUGI, A. HATA, K. OHTAWA, O. KANAGAWA & A. MIYAWAKI – « Contrasting quiescent G0 phase with mitotic cell cycling in the mouse immune system », PLoS ONE **13** (2013). (Cité en page 105.)
- [74] J. TRYOEN – « Adaptive stochastic Galerkin methods for parametric uncertainty propagation in hyperbolic systems », Thèse, Ecole des Ponts ParisTech, 2013. (Cité en page 184.)
- [75] K. TURNBULL, A. BRADEN & P. MATTNER – « The pattern of follicular growth and atresia in the ovine ovary », Aust. J. Biol. Sci. **30** (1977), p. 229–241. (Cité en page 144.)
- [76] K. TURNBULL, P. MATTNER, J. GEORGE & R. SCARAMUZZI – « The relation between patterns of ovarian follicle growth and ovulation rate in sheep », Aust. J. Biol. Sci. **31** (1978), p. 649–655. (Cité en page 144.)
- [77] J. J. TYSON & B. NOVAK – « Temporal organization of cell cycle », Curr. Biol. **18** (2008), p. R759–768. (Cité en pages 51 et 70.)
- [78] T. VALLERON – Cours polycopié. (Cité en page 122.)