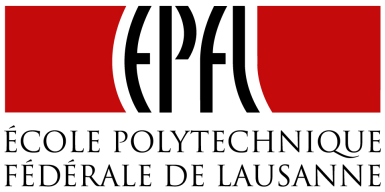

Chair of Modelling and Scientific Computing

Master in Applied Mathematics

Master Thesis
June 24, 2016

Inverse Problems in Tumor Growth Modelling

Léo DISERENS
N^o SCIPER : 217337
under the supervision of L.Dede'
and A.Manzoni

Abstract

In this project, we deepen the analysis of a tumour growth model, recently proposed by Garcke et al. in [1]. This model describes tumour and healthy cells evolution as well as tumour cells' nutrients, mixture velocity and pressure in the domain. Furthermore, it takes into account chemotaxis and apoptosis death of tumour cells, through a system of parabolic nonlinear PDE, that is a Cahn-Hilliard Darcy model, together with an advection-diffusion-reaction equation describing the evolution of nutrients. We perform a dimensional analysis and we build a numerical solver by use of the finite element method in space, a Backward Euler in time and a Newton method to tackle the nonlinearity. We perform several numerical simulations in order to recover results obtained in the article and to catch a general growth of the tumour depending on parameters of interest. Finally, a PDE-constrained optimization problem is formulated and solved, aiming at determining the shape of the tumour after a fixed time from an initial guess of its location. From the numerical simulations we obtained for the nutrients, we notice that the concentration of nutrients in an observable zone around the tumor region could possibly bring enough informations to achieve this goal. Therefore, a previous numerical simulation of nutrients will be taken as a target, in order to recover the controlled tumour function, previously simulated numerically. In this respect, preliminary numerical results show that, to some extents, it is possible to identify the general shape of the tumor, even if the exact result of the numerical simulation could not be recovered.

Acknowledgements

First, I would like to thank Dr. Luca Dede' and Dr. Andrea Manzoni for having allowed me to do this Master Thesis under their supervision. Their good advices, patience and kindness helped me a lot in performing this project. I would like to thank also Dr. Federico Negri and Dr. Davide Forti for having welcomed me in their office and for their very helpful advices and help.

I would like to thank all my mathematics friends that I have met during these five years at EPFL. Especially, they have made these tough years going well, and I'm grateful for that. Of course I am not forgetting all my other friends and the most precious ones : Harrys, thank you for being my friends.

Finally, I want to thank my precious family. Papa, Maman, Marius, merci pour votre support et votre amour sans faille.

Contents

Introduction

Cancer is nowadays a dangerous and often deadly disease, even if great improvements have been made so far. Since the past decade mathematicians started to obtain PDE models enabling to represent the growth of a tumour, taking into account several biologic behaviours, see e.g. [2], [1]. Among the differential models used by mathematicians, the class of phase field models is commonly accepted as a powerful tool to model the evolution of a tumor. The long-term goal in this respect would be the setting of a computational framework in cancer research, where the growth of a tumour could be simulated numerically and allowing to predict the adapted treatment to heal the patient. Nevertheless, a tumour, depending on its nature, its characteristics and also the body's immune system, can be growing in so many different manners that we are still far from having reached this goal. A tumour growth model, as we mentioned, shall potentially take into account different biological effects and characteristics of the tumour, but also of the healthy tissue interactions and mechanisms at the cellular scale. The chemotaxis effect is an example of biological effect, representing a growth of the tumour towards the largest amount of chemical species acting as tumour's nutrients. These effects highly depend on physical parameters of interest, enforcing or weakening the growth of the tumour or just modifying its diffusion.

In order to be able to predict the growth of a tumour, a major part is to determine those parameters which control the biologic effects. Experimentally, if the precise detection of a tumour is still a current research, observations can be made in the tumour cells' area of other body characteristics, e.g. an excess of H^+ ion concentration in the healthy tissue, see [3] or the concentration of some tumour nutrients around the tumour distribution, as previously mentioned. Ultimately, by relying on a tumour growth model and a set of experimental observation, we aim at determining the non-dimensional values of these parameters in the model, by solving a parameter identification (or estimation) problem. From another point of view, by considering an observation of nutrients in an observable zone around the tumor region, we could also determine if the shape of the tumor can be recovered numerically using an inverse problem, in a zone where the tumor cells cannot be detected with precision. While a direct problem consists in simulating a growth of a tumour with given parameters, an identification problem will seek either a set of parameters or an unknown function, called the control variable, by matching tumour growth or other simulated functions with a given observation or target. Hence, a control can be set, e.g. on the actual value of a parameter, governing a biological effect, like the chemotaxis, or on the initial and final distribution of the tumour, see e.g. [4].

In this project we consider the tumour growth model recently proposed by [1] and we approach numerically one of the system of parabolic PDEs governing the model. In this model nutrients are taken into account as well as the evolution of the tumor. Since

the precise detection of the tumor location is tough, the goal in this project will be to determine the evolution of the tumor from the configuration of nutrients consumed by the tumor.
In Chapter 1, we make a summary of the model development provided in [1], with a comparison of this model and the one provided in [2]. We also develop a dimensional analysis and prove the role of the nutrient flux in the mass growth of the tumor.
In Chapter 2 we develop the numerical approximation of the system of PDEs. First, we make the numerical approximation of the parabolic nonlinear Cahn-Allen equation, in order to introduce the numerical method we rely on in this work.
In Chapter 3, we show numerical results we obtained by simulating the system of equations, using the redbKIT package developed at CMCS ([5]). A mesh adaptive algorithm has been implemented in order to speed up simulation time and increase the mesh precision at the interface between tumor and healthy cells. This algorithm is presented with the numerical results we obtained by applying it to the simulations of the tumor growth model.
In Chapter 4 we formulate the problem of identifying the final shape of a tumor after a fixed time, targeting a previously simulated nutrient concentration in an observable domain, as an optimal control problem, showing the obtained results and drawing some conclusions on the potential of this approach.

Chapter 1

Tumor growth modelling by a Cahn-Hilliard Darcy model

In this section we introduce the main steps of the derivation of the tumor growth model we consider in this work. As an introduction of the phase field model, we briefly present the Cahn-Allen equation as a simpler model to describe phase separation. From the non-dimensional system of equations of the tumor growth model, we will be interested in determining the dimension of the parameters used in the model. Finally, we will show the role of the flux of nutrients across the boundary of the domain in the mass growth rate of the tumor.

Phase field models allow to solve interfacial problems, modelling the evolution of two heterogeneous mixtures, called phases, and describing their evolution in time. The system is described by a field representing the phase concentrations, taking two distinct values (e.g., -1 and 1) for each phase. This field usually shows a sharp but smooth slope in the interface region. The Cahn-Hilliard equation is an example of nonlinear phase field model. This parabolic fourth-order equation models spinodal decomposition process of phases, see e.g. [6] or [7]. The model we consider in this project is a Cahn-Hilliard Darcy model, coupled with a dynamic advection-diffusion-reaction equation for the simulation of the nutrients. A Darcy equation models the flow of a viscous fluid, described by the pressure gradient in a porous media.

1.1 The Cahn-Allen equation

The Cahn-Allen equation is a parabolic nonlinear partial differential equation, modelling the phase separation of two components of a mixture in isothermal situation. As the Cahn-Hilliard equation previously mentioned, the Cahn-Allen equation is a phase field model describing phase separation, with the difference that the solution is not mass conservative, see e.g. [8] or [9]. In this section we introduce this simpler phase field model and detail its numerical approximation in Chapter 2, in order to apply the same numerical scheme on the tumor growth model.

The solution $u(\vec{x}, t)$ of the Cahn-Allen equation represents the relative difference in volume of the components at a point $\vec{x} \in \Omega$ and at time $t \in (0, T)$. Let $\Omega \subset \mathbb{R}^n$ and $T > 0$. The equation reads as follow :

Find $u : \Omega \subset \mathbb{R}^n \times (0,T) \to \mathbb{R}$ such that

$$\begin{aligned} \frac{\partial u(t)}{\partial t} - \mu \Delta u(t) + \sigma(u(t)) &= 0 \quad \text{in } \Omega \times (0,T) \\ \mu \frac{\partial u}{\partial \nu} &= 0 \quad \text{on } \partial\Omega \times (0,T) \\ u &= u_D \quad \text{in } \Omega \times \{0\} \end{aligned} \tag{1.1}$$

The function $\sigma(u) = 2u(u-1)(2u-1)$ and μ is a positive coefficient. The solution of (1.1) aims at minimizing the energy functional

$$\hat{\Psi}(t) = \Psi(u(t)) = \int_\Omega F(u(t)) + \frac{1}{2}\mu |\nabla_\Omega u|^2 \, \mathrm{d}x, \tag{1.2}$$

with $F' = \sigma$. Indeed, from (1.1) we obtain

$$\begin{aligned} \frac{d}{dt}\hat{\Psi}(t) &= \int_\Omega \left[\sigma(u(t)) \frac{\partial u}{\partial t} + \mu \nabla u \cdot \nabla \frac{\partial u}{\partial t} \right] \, \mathrm{d}x \\ &= - \left|\left| \frac{\partial u}{\partial t} \right|\right|_{L^2(\Omega)} \leq 0. \end{aligned}$$

Now let $V = H^1(\Omega)$ and define the bilinear form

$$a(u,v) = \int_\Omega \mu \nabla u \cdot \nabla v \, \mathrm{d}x \tag{1.3}$$

and the nonlinear functional

$$c(u)(v) = \int_\Omega \sigma(u) v \, \mathrm{d}x, \tag{1.4}$$

for $u, v \in V$. Finally, let us define the residual

$$R(u(t))(v) = \left(\frac{\partial u(t)}{\partial t}, v \right)_\Omega + a(u(t), v) + c(u(t))(v) \ \forall v \in V \tag{1.5}$$

so that the weak formulation of the system (1.1) can be written as follows :

Find $u \in L^2((0,T); V)$ such that

$$\begin{aligned} &R(u(t))(v) = 0 \quad , \forall v \in V, \ t > 0 \\ &u(0) = u_0 \end{aligned} \tag{1.6}$$

1.2 Phase field model derivation

Mainly the theory of mixture is used in order to introduce the heterogeneous tumor and healthy cells in an isothermal system, their property and, by use of the balance laws, the equation linking them together. Furthermore an unspecified chemical species is computed as well, consumed only by tumor and allowing them to grow. The chemotaxis effect, representing an active growth of the tumor toward the largest amount of nutrient concentration, is then computed as well. In the case of nutrient concentration, we have no mass and flux restrictions. Finally a phase field model implies the potential energy of the system, described by the relative mass of the mixtures and their first derivative. In

this case it is introduced by respecting the second law of thermodynamics, stating that the entropy of an isolated system never stops increasing over time. This is described here by a upper limit of the rate of the total energy.

Let us consider two component mixtures consisting in the tumor and healthy cells and we denote their actual mass per volume by $\rho_i, i = 1, 2$ respectively. These functions take positive values in a bounded domain $\Omega \subset \mathbb{R}^n$. Let $\bar{\rho}_i, i = 1, 2$ be the mass density of the component i. The volume density of each components is defined as

$$u_i = \frac{\rho_i}{\bar{\rho}_i}, \ i = 1, 2. \tag{1.7}$$

These functions reveals the relative volume density of each component in the domain and hence we have $u_i : \Omega \to [0, 1]$. This implies $\rho_i : \Omega \to [0, \bar{\rho}_i], i = 1, 2$. Some other assumptions are made on the mixture :

- We assume the system to be isolated so that there is no other kind of mixture in the domain, implying $u_1 + u_2 = 1$.
- Mass transfer between the components is taken into account in this model. Healthy cells are transformed into tumor cells when these latter are growing, and tumor cells are transformed into healthy cells when tumor cells die by apoptosis. Hence, the total mass of the components, defined as

  $$\rho(t) = \rho_1(t) + \rho_2(t)$$

  may vary over time.

Nutrients are modelled as chemical species and their relative concentration inside the domain is denoted by $\sigma : \Omega \to [0, 1]$. We define the dimensionless velocity of each component i by $\mathbf{v}_i$. The volume-averaged velocity of the mixture is defined by

$$\mathbf{v} := u_1\mathbf{v}_1 + u_2\mathbf{v}_2. \tag{1.8}$$

From the theory of mixture, the balance law equation equals the mass rate of change of a component and the mass transfer from other components to this one, see e.g. [10]. The equation reads

$$\frac{\partial \rho_i}{\partial t} + \nabla \cdot (\rho_i \mathbf{v}_i) = \Gamma_i, \ \text{for } i = 1, 2, \tag{1.9}$$

being $\Gamma_i, \ i = 1, 2$ a source term for each component describing the mass exchange from the component $3 - i$ to the component i for $i = 1, 2$. The balance law equation for the nutrients is postulated in [1] and is given by

$$\frac{\partial \sigma}{\partial t} + \nabla \cdot (\sigma \mathbf{v}) + \nabla \cdot \mathbf{J}_\sigma = -\mathcal{S}, \tag{1.10}$$

where $\mathbf{J}_\sigma$ is defined by the flux of the nutrients and allows to model the diffusion of the nutrients and the chemotaxis effect. This latter effect defines the attraction of the tumor cells for the nutrients and vice-versa. Finally, the function $\mathcal{S}$ defines a source term for the nutrients. From (1.9), taking the divergence of the equation (1.8) yields

$$\nabla \cdot \mathbf{v} = \frac{\Gamma_2}{\bar{\rho}_2} + \frac{\Gamma_1}{\bar{\rho}_1} =: \Gamma_{\mathbf{v}}. \tag{1.11}$$

Now, in order to model both healthy and tumor cells, let us define the unknown function φ as the difference in volume fraction of both components ;

$$\varphi := u_2 - u_1$$

such that $\varphi(x) \in [-1, 1]$. It follows that $\varphi(x) = 1$ for a point lying in the tumor region in Ω, and $\varphi = -1$ for a point lying in the healthy cells region. Applying the definition of φ and the definition (1.7) to the equation (1.9), we obtain the balance law equation for the difference in volume fraction of both phases

$$\frac{\partial \varphi}{\partial t} + \nabla \cdot (\varphi \mathbf{v}) + \nabla \cdot \mathbf{J} = \frac{\Gamma_2}{\bar{\rho}_2} - \frac{\Gamma_1}{\bar{\rho}_1} =: \Gamma_\varphi, \tag{1.12}$$

where $\mathbf{J} = -u_1(\mathbf{v}_1 - \mathbf{v}) + u_2(\mathbf{v}_2 - \mathbf{v})$ is defined as the flux for the functino φ.

In a phase field model, the system aims at minimizing the energy emerging from the interaction between the phases and from the interface. Together with the balance laws for the mass of tumor and healthy cells, let us introduce the energy density of the system of the following form

$$e(\varphi, \nabla\varphi, \sigma) = f(\varphi, \nabla\varphi) + N(\varphi, \sigma). \tag{1.13}$$

The function f determines the interfacial energy, that is, it controls the sharp interface of the function φ defining the separation between the components and is defined as

$$f(\varphi, \nabla\varphi) = \beta \left(\frac{1}{\varepsilon} \Psi(\varphi) + \frac{\varepsilon}{2} |\nabla\varphi|^2 \right), \tag{1.14}$$

for $\beta, \varepsilon > 0$, ε a very small term directly proportional to the width of the sharp interface. The function $\Psi : \mathbb{R} \to \mathbb{R}$ defines the potential of the system with equal minima $\Psi(\pm 1) = 0$, ideally forcing the mixture to separate into phases, i.e. $|\varphi(x)| = 1$ for $x \in \Omega$ except in the sharp interface subdomain. We use in this project the so-called double-well potential

$$\Psi(\varphi) = \frac{1}{4}(1 - \varphi^2)^2. \tag{1.15}$$

Finally, the function $N(\varphi, \sigma)$ in (1.13) models the energy emerging from the interactions between the tumor tissues and the nutrients. Also, this term controls the chemotaxis behaviour of the cells. We will use the notation $N_{,\sigma}$ and $N_{,\varphi}$ to denote the derivatives of N with respect to σ and φ respectively.

In order to derive the diffuse interface model, the second law of thermodynamics in isothermal situations is applied. This law stipulates that the entropy of the system increases until having reached the upper bound. This law is applied to the model in term of system's energy density, stating that the energy rate plus the useful work on any part of the system must be less or equal than 0. For any volume of mixture $V(t) \subseteq \Omega$ moving by the fluid velocity we have the following inequality

$$\frac{d}{dt} \int_{V(t)} e \, \mathrm{d}x \leq - \int_{\partial V(t)} \mathbf{J}_e \cdot \nu \, \mathrm{d}S + \int_{V(t)} c_\varphi \Gamma_\varphi + c_{\mathbf{v}} \Gamma_{\mathbf{v}} - c_S \mathcal{S} \, \mathrm{d}x. \tag{1.16}$$

The term $\mathbf{J}_e$ is an energy flux, $\mathcal{S}$ is the source term for the nutrients, $c_\varphi, c_{\mathbf{v}}$ and $c_\mathcal{S}$ are constants and $\Gamma_{\mathbf{v}}$, Γ_φ are defined in (1.11) and (1.12) respectively. Using the divergence

theorem and the transport theorem, see e.g. [11], the equation (1.16) in a local form yields

$$\frac{\partial e}{\partial t} + \nabla \cdot (e\mathbf{v}) + \nabla \cdot \mathbf{J}_e - c_\varphi \Gamma_\varphi - c_{\mathbf{v}} \Gamma_{\mathbf{v}} + c_S \mathcal{S} \leq 0 \tag{1.17}$$

At this point in the development of the model, a Lagrange multiplier method is used to solve the inequality (1.17) under the constraints governed by the equations (1.10), (1.11) and (1.12). Given the Lagrange multiplier functions $\lambda_{\mathbf{v}}$, λ_σ and λ_φ, the following inequality must hold

$$\begin{aligned} -\mathcal{D} := & \frac{\partial e}{\partial t} + \nabla \cdot (e\mathbf{v}) + \nabla \cdot \mathbf{J}_e - c_\varphi \Gamma_\varphi - c_{\mathbf{v}} \Gamma_{\mathbf{v}} + c_S \mathcal{S} \\ & - \lambda_{\mathbf{v}} \left(\nabla \cdot \mathbf{v} - \Gamma_{\mathbf{v}} \right) \\ & - \lambda_\sigma \left(\frac{\partial \sigma}{\partial t} + \nabla \cdot (\sigma \mathbf{v}) + \nabla \cdot \mathbf{J}_\sigma + \mathcal{S} \right) \\ & - \lambda_\varphi \left(\frac{\partial \varphi}{\partial t} + \nabla \cdot (\varphi \mathbf{v}) + \nabla \cdot \mathbf{J} - \Gamma_\varphi \right) \leq 0. \end{aligned} \tag{1.18}$$

Finally, constitutive assumptions are made on the unknown model's parameters, in order to respect the Lagrangian inequality (1.18), i.e. we postulate explicit definitions of the fluxes $\mathbf{J}, \mathbf{J}_\sigma$ and $\mathbf{J}_e$, the energy constants $c_\varphi, c_{\mathbf{v}}$ and $c_{\mathcal{S}}$ and the Lagrange multipliers $\lambda_\varphi, \lambda_{\mathbf{v}}$ and λ_σ. The tumor growth model suggested by [1], arising from these last assumptions, is the following, endowed with homogeneous Neumann boundary conditions for the unknown function φ and the chemical potential function μ

$$\begin{aligned} \nabla \cdot \mathbf{v} &= \Gamma_{\mathbf{v}} && \text{in } \Omega \times (0,T] \\ \mathbf{v} &= -K(\nabla p + (N_{,\varphi} - \mu)\nabla \varphi) && \text{in } \Omega \times (0,T] \\ \tfrac{\partial \varphi}{\partial t} + \nabla \cdot (\varphi \mathbf{v}) &= \nabla \cdot (m(\varphi)\nabla \mu) + \Gamma_\varphi && \text{in } \Omega \times (0,T] \\ \mu &= \frac{\beta}{\varepsilon} \Psi'(\varphi) - \beta \varepsilon \Delta \varphi + N_{,\varphi} && \text{in } \Omega \times (0,T] \\ \tfrac{\partial \sigma}{\partial t} + \nabla \cdot (\sigma \mathbf{v}) &= \nabla \cdot (n(\varphi)\nabla N_{,\sigma}) - \mathcal{S} && \text{in } \Omega \times (0,T] \\ \nabla \varphi \cdot \nu &= \nabla \mu \cdot \nu = 0 && \text{on } \partial\Omega \times (0,T]. \end{aligned} \tag{1.19}$$

Different suitable boundary conditions for the nutrients can be applied to this model, depending on the importance and the behaviour of the nutrients. Typically, a Robin type of boundary condition is treated as the general case :

$$(n(\varphi)\nabla N_{,\sigma}) \cdot \nu = c(\sigma_\infty - \sigma), \text{ on } \partial\Omega \times (0,T),$$

where c, σ_∞ are positive constants. Different boundary conditions can be used for the pressure and the mixture velocity.

In the article we followed, different specific models are suggested, depending on the assumptions made on the model, e.g. assuming no excess of total mass or the absence of nutrients. The specific model we will work on in this project is the one used for the numerical result in [1], in order to test with our own code for similar results. This model assumed no excess of total mass, meaning that the quantity ρ is constant over time. In particular, it means that the source terms $\Gamma_i, i = 1, 2$ cancel out in (1.9). Mathematically, we obtain $\Gamma_2 = -\Gamma_1 = \Gamma$. Hence, we can redefine

$$\Gamma_{\mathbf{v}} = \alpha\Gamma, \ \Gamma_\varphi = \bar{\rho}_S \Gamma,$$

with $\alpha = \frac{1}{\bar{\rho}_2} - \frac{1}{\bar{\rho}_1}$ and $\bar{\rho}_S = \frac{1}{\bar{\rho}_2} + \frac{1}{\bar{\rho}_1}$. We also assume that the densities are equal and we set $\mathbf{v} = \vec{0}$. From those specifications, we obtain the following simplified system form

$$\begin{array}{rcll} \frac{\partial \varphi}{\partial t} & = & \nabla \cdot (m(\varphi)\nabla\mu) + \bar{\rho}_S \Gamma & \text{in } \Omega \times (0,T] \\ \mu & = & \frac{\beta}{\varepsilon}\Psi'(\varphi) - \beta\varepsilon\Delta\varphi + N_{,\varphi} & \text{in } \Omega \times (0,T] \\ \frac{\partial \sigma}{\partial t} & = & \nabla \cdot (n(\varphi)\nabla N_{,\varphi}) - \mathcal{S} & \text{in } \Omega \times (0,T] \\ \nabla\varphi \cdot \nu & = & \nabla\mu \cdot \nu = 0 & \text{on } \partial\Omega \times (0,T] \\ \sigma & = & \sigma_B & \text{on } \partial\Omega \times (0,T], \end{array} \tag{1.20}$$

where Dirichlet boundary conditions have been chosen for the nutrients.

From the model (1.20), we choose particular definitions for the source terms Γ and $\mathcal{S}$, the energy term $N(\varphi, \sigma)$ and the mobility tensor $m(\varphi), n(\varphi)$, as given in [1]. First, quasi-steady state of the nutrients is considered, that is, $\partial_t \sigma = 0$. Then, the following definitions are set :

- $\Gamma = (\mathcal{P}\sigma - \mathcal{A})h(\varphi)$;

  This function represents the reaction or source term for the equation of φ, modelling the propagation rate of the tumour cells. The function $h(\varphi)$ is equal to 0 in healthy region and 1 in tumor region in order to simulate propagation and nutrient consumption uniquely in the tumor cells region. It is defined as $h(\varphi) = \frac{1}{2}(1+\varphi)$. The term $\mathcal{P}$ defines the proliferation rate and $\mathcal{A}$ the apoptosis rate of the tumor cells, which is defined as the natural programmed death of the cells. The tumor necrosis has not been considered in this model. In tumor cells regions, the source term is equal to $(\mathcal{P}\sigma - \mathcal{A})$, which is defined as the growth of tumor towards the nutrient concentration, minus the apoptosis death of these cells.

- $N(\varphi, \sigma) = \dfrac{\chi_\sigma}{2}\sigma^2 + \chi_\varphi \sigma(1-\varphi)$;

  This term defines the energy provided by the interaction of the tumour cells and the nutrients. Specifically, the derivative of $N(\varphi, \sigma)$ with respect to φ controls the chemotaxis effect, that is a growth effect of the nutrients towards the tumor cells. This choice of the function $N(\varphi, \sigma)$, allows to obtain a significant value of the diffusion for the nutrient, i.e. we obtain $N_{,\sigma} = \chi_\sigma \sigma + \chi_\varphi(1-\varphi) \geq 0$. Hence, in the tumor region, where $\varphi = 1$, only diffusion takes place, but in the healthy cells region we also have a chemotactic behaviour of the nutrients.

- $\mathcal{S} = \mathcal{C}\sigma h(\varphi)$;

  This function represents the source term for the nutrients. It is equal to 0 in the domain $\{x \in \Omega : \varphi(x) = -1\}$ and is equal to $\mathcal{C}\sigma$ in the domain $\{x \in \Omega : \varphi(x) = 1\}$, representing an absorption of the nutrients by the tumor cells.

- $m(\varphi) = \dfrac{1}{2}(1+\varphi)^2$, $n(\varphi) = \lambda\chi_\varphi^{-1}\mathcal{D}(\varphi)$, together with $\lambda = \dfrac{\chi_\varphi}{\chi_\sigma}$ and

$$D(\varphi) = \frac{1+\varphi}{2} + \mathcal{D}\frac{1-\varphi}{2};$$

The functions m, n represent the non-negative mobilities of the tumor-healthy cells and the nutrient cells respectively. A non-linear mobility term for the function φ allows to obtain more movements from the tumor. Furthermore, the healthy cells remain in a steady state, since $m(\varphi) = 0$ in the domain $\{x \in \Omega : \varphi(x) = -1\}$.

The following system of equations is the one we treat in this project

$$\begin{aligned}
\tfrac{\partial \varphi}{\partial t} &= \nabla \cdot (\tfrac{1}{2}(1+\varphi)^2 \nabla \mu) + (\mathcal{P}\sigma - \mathcal{A})(\varphi + 1) && \text{in } \Omega \times (0,T] \\
\mu &= \tfrac{\beta}{\varepsilon}\Psi'(\varphi) - \beta\varepsilon\Delta\varphi - \chi_\varphi \sigma && \text{in } \Omega \times (0,T] \\
0 &= \nabla \cdot (D(\varphi)\nabla\sigma) - \lambda \nabla \cdot (D(\varphi)\nabla\varphi) - \tfrac{1}{2}\mathcal{C}\sigma(\varphi+1) && \text{in } \Omega \times (0,T] \\
\nabla\varphi \cdot \nu &= \nabla\mu \cdot \nu = 0 && \text{on } \partial\Omega \times (0,T] \\
\sigma &= \sigma_B && \text{on } \partial\Omega \times (0,T]
\end{aligned} \tag{1.21}$$

For a different definition of the last terms, the model can become quite different. In [1], the authors suggest to use the following functions

- $\Gamma = \frac{1}{2}P(\varphi)(N_{,\sigma} - \mu)$
- $\mathcal{S} = P(\varphi)(N_{,\sigma} - \mu)$
- $P(\varphi) = \begin{cases} \delta P_0(1+\varphi), & \text{if } \varphi \leq -1 \\ 0, & \text{otherwise.} \end{cases}$

The function $P(\varphi)$ takes a similar role of the function $h(\varphi)$ and influences the propagation of the tumor cells from the parameter P_0. With such terms, we would obtain the following model

$$\begin{aligned}
\tfrac{\partial \varphi}{\partial t} &= \nabla \cdot (m(\varphi)\nabla\mu) + \delta P_0(\varphi+1)(\chi_\sigma\sigma + \chi_\varphi(1-\varphi) - \mu) \\
\mu &= \tfrac{\beta}{\varepsilon}\Psi'(\varphi) - \beta\varepsilon\Delta\varphi - \chi_\varphi\sigma \\
\tfrac{\partial \sigma}{\partial t} &= \nabla \cdot (n(\varphi)(\chi_\sigma\nabla\sigma - \chi_\varphi\nabla\varphi)) - \delta P_0(\varphi+1)(\chi_\sigma\sigma + \chi_\varphi(1-\varphi) - \mu) \\
\nabla\varphi \cdot \nu &= \nabla\mu \cdot \nu = 0 \\
\sigma &= \sigma_B
\end{aligned} \tag{1.22}$$

We notice that an important change is made on the source terms in both equations of φ and σ, making it also depends on the chemical potential μ. This tumor growth model is the same as the one developed by Hawkins-Daarud et al. in [2].

Now let us develop the weak formulation of the system of equations (1.21), in order to approximate it numerically in chapter 2. Let $V = H^1(\Omega)$, $V_B = \{v \in H^1(\Omega) : v|_{\partial\Omega} = \sigma_B\}$ and $V_0 = H_0^1(\Omega)$. The weak formulation of the problem (1.21) is given by

Find $\varphi(t), \mu(t) \in V$ and $\sigma(t) \in V_B$ $\forall t \in (0,T]$ such that :

$$\begin{aligned}
\left(\frac{\partial\varphi}{\partial t}, v\right) + \left(\frac{1}{2}(1+\varphi)^2\nabla\mu, \nabla v\right) - ((\mathcal{P}\sigma - \mathcal{A})(\varphi+1), v) &= 0, \quad \forall v \in V \\
(\mu, \xi) + \frac{\beta}{\varepsilon}\left((1-\varphi^2)\varphi, \xi\right) - \beta\varepsilon\left(\nabla\varphi, \nabla\xi\right) + \left(\chi_\varphi\sigma, \xi\right) &= 0, \quad \forall \xi \in V \\
\left(D(\varphi)\nabla\sigma, \nabla w\right) - \lambda\left(D(\varphi)\nabla\varphi, \nabla w\right) + \frac{1}{2}\mathcal{C}\left(\sigma(\varphi+1), w\right) &= 0, \quad \forall w \in V_B
\end{aligned} \tag{1.23}$$

1.3 Dimensional Analysis of the model (1.21)

Let us perform a dimensional analysis of the system of equations (1.21). In order to obtain the most meaningful interpretation of the parameter, we keep $\chi_\sigma = \lambda\chi_\varphi$ like developed in the previous section. Since φ, σ denote the concentration functions, they are dimensionless quantities, so that $[\varphi] = [\sigma] = \emptyset$. We set the following variables

$$x = x^\star L_0, \quad t = t^\star T_0, \quad \mu = \mu^\star \mu_0, \tag{1.24}$$

where $x^\star, t^\star, \mu^\star$ are dimensionless variables taking value in $[0,1]$ and L_0, T_0 and μ_0 are the dimensional values of the corresponding variables. Expressing the system of equations (1.21) using (1.24) yields

$$\begin{array}{rcl} \frac{1}{T_0}\frac{\partial^\star \varphi}{\partial t^\star} & = & \frac{1}{L_0}\nabla^\star \cdot (\frac{\mu_0}{L_0}\frac{1}{2}(1+\varphi)^2\nabla^\star\mu^\star) + \mathcal{P}\sigma(\varphi+1) - \mathcal{A}\sigma(\varphi+1) \\ \mu_0\mu^\star & = & \frac{\beta}{\varepsilon}\Psi'(\varphi) - \frac{1}{L_0^2}\beta\varepsilon\Delta^\star\varphi - \chi_\varphi\sigma \\ 0 & = & \frac{\chi_\sigma}{L_0}\nabla^\star\cdot(\frac{\mathcal{D}}{L_0}\frac{1-\varphi}{2}\nabla^\star\sigma) - \frac{\chi_\varphi}{L_0}\nabla^\star\cdot(\frac{\mathcal{D}}{L_0}\frac{1-\varphi}{2}\nabla^\star\varphi) - \frac{\mathcal{C}\chi_\sigma}{2}\sigma(\varphi+1) \\ & + & \frac{\chi_\sigma}{L_0}\nabla^\star\cdot(\frac{1}{L_0}\frac{1+\varphi}{2}\nabla^\star\sigma) - \frac{\chi_\varphi}{L_0}\nabla^\star\cdot(\frac{1}{L_0}\frac{1+\varphi}{2}\nabla^\star\varphi). \end{array} \tag{1.25}$$

Grouping together dimensional parameters allows to obtain the following system

$$\begin{array}{rcl} N_1\frac{\partial^\star\varphi}{\partial t^\star} & = & \nabla^\star\cdot(\frac{1}{2}(1+\varphi)^2\nabla^\star\mu^\star) + N_2\sigma(\varphi+1) - N_3\sigma(\varphi+1) \\ \mu^\star & = & N_4\Psi'(\varphi) - N_5\Delta^\star\varphi - N_6\sigma \\ 0 & = & N_{7,H}\nabla^\star\cdot(\frac{1-\varphi}{2}\nabla^\star\sigma) - N_{8,H}\nabla^\star\cdot(\frac{1-\varphi}{2}\nabla^\star\varphi) - N_9\sigma\frac{1}{2}(\varphi+1) \\ & + & N_{7,T}\nabla^\star\cdot(\frac{1+\varphi}{2}\nabla^\star\sigma) - N_{8,T}\nabla^\star\cdot(\frac{1+\varphi}{2}\nabla^\star\varphi), \end{array} \tag{1.26}$$

where the dimensional constants are

$$\begin{array}{llll} N_1 = \frac{L_0^2}{T_0\mu_0}, & N_4 = \frac{\beta}{\varepsilon\mu_0}, & N_{7,H} = \chi_\sigma\mathcal{D}, & N_{7,T} = \chi_\sigma \\ N_2 = \frac{\mathcal{P}L_0^2}{\mu_0}, & N_5 = \frac{\beta\varepsilon}{\mu_0 L_0^2}, & N_{8,H} = \chi_\varphi\mathcal{D}, & N_{8,T} = \chi_\varphi \\ N_3 = \frac{\mathcal{A}L_0^2}{\mu_0}, & N_6 = \frac{\chi_\varphi}{\mu_0}, & N_9 = \mathcal{C}L_0^2. & \end{array} \tag{1.27}$$

In order to limit the number of parameters, we set $N_1 = N_4 := 1$. From these last equalities and the dimensional constants (1.27), we obtain the following dimensional quantities

$$\mu_0 = \frac{\beta}{\varepsilon}, \qquad T_0 = \frac{L_0^2}{\mu_0} = \frac{\varepsilon L_0^2}{\beta}. \tag{1.28}$$

Adding (1.28) for N_5, we finally obtain

$$N_5 = \frac{\varepsilon^2}{L_0^2} \qquad N_6 = \frac{\chi_\varphi\varepsilon}{\beta} = \frac{\chi_\varphi L_0^2}{T_0}. \tag{1.29}$$

This result shows that the parameter ε has the dimension of a length and proportional to the size of the domain. Furthermore, we obtain that T_0 is proportional to L_0^3, which means that the simulation time should be strongly increased if the domain becomes longer. In (1.29) we also observe that $\chi_\varphi = T_0 L_0^{-2}\chi_\varphi^\star$, where $\chi_\varphi^\star$ is the non-dimensional value of χ_φ. We can also remark that a higher value of β will decrease the characteristic time T_0, which implies that the parameter β is directly linked to the growth's speed of the tumor.

From (1.28) we also obtain

$$N_2 = \frac{\mathcal{P}\varepsilon L_0^2}{\beta} = \mathcal{P}T_0 \qquad N_3 = \frac{\mathcal{A}\varepsilon L_0^2}{\beta} = \mathcal{A}T_0 \quad \text{and so} \quad \frac{N_2}{N_3} = \frac{\mathcal{P}}{\mathcal{A}}. \tag{1.30}$$

We observe first that the dimension of these parameters are $\mathcal{P} = T_0^{-1}\mathcal{P}^\star$ and $\mathcal{A} = T_0^{-1}\mathcal{A}^\star$, where $\mathcal{P}^\star$, $\mathcal{A}^\star$ are the non-dimensional values of $\mathcal{A}$ and $\mathcal{P}$ respectively. By the definition of these parameters, we obtain that the tumor is supposed to grow towards nutrients if the ratio $\frac{N_2}{N_3}$ is large enough. Indeed, if we take the first equation in (1.21) in the subdomain $\{x \in \Omega : \varphi(x) = 1\}$, we obtain that

$$\frac{\partial \varphi}{\partial t} = 2\chi_\varphi \Delta\sigma + 2(\mathcal{P}\sigma - \mathcal{A}).$$

This implies that the growth of the tumor in the boundary of the tumor region will highly depend on the curvature of the nutrients in this region plus a reaction term given by $2(\mathcal{P}\sigma - \mathcal{A})$. Hence, if this latter term is negative, it will act as a sink for the tumor, so that the tumor-cells will shrink. This ratio has to be larger than the nutrient proportion in the tumor region, in order to imply growth of the tumor. In the case $\mathcal{A} = 0$, this ratio goes to infinity and $N_3 = 0$. This does not affect the other dimensional term. By (1.29) we obtain that $\mathcal{D} = L_0^2 T_0^{-1}\mathcal{D}^\star$, where $\mathcal{D}^\star$ is the non-dimensional value of $\mathcal{D}$ and χ_σ has the same dimensional value as χ_φ.

1.4 Role of the nutrient flux

In this section we show that the mass of the tumor can be obtained by measuring the integral of the flux on the boundary of the domain.

For $h \in C^2(\mathbb{R})$ such that $h(1) = 1$ and $h(-1) = 0$, we define the total mass of the tumor by

$$M_{\text{tum}}(t) = \int_\Omega h(\varphi(t)) \, \mathrm{d}\vec{x}, \tag{1.31}$$

together with the total mass of the nutrients

$$M_{\text{nut}}(t) = \int_\Omega \sigma(t) \, \mathrm{d}\vec{x} \tag{1.32}$$

and the total flux of the nutrient over the outer boundary

$$\mathcal{F}_{\text{nut}}(t) = \int_{\partial\Omega} D(\varphi(t)) \nabla\sigma(t) \cdot \nu \, \mathrm{d}\gamma. \tag{1.33}$$

Now from the system of equation (1.21) and the boundary conditions we have considered, we obtain the following :

$$\begin{aligned}\frac{d}{dt} M_{\text{tum}}(t) &= \int_\Omega h'(\varphi(t)) \frac{\partial \varphi}{\partial t} \, \mathrm{d}\vec{x} \\ &= -\int_\Omega \frac{1}{2} h''(\varphi(t))(1+\varphi)^2 \nabla\mu(t) \cdot \nabla\varphi(t) \, \mathrm{d}\vec{x} + \int_\Omega 2(\mathcal{P}\sigma(t) - \mathcal{A}) h(\varphi(t)) h'(\varphi(t)) \, \mathrm{d}\vec{x}\end{aligned}$$

Now by integrating the third equation of (1.21) over the domain we obtain

$$\begin{aligned}0 &= \oint_{\partial\Omega} D(\varphi(t)) \nabla\sigma(t) \cdot \nu \, \mathrm{d}\gamma - \lambda \oint_{\partial\Omega} D(\varphi(t)) \nabla\varphi \cdot \nu \, \mathrm{d}\gamma - \int_\Omega \mathcal{C}\sigma(t) h(\varphi(t)) \, \mathrm{d}\vec{x} \\ &= \mathcal{F}_{\text{nut}}(t) - \int_\Omega \mathcal{C}\sigma h(\varphi(t)) \, \mathrm{d}\vec{x}.\end{aligned}$$

Considering $h(\varphi) = \frac{1}{2}(1+\varphi)$, as we defined in section 1.1, we obtain the following ordinary differential equation from the two last equations

$$\frac{d}{dt}M_{\text{tum}}(t) = \frac{\mathcal{P}}{\mathcal{C}}\mathcal{F}_{\text{nut}}(t) - \mathcal{A}M_{\text{tum}}(t),\ \forall t \in (0,T) \tag{1.34}$$

The analytical solution of (1.34) is given by

$$M_{\text{tum}}(t) = \exp(-\mathcal{A}t)\left(\int_0^t \frac{\mathcal{P}}{\mathcal{C}}\exp(\mathcal{A}s)\mathcal{F}_{\text{nut}}(s)\ \mathrm{d}s + \int_\Omega h(\varphi_0)\ \mathrm{d}\vec{x}\right)\ \forall t \in (0,T). \tag{1.35}$$

In the case we assume tumour cells do not die by apoptosis, that is $\mathcal{A} = 0$, we obtain

$$M_{\text{tum}}(t) = \frac{\mathcal{P}}{\mathcal{C}}\int_0^t \mathcal{F}_{\text{nut}}(s)\ \mathrm{d}s + \int_\Omega h(\varphi_0)\ \mathrm{d}\vec{x},\ \forall t \in (0,T). \tag{1.36}$$

From (1.34), we see that the growth rate of the tumor cells is directly proportional to the ratio $\frac{\mathcal{P}}{\mathcal{C}}$. This last result seems reasonable according to the role of these two parameters in the mass transition between tumor and healthy cells and the consumption of nutrients by the tumor cells. Indeed, a bigger value of $\mathcal{P}$ will increase the proliferation of the tumor to the nutrients whereas a bigger value of $\mathcal{C}$ will increase the consumption of the nutrients in the tumor-cells regions, thus inducing a decrease of the volume of the tumor region. Even more important is the direct role of the nutrient flux in the growth rate of the tumor. In Chapter 3, we will show results confirming the observations we have made so far.

Chapter 2

Numerical Approximation

In this chapter we develop suitable techniques for the numerical approximation of the system of equations (1.21). First we detail the numerical approximation of the Cahn-Allen equation, in order to present the algorithm which will be used for the more sophisticated tumor growth' system of equations. For both these equations, we will use a finite element method to semi-discretize the problem in space; a Backward Euler method (or, more generally speaking, a θ-method) will then be applied for the time discretization. Finally, we will explain in details how to apply the Newton algorithm for the sake of tackling the nonlinearities.

2.1 Numerical approximation of Cahn-Allen equation

First we describe the numerical approximation of the Cahn-Allen equation, which is instrumental to set up the numerical approximation of the tumour growth model. We approximate numerically the solution of the weak formulation of the Cahn-Allen equation described in (2.2), using the finite element method for the semi-discretization in space, the θ-method for the time discretization and finally we use the Newton method in order to tackle the nonlinearity. For further details in the numerical approximation of the Cahn-Allen equations, see [9]. Let $h > 0$ and let $\mathcal{T}_h = \{K\}_{j=1}^{N_h}$ be a triangulation of Ω, for $N_h > 0$. We introduce $V_h = X_h^1 \cap V$ the finite element space with $\{\psi_j\}_{j=1}^{N_h}$ the finite element basis, where $X_h^1 = \{v \in C^0(\Omega) : v|_K \in \mathbb{P}_1,\ \forall K \in \mathcal{T}_\langle\}$. Hence, for $t > 0$ and $u_h(t) \in V_h$ we express

$$u_h(t) = \sum_{j=1}^{N_h} u_j(t)\psi_j. \tag{2.1}$$

The weak formulation for the semi-discretized problem in space reads :

$$\begin{aligned} &\text{Find } u_h(t) \in V_h, \forall t \in (0,T) \text{ such that} \\ &R_h(u_h(t))(v_h) = \left(\frac{\partial u_h(t)}{\partial t}, v_h\right)_\Omega + a(u_h(t), v_h) + c(u_h(t))(v_h) = 0 \quad \forall v_h \in V_h \\ &u_h(0) = \Pi_h u_0, \end{aligned} \tag{2.2}$$

where Π_h is the L^2 projection onto V_h and the residual $R_h(u) : V_h \to \mathbb{R}$ is the restriction of (1.5) on V_h.

input : $\mathcal{T}_h$, δ_t,T, $\Pi_h u_0$, R_θ and its Gateaux derivative, *tol*
output: Numerical solutions u_h^k in V_h for $k = 0, \ldots, N_t$

$u_h^0 \leftarrow \Pi_h u_0$;
for $k \leftarrow 0$ **to** $N_t - 1$ **do**
  $u_{h,0}^{k+1} \leftarrow u_h^k$;
  $i \leftarrow 0$;
  $error \leftarrow 2tol$;
  while $error > tol$ **do**
    `Solve`

$$d_u R_\theta(u_{h,i}^{k+1})(v_h)(\delta u_h) = -R_\theta(u_{h,i}^{k+1})(v_h) \ , \forall v_h \in V_h \tag{2.3}$$

    `Update` $u_{h,i+1}^{k+1} \leftarrow u_{h,i}^{k+1} + \delta u_h$;
    $error \leftarrow ||\delta u_h||_{L^2(\Omega)}$;
    $i \leftarrow i + 1$;
  end
end

Algorithm 1: Newton method for the nonlinear equation (2.5)

Now let $\delta_t > 0$ and $\{t_k\}_{k=0}^{N_T}$ be a partition of the interval $[0, T]$, with $N_T = T/dt$ and $t_{k+1} - t_k = \delta_t$, $k = 0, \ \ldots, N_T - 1$. We denote by u_h^k the approximation of u_h at $t = t_k$, $k = 0, \ldots, N_T$. To discretize the equation in time, we use the θ-method, yielding

$$\left(\frac{u_h^{k+1} - u_h^k}{\delta_t}, v_h\right)_\Omega + a(\theta u_h^{k+1} + (1-\theta)u_h^k, v_h) + c(\theta u_h^{k+1} + (1-\theta)u_h^k)(v_h) = 0, \ \forall v_h \in V_h, \tag{2.4}$$

being $\theta \in [0, 1]$. We recall that for $\theta = 0$, we obtain the Forward Euler method and for $\theta = 1$ the Backward Euler method.

To solve the fully discretized problem, we define for any $v_h \in V_h$

$$\begin{aligned} R_\theta(u_h^{k+1})(v_h) \ &= (u_h^{k+1}, v_h)_\Omega + \delta_t \theta a(u_h^{k+1}, v_h) + \delta_t c(\theta u_h^{k+1} + \delta_t(1-\theta)u_h^k)(v_h) \\ &\quad + \delta_t(1-\theta)a(u_h^k, v_h) - (u_h^k, v_h)_\Omega \end{aligned} \tag{2.5}$$

and apply the Newton method as detailled in Algorithm 1. Here, $u_{h,i}^k$ denotes the Newton approximation of u_h^k at the i^{th} step, for $k = 0, \ldots, N_t$, $i = 0, 1, \ldots$. The Newton method is presented in the Algorithm (1).

To assemble the algebraic structure corresponding to problem (2.3), we need to compute the Gateaux derivative of (2.5). We define $\phi_\theta(u) = \theta u + (1-\theta)u_h^{k-1}$ and we obtain

$$\begin{aligned} d_u R_\theta(u_{h,i}^{k+1})(v_h)(\delta u_h) &= (\delta u_h, v_h)_\Omega + \theta \delta_t a(\delta u_h, v_h) + \delta_t d_u c(\phi_\theta(u_{h,i}^{k+1}))(v_h)(\delta u_h) \\ &= (\delta u_h, v_h)_\Omega + \theta \delta_t a(\delta u_h, v_h) + \theta \delta_t(\sigma'(\phi_\theta(u_{h,i}^{k+1}))\delta u_h, v_h)_\Omega \end{aligned}$$

Finally, we observe that equation (2.3) is equivalent to :

Find $\delta u_h \in V_h$ satisfying

$$\begin{aligned} &(\delta u_h, v_h)_\Omega + \theta \delta_t a(\delta u_h, v_h) + \theta \delta_t(\sigma'(\phi_\theta(u_{h,i}^{k+1}))\delta u_h, v_h)_\Omega = \\ &(u_h^{k+1} - u_h^k, v_h)_\Omega + \delta_t a(\phi_\theta(u_{h,i}^{k+1}), v_h) + \delta_t(\sigma(\phi_\theta(u_{h,i}^{k+1})), v_h), \ \forall v_h \in V_h. \end{aligned} \tag{2.6}$$

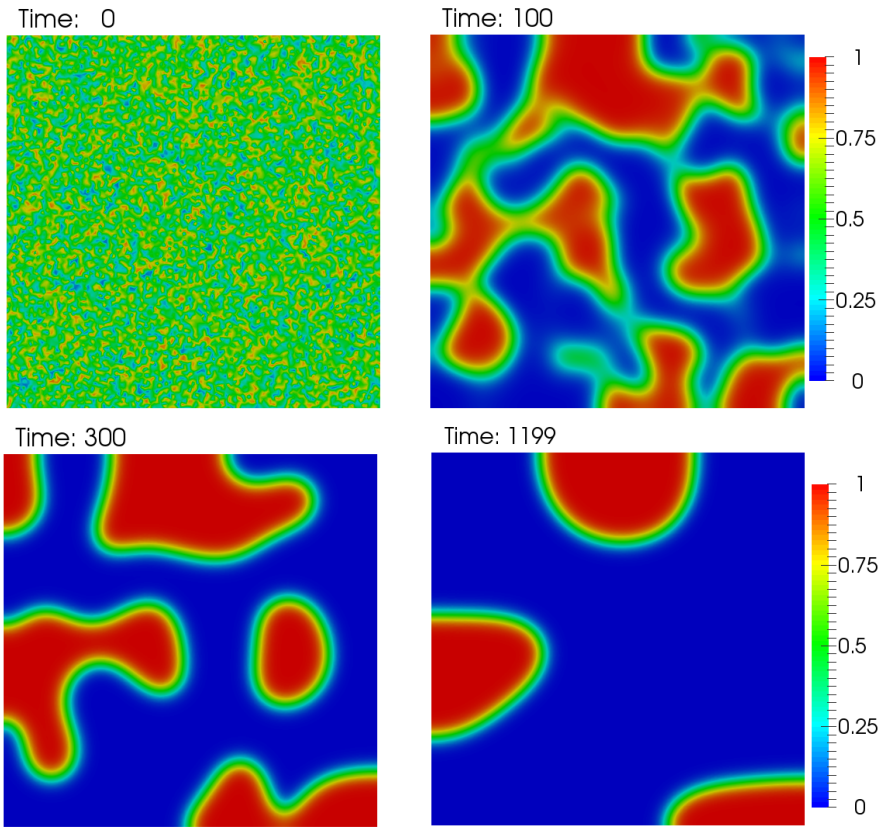

Figure 2.1: Result of the simulation of the Cahn-Allen equation.

In order to solve this problem at each iteration of the Newton algorithm, we define the following forms; for $k = 0, \ldots, N_t - 1$, $i = 0, 1, 2, \ldots$, let $v_h \in V_h$:

i) $\tilde{a}_{k,i}(\delta u_h, v_h) = (\mu\theta\delta_t \nabla\delta u_h, \nabla v_h)_\Omega + ((1 + \theta\delta_t\sigma'(\phi_\theta(u_{h,i}^{k+1})))\delta u_h, v_h)_\Omega$;

ii) $\tilde{f}_{k,i}(v_h) = (u_h^k - u_{h,i}^{k+1} - \delta_t\sigma\phi_\theta(u_{h,i}^{k+1}), v_h)_\Omega - (\theta\delta_t\nabla\phi_\theta(u_{h,i}^{k+1}), \nabla v_h)_\Omega$

Now, if $\delta u = \sum_{m=1}^{N^h} \delta u_m \psi_m$ denotes the movement obtained at each Newton step, writing $\tilde{A}_{l,m} = \tilde{a}_{k,i}(\phi_m, \phi_l)$, $\tilde{F}_l = \tilde{f}_{k,i}(\phi_l)$, $1 \leq l, m \leq N_h$ and $(\delta U)_m = \delta u_m$, we obtain the linear system

$$\tilde{A}\delta U = \tilde{F}.$$

In Figure 2.1 we show the results we obtained for the Cahn-Allen equation using a discretized initial condition $\{u_j(0)\}_{j=1}^{N_h}$ like in (2.1) where

$$u_j(0) = \frac{1}{2} + \frac{3}{4}(\text{rand}(0,1) - \frac{1}{2}), \quad j = 1, \ldots, N_h,$$

and $\theta = \frac{1}{2}$ for this simulation, corresponding to the Crank-Nicholson scheme. We can observe that the phase separation starts quickly until the time step 100, then goes slowly. Since the system is not mass conservative, it is supposed to converge to the steady state $u = 0$ or $u = 1$ after a long-time simulation; see e.g. [8] for further details.

2.2 Numerical Approximation of Tumor growth system

In this section we show how to approximate the solution of the system of equations (1.20) with a semi-discretization in space relying on the finite element method and a time-discretization exploiting the Backward Euler scheme.

Let $\mathcal{T}_h$ be a triangulation of Ω, where we denote by $h = \max_{K\in\mathcal{T}_h} h_k$, by N_h the number of interior and Neumann boundary nodes in the mesh and by N_h^t the total number of nodes, where Dirichlet boundary nodes are numbered at last for a sake of simplicity. A $\mathbb{P}_1$ finite element basis is used here and we set $V_h = X_h^1 \cap V$, like defined in section 2.1 with $\{\psi_j\}_{j=1}^{N_h}$ the finite element basis. For $t \in [0,T]$ and for $v_h(t) \in V_h$ we express

$$v_h(\mathbf{x},t) = \sum_{j=1}^{N_h^t} v_j(t)\psi_j(\mathbf{x}).$$

We define also $V_{0,h} = \{v_h \in V_h : v_j = 0,\ \forall j = N_h+1,\dots,N_h^t\}$. Now let $\delta_t > 0$ and $\{t_k\}_{k=0}^{N_t}$ be a partition of $[0,T]$, with $t_k = k\delta_t$ and $t_{N_t} = T$; let us denote by φ_h^k the time-approximation of $\varphi_h(t_k)$. After the semi-discretization in space and time discretization using a Backward Euler scheme for the equation (1.23), we obtain the following residuals :

$$\begin{aligned}
&1.\ R_1(\varphi_h^k,\mu_h^k,\sigma_h^k)(v_h) = (\varphi_h^k - \varphi_h^{k-1}) + \tfrac{1}{2}\delta_t((1+\varphi_h^k)^2\nabla\mu_h^k,\nabla v_h) - \delta_t((\mathcal{P}\sigma - \mathcal{A})(\varphi_h^k+1), v_h);\\
&2.\ R_2(\varphi_h^k,\mu_h^k,\sigma_h^k)(\xi_h) = (\mu_h^k,\xi_h) + \tfrac{\beta}{\varepsilon}((1-\varphi_h^{k^2})\varphi_h^k,\xi_h) - \beta\varepsilon(\nabla\varphi_h^k,\nabla\xi_h) + (\chi_\varphi\sigma_h^k,\xi_h);\\
&3.\ R_3(\varphi_h^k,\mu_h^k,\sigma_h^k)(w_h) = (D(\varphi_h^k)(\nabla\sigma_h^k - \lambda\nabla\varphi_h^k),\nabla w_h) + \tfrac{1}{2}(\mathcal{C}\sigma_h^k(\varphi_h^k+1), w_h).
\end{aligned} \tag{2.7}$$

We define now the residual

$$R(\varphi_h^k,\mu_h^k,\sigma_h^k)(v_h,\xi_h,w_h) = \begin{pmatrix} R_1(\varphi_h^k,\mu_h^k,\sigma_h^k)(v_h)\\ R_2(\varphi_h^k,\mu_h^k,\sigma_h^k)(\xi_h)\\ R_3(\varphi_h^k,\mu_h^k,\sigma_h^k)(w_h)\end{pmatrix}. \tag{2.8}$$

We define the Newton approximation's functions $\varphi_{h,i}^k, \mu_{h,i}^k, \sigma_{h,i}^k$, for $i = 0,1,\dots$. Then, the Newton Algorithm 2 is applied to solve $R(\varphi_h^k,\mu_h^k,\sigma_h^k)(v_h,\xi_h,w_h) = 0\ \forall(v_h,\xi_h) \in V_h\times V_h, w_h \in V_{0,h}$, for $k = 0,\dots,N_t$, using the equation

$$dR(\varphi_{h,i}^k,\mu_{h,i}^k,\sigma_{h,i}^k)(v_h,\xi_h,w_h)(\delta\varphi,\delta\mu,\delta\sigma) = -R(\varphi_{h,i}^k,\mu_{h,i}^k,\sigma_{h,i}^k)(v_h,\xi_h,w_h) \tag{2.9}$$

$\forall(v_h,\xi_h) \in V_h\times V_h, w_h \in V_{0,h}$ for $k = 1,\dots,N_t$ to update the Newton approximation functions.

To assemble the algebraic structure corresponding to problem (2.9) and for the sake of simplicity, let us rewrite our functions like

$$\varphi_h = u_1,\ \mu_h = u_2,\ \sigma_h = u_3 \text{ and } \delta\varphi = \delta u_1,\ \delta\mu = \delta u_2,\ \delta\sigma = \delta u_3.$$

We denote by $u_i = \sum_{j=1}^{N_h^t} u_{i,j}\psi_j$, where $(U_i)_j = u_{i,j}$ the nodal value at the node x_j, $j = 1,\dots,N_h^t$, for the function u_i, $i = 1,2,3$. Similarly for the functions δu_i, the nodal value vector is denoted by δU_i.

input : $\mathcal{T}_h$, δ_t,T, $\Pi_h\varphi_0$, R, dR, tol
output: Numerical sets of solutions $(\varphi_h^k, \mu_h^k, \sigma_h^k)$ in V_h for $k = 0, \ldots, N_t$

$\varphi^0 \leftarrow \Pi_h\varphi 0$;
$\mu_h^0 \leftarrow 0$;
`Solve` $R_3(\varphi_h^0, \mu_h^k, \sigma_h^0)(w_h) = 0,\ \forall w_h \in V_{h,0}$, to find σ_h^0;
for $k \leftarrow 0$ **to** $N_t - 1$ **do**
  $\varphi_{h,0}^{k+1} \leftarrow u_h^k$;
  $\mu_{h,0}^{k+1} \leftarrow \mu_h^k$;
  $\sigma_{h,0}^{k+1} \leftarrow \sigma_h^k$;
  $i \leftarrow 0$;
  $error \leftarrow 2tol$;
  while $error > tol$ **do**
    `Solve` Equation (2.9) to find $(\delta\varphi, \delta\mu, \delta\sigma)$, such that $\delta\sigma = 0,\ \text{on } \partial\Omega$.
    `Update` $\varphi_{h,i+1}^{k+1} \leftarrow \varphi_{h,i}^{k+1} + \delta\varphi$;
    `Update` $\mu_{h,i+1}^{k+1} \leftarrow \mu_{h,i}^{k+1} + \delta\mu$;
    `Update` $\sigma_{h,i+1}^{k+1} \leftarrow \sigma_{h,i}^{k+1} + \delta\sigma$;
    $error \leftarrow ||\delta\varphi||_{L^2(\Omega)} + ||\delta\mu||_{L^2(\Omega)} + ||\delta\sigma||_{L^2(\Omega)}$;
    $i \leftarrow i + 1$;
  end
end

Algorithm 2: Newton method for the nonlinear system of equation (2.9)

Then we obtain the following matrix formulation for the equation (2.9) :

$$\begin{pmatrix} A_{11} & A_{12} & A_{13} \\ A_{21} & A_{22} & A_{23} \\ A_{31} & A_{32} & A_{33} \end{pmatrix} \begin{pmatrix} \delta U_1 \\ \delta U_2 \\ \delta U_3 \end{pmatrix} = \begin{pmatrix} F_1 \\ F_2 \\ F_3 \end{pmatrix}, \tag{2.10}$$

where

$$\begin{aligned} (A_{lm})_{no} &= d_{u_m} R_l(u_{1,i}^k, u_{2,i}^k, u_{3,i}^k)(\psi_n)(\psi_o), \quad l, m = 1, 2, 3,\ n, o = 1, \ldots, N_h^t \\ (F_l)_n &= -R_l(u_{1,i}^k, u_{2,i}^k, u_{3,i}^k)(\psi_n), \qquad\qquad l = 1, 2, 3,\ n = 1, \ldots, N_h^t. \end{aligned} \tag{2.11}$$

In order to apply homogeneous Dirichlet boundary conditions to $\delta\sigma$ at each Newton iteration, we must set $(\delta U_3)_j = 0$, for $j = N_h + 1, \ldots, N_h^t$ and these matrices are then restrained to the inner domain dimension, so that we obtain

$$A_{3,m}^I \in \mathbb{R}^{N_h \times N_h^t},\ A_{l,3}^I \in \mathbb{R}^{N_h^t \times N_h}, A_{3,3}^I \in \mathbb{R}^{N_h \times N_h^t}, F_l^I \in \mathbb{R}^{N_h}\ l, m = 1, 2.$$

Since we impose $\delta u_3 = 0$ on $\partial\Omega$, F_3^I remains unchanged. By setting $A_{lm}^I = A_{lm}$ and $F_l^I = F_l$ for $l, m = 1, 2$, we finally obtain our system :

$$A^I \delta U = F^I,$$

with $(A^I)_{lm} = A_{lm}^I$ and $(F^I)_l = F_l^I$, $l, m = 1, 2, 3$. This is the way the code has been implemented to solve the numerical approximation of the problem (1.21), using the redbKIT toolbox [5] in Matlab.

Chapter 3

Numerical Results

In this section we present the results we obtained by implementing the techniques presented in the previous chapter. First we show that our code allow us to reproduce the numerical results obtained in [1]. Then, we present other simulations where different parameter settings have been considered in order to catch a general growth of the tumor depending on these latter. Different mesh strategies have employed for the simulations. Indeed, the interface thickness is proportional to ε, hence it is important, denoting by h the maximum diameter of an element in the interface layer, to have at least $h \sim \varepsilon$, or rather much smaller. For these first simulations, the mesh has been refined at the beginning of the simulation in order to be very fine in the passage of the interface, i.e. from a general guess of the area $\{x \in \Omega : |\varphi(x,t)| < 1, \forall t \in [0,T]\}$ for $t \in [0,T]$ and the mesh has been refined several times in this area. In the second part of this chapter, we will explain how we tackled this problem by using an adaptive mesh algorithm instead.

The number of time the mesh has been refined is denoted by N^o_{ref}. The value of h in the refined area must be divided by $2^{N^o_{\text{ref}}}$ to obtain an upper bound of h. The mesh is regularized after each refinement to keep the elements quality high, meaning that the actual value of the larger diameter of an element in the refined zone is actually smaller. The maximum diameter of an element outside the refined zone is denoted by h_{coarse} in the parameters tables. Furthermore, two other refinements have been applied to the boundary elements where the flux of nutrients is calculated in order to have more precise measures. The total amount of elements is $174'607$. The mesh used during the entire simulation is shown in Figure 3.1.

We consider in these simulations $\Omega = (0,12)^2$. Indeed, the numerical results presented in [1] show that the growth of the tumor is symmetrical on both axes x and y. This motivates us to use only a quarter of the domain that has been used in the article. Hence, we define

$$\Gamma_{\text{obs}} = \{(x,y) \in \Omega : x = 12 \text{ or } y = 12\}.$$

The system of equations (1.21) remains the same, except that we set homogeneous Neumann boundary conditions on $\partial\Omega \backslash \Gamma_{\text{out}}$ and Dirichlet boundary condition on Γ_{out} for the nutrients. The flux of nutrients is then only calculated on Γ_{out}, so the definition becomes

$$\mathcal{F}_{\text{nut}}(t) = \int_{\Gamma_{\text{out}}} D(\varphi(t)) \nabla\sigma(t) \cdot \nu \, \mathrm{d}\gamma. \tag{3.1}$$

Furthermore, the equations (1.35) and (1.36) remains correct for this definition of $\mathcal{F}_{\text{nut}}$

for $\Omega = [0, 12]^2$ and the above modified boundary conditions.

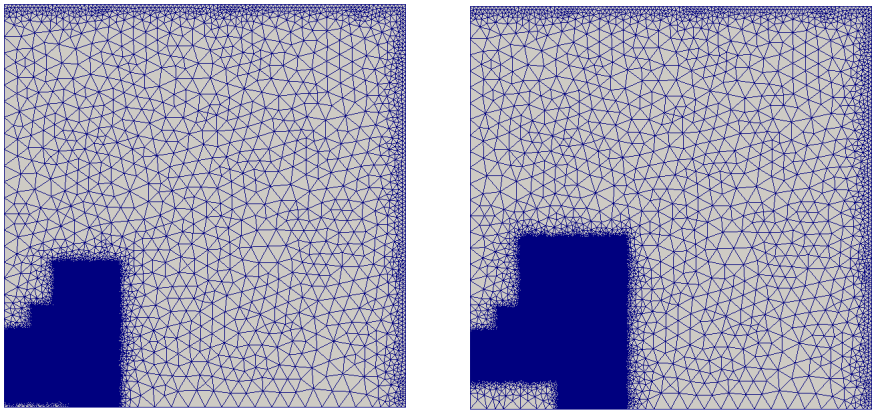

Figure 3.1: Adapted mesh used for the simulations of the tumor growth model on the left. On the right, another mesh used for the simulation with results on Figure 3.4 is shown

h_{coarse}	N^o_{ref}	ε	$\mathcal{P}$	λ	χ_φ	$\mathcal{A}$	D	β	$\mathcal{C}$	δt	T
0.5	5	h	$[0.1, 0.5, 0.8]$	0.03	5	0	1	0.1	2	10^{-3}	10

Table 3.1: Set of parameters for the simulations displayed in Figure 3.2, 3.3 and 3.4.

For the first simulations, we use the same parameter as used in [1], investigating the influence of the parameter $\mathcal{P}$ in the growth of the tumor. We recall that this parameter takes place in the source term for the tumour cells and create the tumour propagation. We make three different simulations, using $\mathcal{P} = 0.1, 0.5, 0.8$ and the parameters listed in Table 3.1. The results are shown in Figure 3.2 for $\mathcal{P} = 0.1$, in Figure 3.3 for $\mathcal{P} = 0.5$ and in Figure 3.4 for $\mathcal{P} = 0.8$. We can see that a modification of $\mathcal{P}$ influences in a significant manner the shape of the tumor. Indeed, the "finger" of tumor arising at the end of the simulation becomes thicker with a larger value of $\mathcal{P}$. We also notice that the phase field interface is clearly visible in the simulation of nutrients, arising from a strictly positive value of the parameter λ.

In Figure 3.5 we compare the evolution of different characteristics of the tumor growth model for the parameter settings of Table 3.1. On top, the evolution of the free-energy is plotted on the left, that is the integral over the domain of the energy density function e in the equation (1.13). On the right we report the evolution of the approximated function (1.31) over $[0, T]$. At the bottom the evolution of the approximated functions (1.32) on the left and of (3.1) on the right are reported. We can observe that most of all, the mass of the tumor is the most influenced by the variation of the parameter $\mathcal{P}$. From (1.36) we know that M_{tum} is proportional to this parameter. Even if the profile of M_{tum} seems to be almost linear we observe it is not the case for the other graphs, whose slopes are suddenly changing almost at same time, between 4 and 5. We can see in the previous figures that it corresponds to the time when the tumor is changing significantly of shape, even if the mass is still growing linearly.

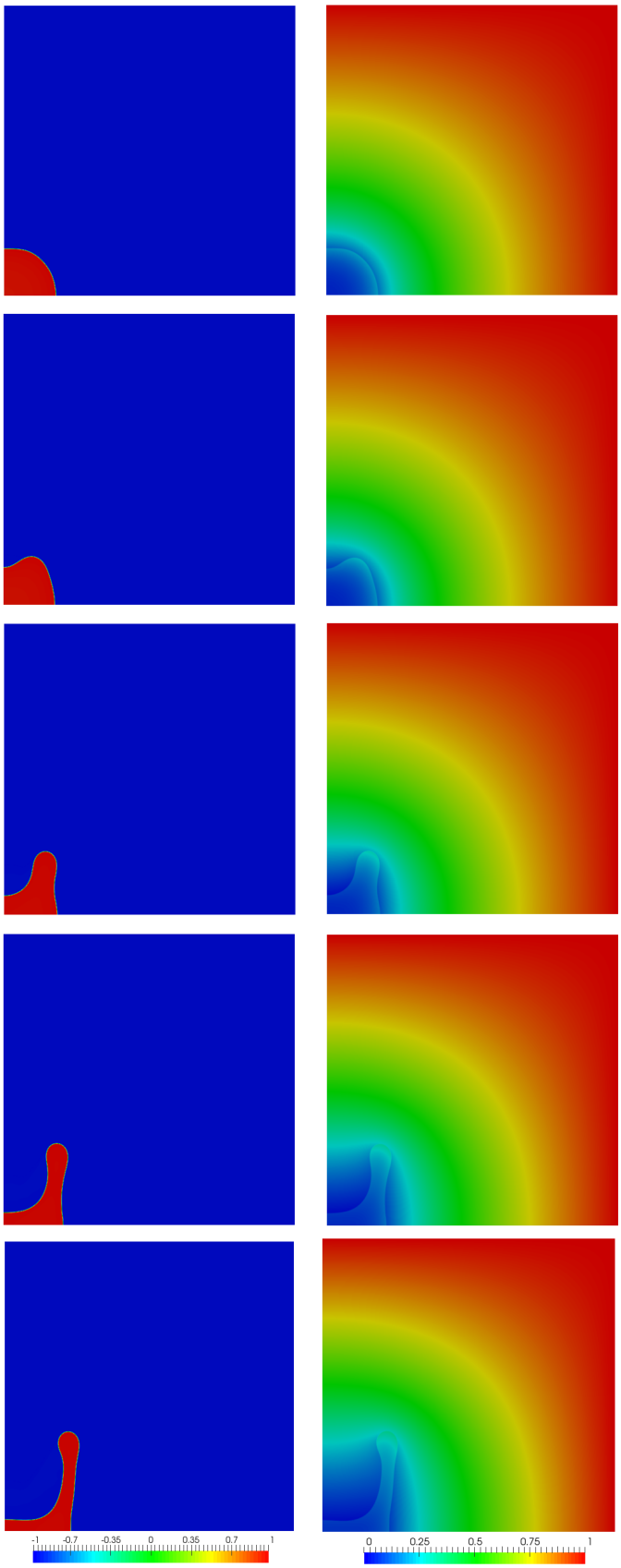

Figure 3.2: Results of the simulation using the parameters in Table 3.1 for $\mathcal{P} = 0.1$ at time $2, 4, 6, 8, 10$ from top to bottom, for the tumor concentration function φ (left) and the nutrient concentration function σ (right).

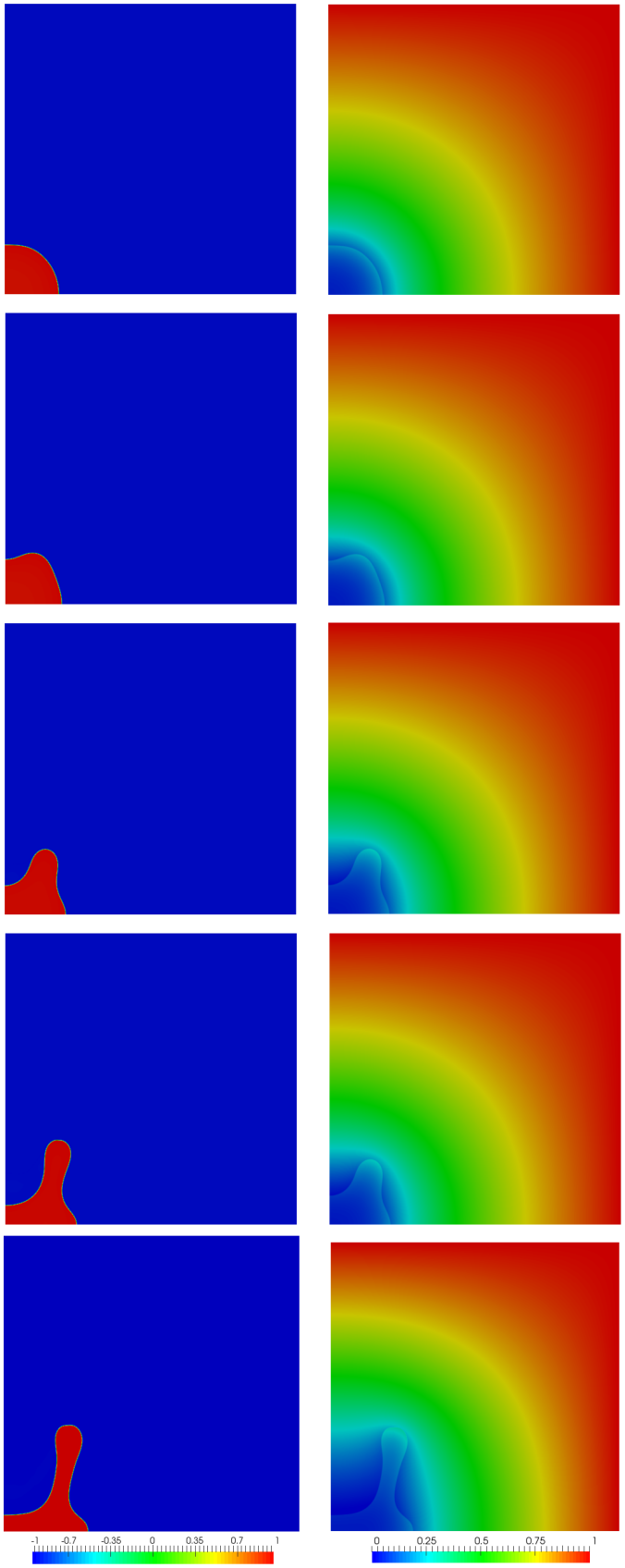

Figure 3.3: Results of the simulation using the parameters in Table 3.1 for $\mathcal{P} = 0.5$ at time $2, 4, 6, 8, 10$ from top to bottom, for the tumor concentration function φ (left) and the nutrient concentration function σ (right).

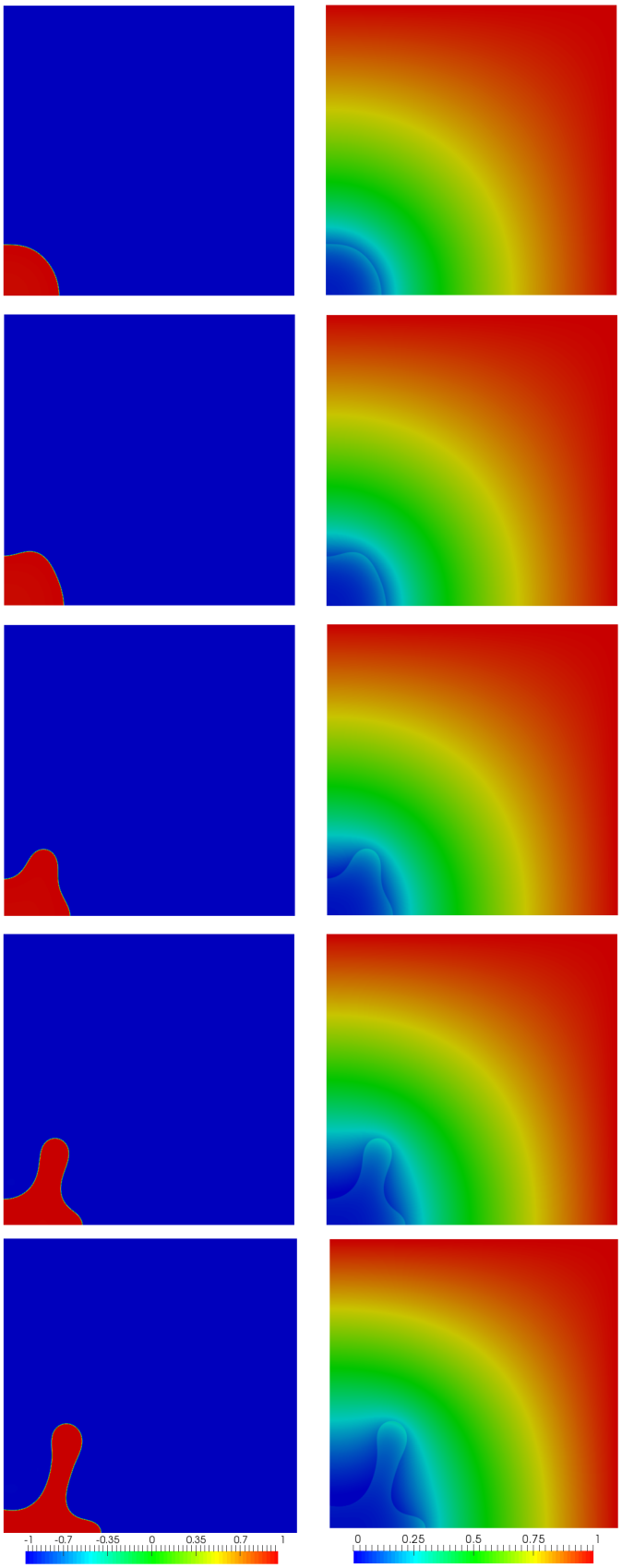

Figure 3.4: Results of the simulation using the parameters in Table 3.1 for $\mathcal{P} = 0.8$ at time $2, 4, 6, 8, 10$ from top to bottom, for the tumor concentration function φ (left) and the nutrient concentration function σ (right).

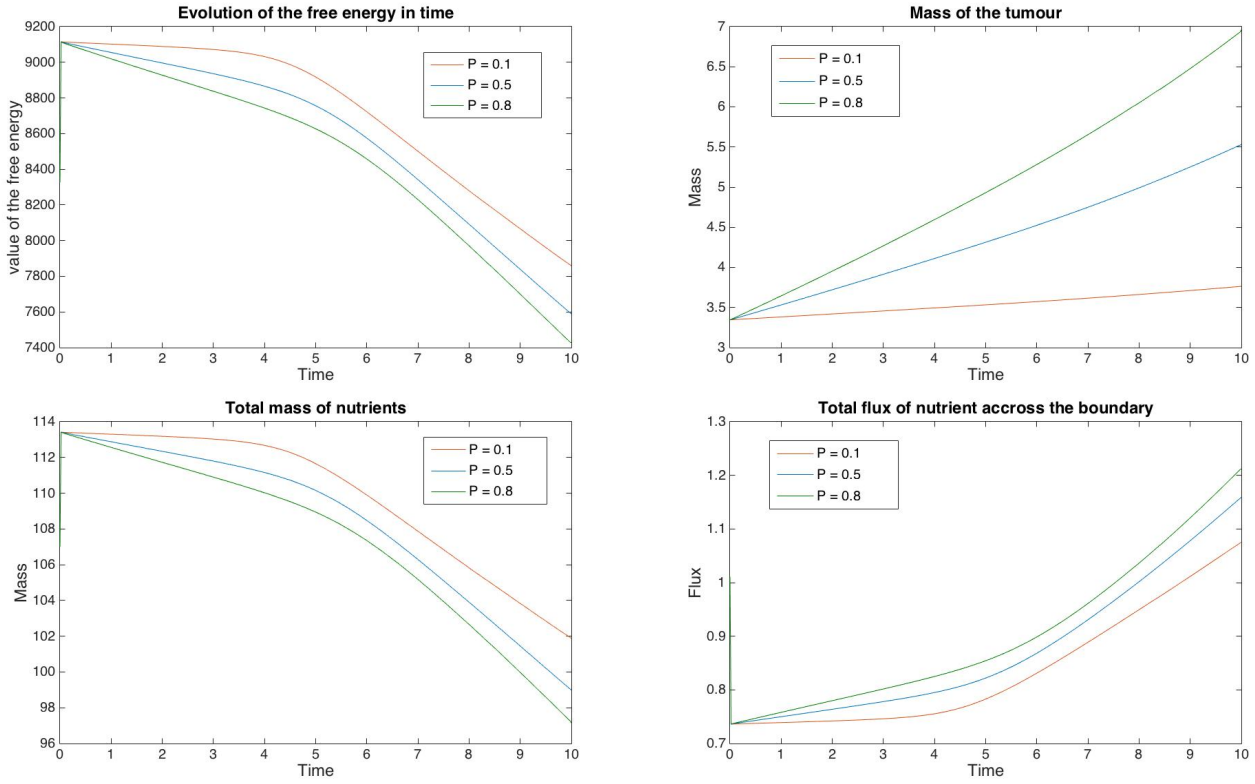

Figure 3.5: Evolution of the free-energy (top, left) and the evolution of M_{tum} (top, right). Evolution of M_{nut} (bottom, left) and of $\mathcal{F}_{\text{nut}}$ (bottom, right) for the simulation using the parameters in 3.1

In the previous simulations the computational time was large because adapting the mesh like we did in Figure 3.1 implies very costly mesh data : in average, one time step iteration took in average 150 seconds to run, for 3 Newton iterations. For this reason, we developed an adaptive mesh algorithm allowing to detect the elements in $\mathcal{T}_h$ where the sharp interface of the function φ is located and to refine over these elements. The mesh refinement frequency is given by a number of time steps $T_{\text{ref}} > 0$. From an initial very coarse mesh $\mathcal{T}_{h,init}$ a first coarser refinement is made before the simulation starts. And after T_{ref} time steps iterations in the Newton Algorithm 1, the refinement is made on the initial mesh. Hence, the mesh is adapted at each refinement from the initial coarse mesh $\mathcal{T}_{h,\text{init}}$. The detection of the sharp interface in an element is done by computing the L^2 norm of $\nabla\varphi$ on this element. Given a threshold inversely proportional to h, the element is refined if the L^2-norm is higher than this value. Finally, we denote by ΔK the element's neighbourhood of the element K. The Algorithm 3 describes the mesh adaptive algorithm. At each loop iteration of the Algorithm 3, the threshold is multiplied by 2, in order to adapt the maximum diameter of an element in the sharp interface region of $\mathcal{T}_{h,\text{ref}}$.

input : $\mathcal{T}_h$, $\mathcal{T}_{h,\text{init}}$, N^o_{ref}, k, T_{ref}, φ^k_h, μ^k_h, σ^k_h
output: $\mathcal{T}_h$, φ^k_h, μ^k_h, σ^k_h

Calculate the interpolant of φ^k_h, μ^k_h, σ^k_h on $\mathcal{T}_h$;
$\mathcal{T}_{h,\text{ref}} \leftarrow \mathcal{T}_{h,\text{init}}$;
for $j \leftarrow 1$ **to** N^o_{ref} **do**
    `Interpolate` φ^k_h on $\mathcal{T}_{h,\text{ref}}$;
    Elem_Refine $\leftarrow$ [ ] `// Initialize the set of elements to refine` ;
    for $K \in \mathcal{T}_{h,ref}$ **do**
        if $||\nabla\varphi^k_h||_{L^2(K)} > 2^{j-1}|K|/h$ **then**
            Elem_Refine $\leftarrow$ Elem_Refine $\cup\, \Delta K$;
        end
    end
    $K_{\text{ref}} \leftarrow$ refinement of Elem_Refine;
    $\mathcal{T}_{h,\text{ref}} \leftarrow \mathcal{T}_{h,\text{ref}} \cup K_{\text{ref}}$;
end
$\mathcal{T}_h \leftarrow \mathcal{T}_{h,\text{ref}}$;
`Interpolate` φ^k_h, μ^k_h, σ^k_h on $\mathcal{T}_h$;

Algorithm 3: Adaptive mesh algorithm for phase field model

h_{coarse}	N^o_{ref}	ε	$\mathcal{P}$	λ	χ_φ	$\mathcal{A}$	D	β	$\mathcal{C}$	δt	T
0.5	6	h	$[0.1, 0.5, 0.8]$	0.03	8	0	1	0.1	2	10^{-3}	6.5

Table 3.2: Set of parameters for the simulations displayed in Figures 3.6, 3.7 and 3.8.

For the next simulation we investigate the influence of χ_φ in the growth of the tumor. We will use the same value of $\mathcal{P}$ than for the previous simulations in order to compare the behaviour of the tumor for different values of $\mathcal{P}$ and χ_φ. The parameters are displayed in Table 3.2. The results of the simulation are shown on Figure 3.6, 3.7 and 3.8. The tumor-healthy cells evolution is displayed on the left, the nutrient concentration in the middle and the mesh resulting by applying the Algorithm 3 is displayed on the right.

In Figure 3.9 we compare the evolution of different characteristics of the tumor growth model for the parameter settings of Table 3.2. we can remark that for different values of $\mathcal{P}$, the evolution of M_{nut} and $\mathcal{F}_{\text{nut}}$ are pretty similar, compared with the simulations done with $\chi_\varphi = 5$. Up to time 6.5 for both Figure 3.5 and 3.9 the final mass of nutrients is smaller while the final flux is then larger. Since an increase of the parameter χ_φ also magnifies the chemotaxis effect, by enlarging the diffusion parameter in the tumor region, these effects are indeed expected. By equation (1.36), we expect a larger value of M_{tum} at time 6.5, now indeed can be remarked by comparing the top left graph of Figure 3.9 and the top right graph of Figure 3.5.

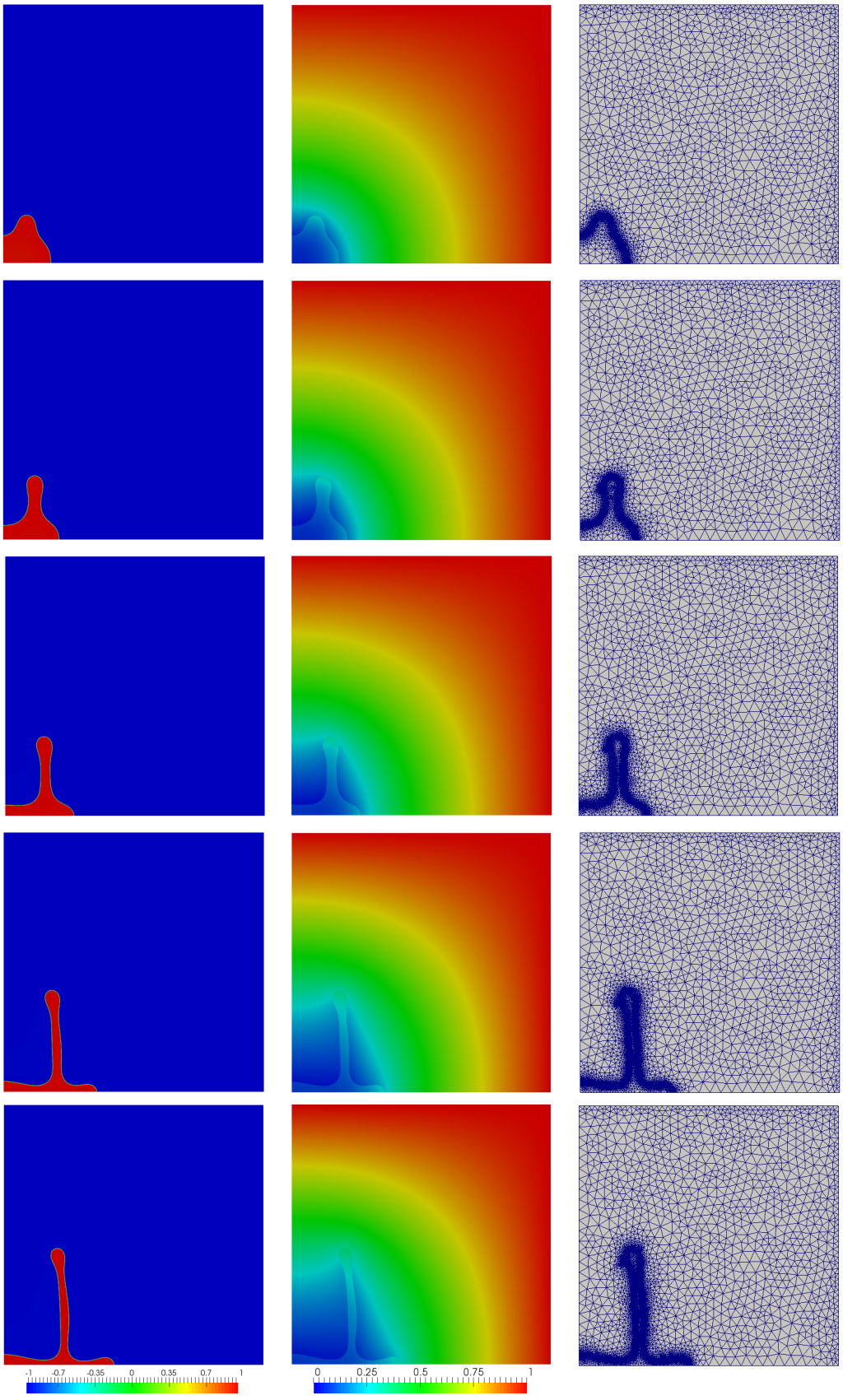

Figure 3.6: Results of the simulation using the parameters in Table 3.2 for $\mathcal{P} = 0.1$ at times $2, 3, 4, 5.5, 6.5$ from top to bottom. Left : function φ; middle : nutrient concentration σ; right : adapted mesh.

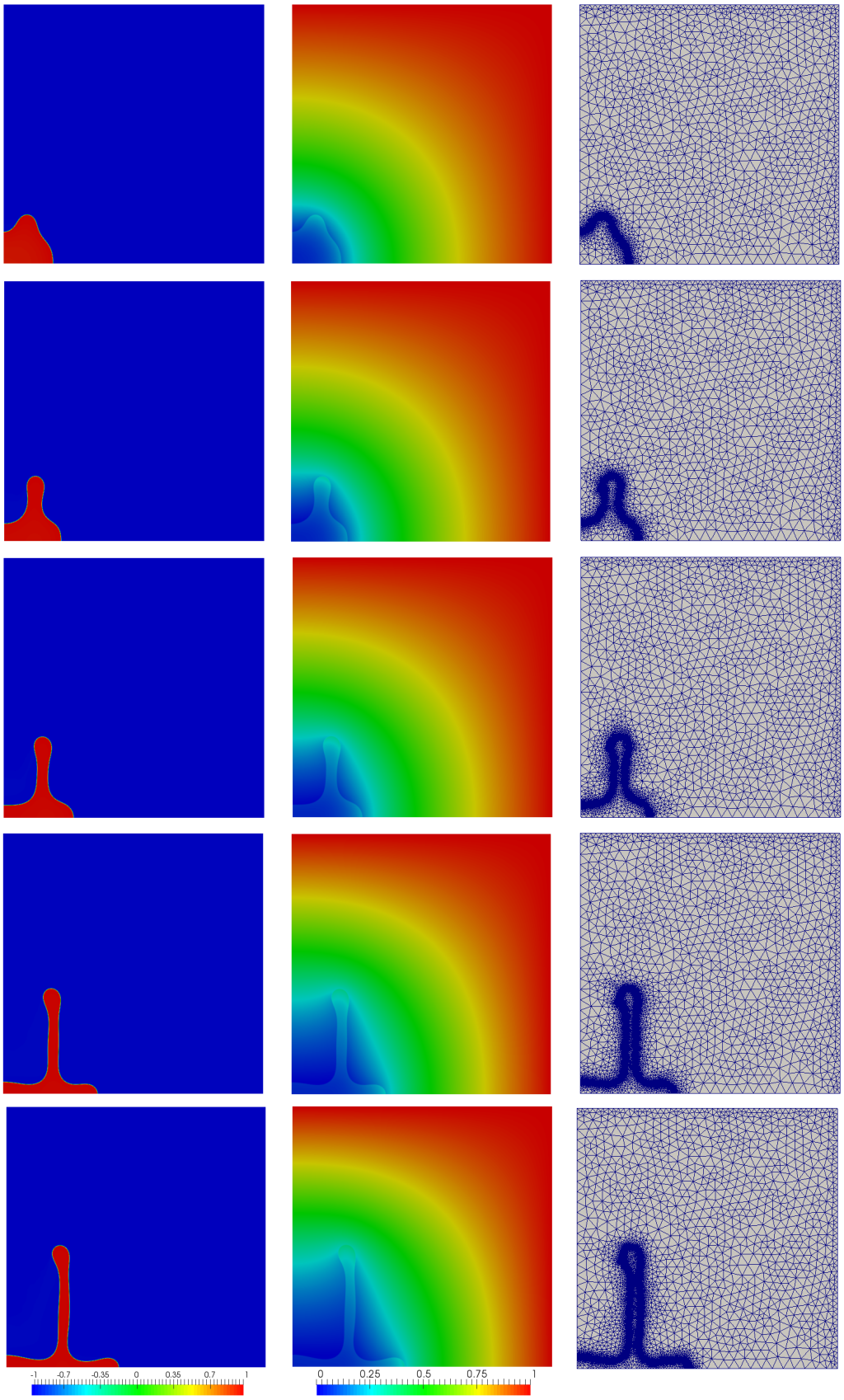

Figure 3.7: Results of the simulation using the parameters in Table 3.2 for $\mathcal{P} = 0.5$ at times $2, 3, 4, 5.5, 6.5$ from top to bottom. Left : function φ; middle : nutrient concentration σ; right : adapted mesh.

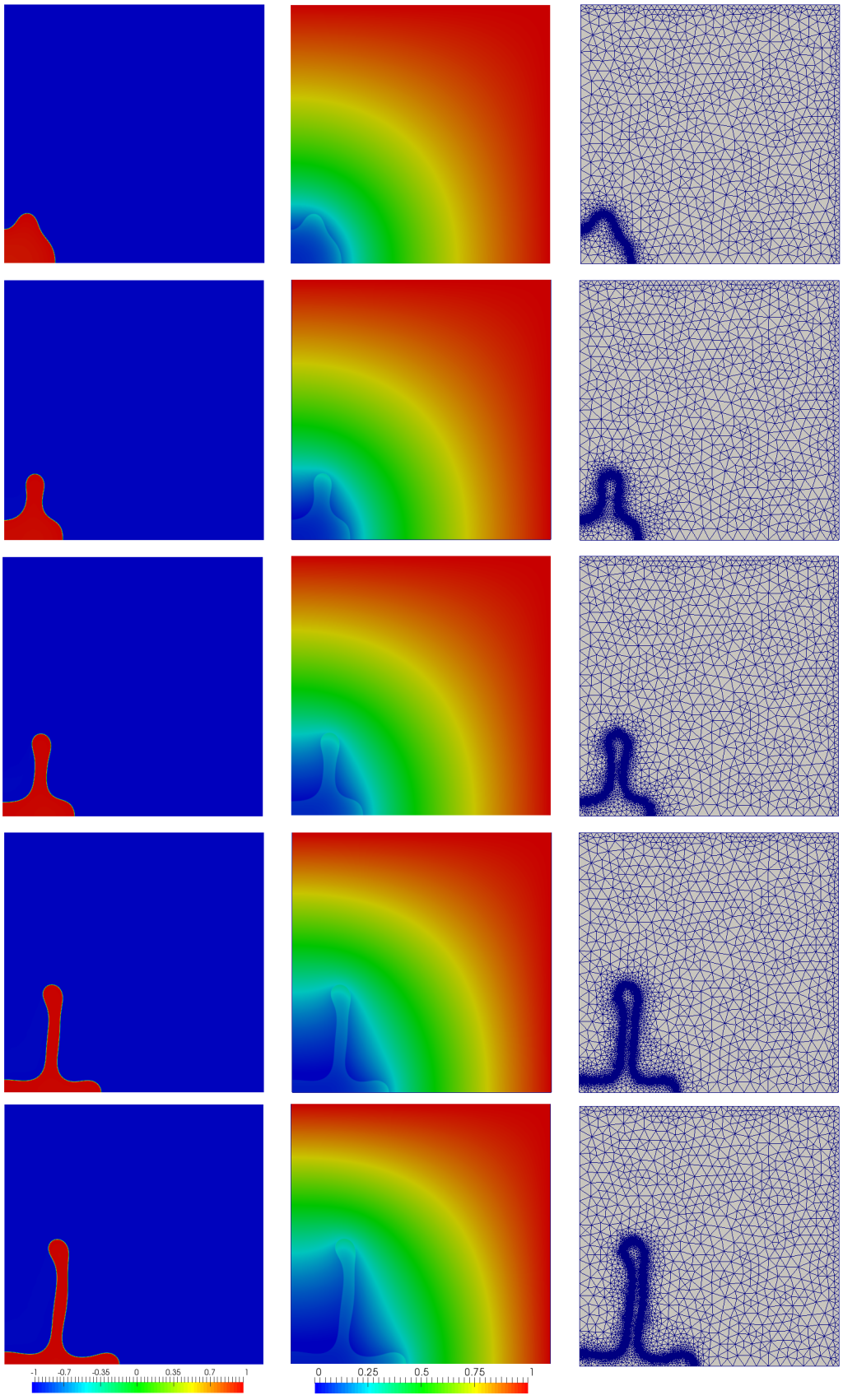

Figure 3.8: Results of the simulation using the parameters in Table 3.2 for $\mathcal{P} = 0.8$ at times $2, 3, 4, 5.5, 6.5$ from top to bottom. Left : function φ; middle : nutrient concentration σ; right : adapted mesh.

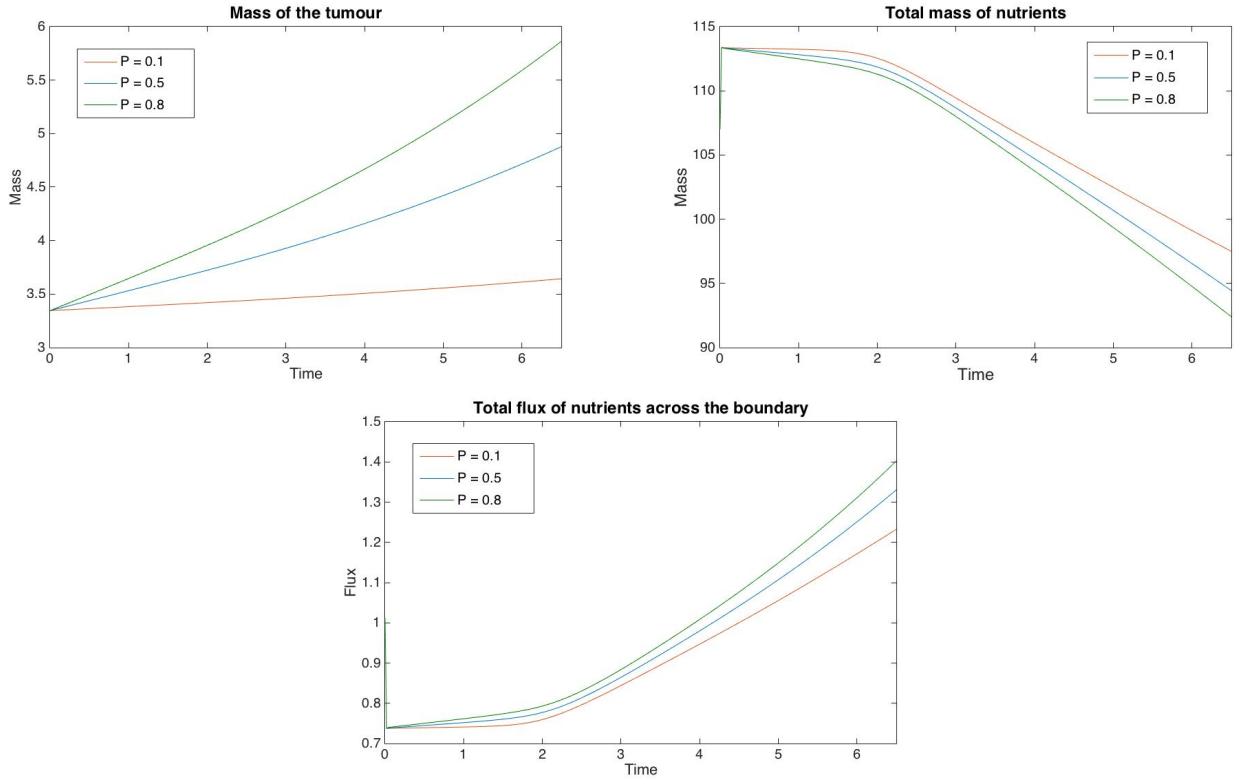

Figure 3.9: Evolution of M_{tum} (top left) and of M_{nut} (top right); evolution of $\mathcal{F}_{\text{nut}}$ (bottom) for the simulation using the parameters in 3.1

There are several improvements to report in the simulation by using this algorithm, with respect to the previous mesh strategy we have been using for simulations on Figures 3.2, 3.3 and 3.4. First the number of elements is drastically smaller: at the first refinement, the length of the interface allow to use only approximately 24′000 elements. At the end when the tumor has evolved, the length of the interface is much larger but still the number of elements never exceeded 132′000, which remains 25% less than for the mesh in the Figure 3.1. This implies a smaller computational time: an iteration of the Algorithm 3 could take up to 90 seconds for the largest length of the sharp interface, but one time step iteration of the Algorithm 1 took in average 100 seconds for 3 Newton iterations for the largest length of the interface. However, up to 3000 time step iterations, the number of elements was smaller than 60′000 and a time step iteration of the Newton Algorithm took up to 15 seconds for 3 Newton iterations. In average, a simulation of the tumour growth model's numerical approximation takes 1 day instead of 3, by using the adaptive mesh Algorithm 3. Secondly, the small number of elements implied in the simulation allows to make one more refinement at the interface, implying even more accuracy in the final solution. Nevertheless, a problem arising from the interpolation has to be mentioned : apparently the calculation of an interpolant for such a big amount of elements takes too much memory and the simulation hence stops.

Chapter 4

Inverse problems for tumor growth detection

From a medical point of view, a key question is related with the detection of the tumor in order to devise efficient therapeutic treatments. We saw that the total mass of the tumor is easily computable as long as we have the value of the total flux of nutrients around the infected zone. But even if we have these data, how can we detect the shape of the tumor at a given time T_{obs} in days ? Indeed it is not always possible to detect the tumor and some attempts have been made to approximate the concentration of the tumor in practice, see [4] or by using fluorescence ratio imaging microscopy, see reference in [3].

Our approach is to consider an observation in an observable subdomain around the tumor region of the nutrient concentration at a given time. Then, we want to investigate if this observation contains enough informations in order to recover the shape of the tumor, using the tumor growth model (1.21). For this purpose, from a target observation of nutrient, which in reality would come from clinical measurements, we want to approach as close as possible the nutrient function from the target and obtain a controlled function φ respecting the model, which will be our optimal control and allow to detect with precision the shape of our tumor. This approach can be formulated as an inverse problem, in the form of an optimal control.

4.1 Formulation of the inverse problem

An optimal control problem is described in the following way : in a system described by a partial differential equation, called the state problem, we calculate its solution $y \in \mathcal{Y}$ depending on a function $u \in \mathcal{U}_{\text{ad}} \subseteq \mathcal{U}$, called the control, for $\mathcal{U}, \mathcal{Y}$ Banach spaces. We have $y = y(u)$. An optimal control problem consists in finding the control such that a function $z = C(y) \in \mathcal{Z}$ of the solution of the state problem matches an observation z_{obs}, or $z = z_{\text{obs}}$, with $C : \mathcal{Y} \to \mathcal{Z}$ and $\mathcal{Z}$ a Banach space. The observation is called distributed if it takes its values in the domain. Otherwise it is called a boundary observation Such a problem may not have a solution. In this case, a control is called optimal if it minimizes a functional $J : \mathcal{U}_{\text{ad}} \to \mathbb{R}$. For more precision on the subject, existence and uniqueness of an optimal control and numerical methods to solve this problem, see e.g [12] or [13].

The weak direct problem from chapter 1 is the following : for $V = H^1(\Omega)$, $V_B = \{v \in V : v|_{\partial\Omega} = \sigma_B\}$ and $V_0 = H_0^1(\Omega)$, we want to to find $(\varphi(t), \mu(t), \sigma(t)) \in V \times V \times V_B$ for all $t \in [0, T], T > 0$ such that the following residuals are equal to 0 :

$$\begin{array}{lcll} R_1(\varphi,\mu,\sigma)(v) & = & \int_\Omega \frac{\partial\varphi}{\partial t} v + \frac{1}{2}(1+\varphi^2)\nabla\mu\cdot\nabla v - 2(\mathcal{P}\sigma - \mathcal{A})h(\varphi)\ \mathrm{d}\vec{x}, & \forall v \in V \\ R_2(\varphi,\mu,\sigma)(\xi) & = & \int_\Omega \mu\xi - \frac{\beta}{\varepsilon}\Psi'(\varphi)\xi - \beta\varepsilon\nabla\varphi\cdot\nabla\xi + \chi_\varphi\sigma\xi\ \mathrm{d}\vec{x}, & \forall \xi \in V \\ R_3(\varphi,\sigma)(w) & = & \int_\Omega D(\varphi)(\nabla\sigma - \lambda\nabla\varphi)\cdot\nabla w + \mathcal{C}\sigma h(\varphi)\ \mathrm{d}\vec{x}, & \forall w \in V_0, \end{array} \tag{4.1}$$

where $\Psi(\varphi) = \frac{1}{4}(1-\varphi^2)^2$ is the double-well potential, $D(\varphi) = \frac{1+\mathcal{D}}{2} + \varphi\frac{1-D}{2}$ is the mobility tensor and $\mathcal{P}, \mathcal{A}, \beta, \varepsilon, \chi_\varphi, \lambda, \mathcal{C}$ are parameters, whose meaning in the equations is explained in the chapter 1. Also, the function $h(\varphi)$ satisfy $h(1) = 1$ and $h(-1) = 0$. For the calculations of these residuals we set homogeneous Neumann boundary conditions on $\partial\Omega\backslash\Gamma_{\text{out}}$ and Dirichlet boundary condition on Γ_{out} for the nutrients. And the flux of nutrient is then calculated on Γ_{out}. Homogeneous Neumann boundary conditions have been imposed to the function φ and μ.

Let us now formulate the inverse problem. Let $z_{\text{aim}} \in \mathcal{Z}$ be our observation on nutrients at a given time $T_{\text{obs}} > 0$, for $\mathcal{Z}$ a Banach space. We define $C : H_B^1(\Omega) \to \mathcal{Z}$ an operator. We define $\mathcal{U} = V$, $\mathcal{Y} = V_B$ and the approach is the following : we want to find $\varphi \in V$, $\nabla\varphi\cdot\nu = 0$ on $\partial\Omega$, such that $C(\sigma) = z_{\text{aim}}$ and subject to

$$\begin{array}{lcll} R_3(\varphi,\sigma)(w) & = & 0, & \forall w \in V_0 \\ \sigma & = & \sigma_B, & \text{on } \Gamma_{\text{obs}} \\ D(\varphi)\nabla\sigma\cdot\nu & = & 0, & \text{on } \partial\Omega\backslash\Gamma_{\text{obs}}. \end{array} \tag{4.2}$$

The system (4.2) is our state system and we want to minimize $||C(\sigma) - z_{\text{aim}}||_{L^2(\Omega)}$ for σ the solution of the system. The residual $R_3(\varphi, \sigma)$ is privileged in the definition of the state problem for different reasons. First, it allows to calculate the state variable from a given control, which is a necessary condition for an optimal control problem, without needing to calculate the chemical potential μ. Furthermore, by using this particular residual, we avoid to treat a time dependant optimal control problem, allowing to save computational time.

However, we have no guarantee that the problem (4.2) has a unique solution. Hence we set the following functionals to minimize :

$$\begin{aligned} J_1(\varphi,\sigma) &= \frac{\alpha}{2}\int_\Omega \chi_{\Omega_{\text{obs}}}|(C(\sigma) - z_{\text{aim}})|^2\ \mathrm{d}\vec{x} + J_\mu(\varphi) + J_{\text{ref}}(\varphi), \\ J_2(\varphi,\sigma) &= \frac{\alpha}{2}\int_{\Gamma_{\text{out}}} |(C(\sigma) - z_{\text{aim}})|^2\ \mathrm{d}\gamma + J_\mu(\varphi) + J_{\text{ref}}(\varphi), \end{aligned} \tag{4.3}$$

with $\alpha > 0$ and $\chi_{\Omega_{\text{obs}}}$ the characteristic function for a subdomain $\Omega_{\text{obs}} \subseteq \Omega$, corresponding to a distributed observation for $J_1(\varphi, \sigma)$ and to a boundary observation for $J_2(\varphi, \sigma)$. The functionals $J_\mu(\varphi, \sigma)$ and $J_{\text{ref}}(\varphi)$ are regularization term for the optimal control problem for the state system (4.2). We gave them the following definitions

$$J_\mu(\varphi) = \frac{\beta}{\varepsilon}\int_\Omega \Psi(\varphi)\ \mathrm{d}\vec{x} + \frac{\beta\varepsilon}{2}\int_\Omega |\nabla\varphi|^2\ \mathrm{d}\vec{x} \tag{4.4}$$

and

$$J_{\text{ref}}(\varphi) = \frac{\gamma}{2}\int_\Omega (\varphi - \varphi_{\text{aim}})^2\ \mathrm{d}\vec{x}, \tag{4.5}$$

being φ_{aim} a sharp interface function close from the presumed optimal control and $\gamma > 0$ a small constant. The functional J_μ aims at minimizing the integral over Ω of the functional $f(\varphi)$, see equation (1.14), thus determining the interfacial energy of the system. It also allows the system to keep a sharp interface and to preserve the phases. The goal of the functional J_{ref} is to force the control to approach the optimal one.

We use the following data to apply to our optimal control problem: the target value $z_{\text{aim}} \in \mathcal{Z}$ is defined by a nutrient simulation σ_{target}, previously obtained in the chapter 3. The goal is to recover the simulated function φ_{target}, satisfying the system (4.2) and

$$J(\varphi_{\text{target}}, \sigma_{\text{target}}) = J_\mu(\varphi_{\text{target}}) + J_{\text{ref}}(\varphi_{\text{target}}) := J^\star.$$

4.2 Optimality conditions and numerical approximation

We start by considering a distributed observation in a subdomain $\Omega_{\text{obs}} \subseteq \Omega$. Indeed, since the domain is large comparing to the targeted function φ and considering the nature of the phase field model, a distributed observation is more reliable in term of existence of an optimal control, because we can approach the value of the nutrient concentration near the interface layer. The boundary observation case will be considered later on, as a more interesting approach in a practical point of view. We consider here the optimality conditions, given by the Lagrangian formulation of the inverse problem.

We denote by $J(\varphi, \sigma) = J_1(\varphi, \sigma)$ the cost functional for this optimal control problem. The observation domain is denoted by

$$\Omega_{\text{obs}} = \{(x, y) \in \Omega : x \geq a \text{ and } y \geq b\} \subseteq \Omega, \tag{4.6}$$

with the parameters $0 \leq a, b, \leq 12$. Hence, we define the operator $C : H^1(\Omega) \to H^1(\Omega)$ by the identity operator in the case of distributed observation. Now we can formulate our optimal control problem. Let $\sigma_B > 0$, the inverse problem becomes

Find $\varphi \in V$ such that $\nabla\varphi \cdot \nu = 0$ on $\partial\Omega$ and minimizing the functional $J_1(\varphi, \sigma)$ in (4.3) subject to

$$\begin{cases} -\nabla \cdot (D(\varphi)(\nabla\sigma - \lambda\nabla\varphi)) + \frac{1}{2}\mathcal{C}\sigma(\varphi + 1) = 0, & \text{in } \Omega \\ D(\varphi)\nabla\sigma \cdot \nu = 0, & \text{on } \partial\Omega\backslash\Gamma_{\text{out}} \\ \sigma = \sigma_B, & \text{on } \Gamma_{\text{out}}. \end{cases}$$

In order to solve this problem and for the sake of efficiency, we introduce a lifting function σ_0 and we set $\sigma = \sigma^\star + \sigma_0$ with $\sigma^\star|_{\Gamma_{\text{out}}} = 0$ and $\sigma_0 = \sigma_B$. In a same way we have $\sigma_{\text{aim}} = \sigma^\star_{\text{aim}} + \sigma_0$. We introduce the Lagrangian

$$\mathcal{L}(\sigma, p, \varphi) = J(\varphi, \sigma) - R_3(\varphi, \sigma)(p), \tag{4.7}$$

being $p \in V_0$ and $R_3(\varphi, \sigma)(p)$ the residual defined in the state system (4.2). Matching (4.7) with the lifting function we obtain the following explicit definition of the Lagrangian

:

$$\begin{aligned}\mathcal{L}(\sigma^\star, p, \sigma) = \frac{\alpha}{2}\int_\Omega \chi_{\Omega_{\text{obs}}}|(\sigma^\star - \sigma^\star_{\text{aim}})|^2 \,\mathrm{d}\vec{x} + \frac{\beta}{\varepsilon}\int_\Omega \Psi(\varphi)\,\mathrm{d}\vec{x} \\ + \frac{\beta\varepsilon}{2}\int_\Omega |\nabla\varphi|^2 \,\mathrm{d}\vec{x} + \frac{\gamma}{2}\int_\Omega(\varphi - \varphi_{\text{aim}})^2 \,\mathrm{d}\vec{x} \\ - \int_\Omega D(\varphi)(\nabla\sigma^\star - \lambda\nabla\varphi)\cdot\nabla p \,\mathrm{d}\vec{x} - \int_\Omega \frac{C}{2}(\sigma^\star + 1)(\varphi + 1)p \,\mathrm{d}\vec{x}\end{aligned} \tag{4.8}$$

A system of first-order necessary optimality conditions can be found by requiring that :

Find $(\sigma^\star, p, \varphi) \in (H^1_{\Gamma_{\text{out}}}(\Omega))^2 \times H^1(\Omega)$ such that

$$\nabla\mathcal{L}(\sigma^\star, p, \varphi)(w, \xi, v) = 0 \quad \forall (w, \xi, v) \in (H^1_{\Gamma_{\text{out}}}(\Omega))^2 \times H^1(\Omega) \tag{4.9}$$

Equation (4.9) implies to solve the following equations :

1. State equation : $\partial_p\mathcal{L}(\sigma, p, \varphi)(\xi) = 0,\ \forall \xi \in H^1_0(\Omega)$.

$$\begin{aligned}\int_\Omega D(\varphi)\nabla\sigma^\star\cdot\nabla\xi \,\mathrm{d}\vec{x} + \int_\Omega \frac{1}{2}\mathcal{C}(\varphi+1)\sigma^\star\xi \,\mathrm{d}\vec{x} = \lambda\int_\Omega D(\varphi)\nabla\varphi\cdot\nabla\xi \,\mathrm{d}\vec{x} \\ - \int_\Omega \frac{1}{2}\mathcal{C}(\varphi+1)\xi \,\mathrm{d}\vec{x},\ \forall\xi \in H^1_0(\Omega)\end{aligned} \tag{4.10}$$

2. Adjoint equation : $\partial_\sigma\mathcal{L}(\sigma, p, \varphi)(w) = 0,\ \forall w \in H^1_0(\Omega)$.

$$\int_\Omega D(\varphi)\nabla p\cdot\nabla w \,\mathrm{d}\vec{x} + \int_\Omega \frac{1}{2}\mathcal{C}(\varphi+1)pw \,\mathrm{d}\vec{x} = \alpha\int_\Omega(\chi_{\Omega_{\text{obs}}}(\sigma^\star - \sigma^\star_{\text{obs}}))w \,\mathrm{d}\vec{x} \tag{4.11}$$

3. Control equation : $\partial_\varphi\mathcal{L}(\sigma, p, \varphi)(v) = 0,\ \forall v \in H^1(\Omega)$.

$$\begin{aligned}0 = \int_\Omega\left[\frac{\beta}{\varepsilon}\Psi'(\varphi) + \gamma(\varphi - \varphi_{\text{aim}}) - D'(\varphi)(\nabla\sigma^\star - \lambda\nabla\varphi)\cdot\nabla p - \frac{1}{2}\mathcal{C}(\sigma^\star+1)p\right]v \,\mathrm{d}\vec{x} \\ + \int_\Omega[\beta\varepsilon\nabla\varphi + \lambda D(\varphi)\nabla p]\cdot\nabla v \,\mathrm{d}\vec{x},\ \forall v \in H^1(\Omega)\end{aligned} \tag{4.12}$$

In order to solve this problem numerically, we use a Discretize, Then Optimize scheme : let $h > 0$ and $\mathcal{T}_h$ a triangulation of Ω using the same notation than in chapter 2. Let N_h be the number of interior and Neumann boundary vertices and N_h^t the total number of nodes, keeping the notations of chapter 2. We denote by $\boldsymbol{\varphi}, \mathbf{p}, \boldsymbol{\sigma} \in \mathbb{R}^{N_h^t}$ respectively the vector of the nodal values of the discretized functions $\sigma_h^\star, p_h \in V_{h,0}$ and $\varphi_h \in V_h$. We consider $J : \mathbb{R}^{N_h^t} \to \mathbb{R}$ a function of $\boldsymbol{\varphi}$.

We consider the scheme for the numerical approximation of the optimal control problem found in [13], see also [12] for further informations. For $k \geq 0$ we denote by $(\boldsymbol{\varphi}^k, \mathbf{p}^k, \boldsymbol{\sigma}^k, J^k)$ the k^{th} iteration's variables in the Algorithm 4. We denote by $g_k \in \mathbb{R}^{N_h^t}$ the gradient of J^k. The gradient g_k can also be considered as the gradient of the function $J(\varphi_h^k, \sigma_h^k)$ and is calculated using the Control equation as follows

$$\int_\Omega g_k v_h \, \mathrm{d}\vec{x} = \mathcal{L}_\varphi(\sigma_h^k, p_h^k, \varphi_h^k)(v_h) \; \forall v_h \in V_h. \tag{4.13}$$

Following [13], we use the gradient method with Armijo rule to minimize the functional (4.3). Finally, we fix the constants $\bar{\sigma} \in (0, \frac{1}{2})$, $\bar{\beta} \in (0, 1)$, $\bar{\tau} > 0$ being Armijo parameters.

input : $\mathcal{T}_h$, ε, $\mathcal{P}$, λ, $\mathcal{A}$, D, β, $\mathcal{C}$, α, γ, $\boldsymbol{\varphi}^0$, Tol, $\bar{\sigma}$, $\bar{\tau}$, $\bar{\beta}$
output: J^k, $\boldsymbol{\sigma}^k$, $\mathbf{p}^k$, $\boldsymbol{\varphi}^k$

Compute the mass matrix M;
`Solve` (4.10) to find $\boldsymbol{\sigma}^0$;
`Solve` (4.11) to find $\mathbf{p}^0$;
`Solve` (4.3) to find $J^0 = J(\boldsymbol{\sigma}^0, \boldsymbol{\varphi}^0)$;
`Solve` $\partial_\varphi \mathcal{L}(\boldsymbol{\sigma}^0, \mathbf{p}^0, \boldsymbol{\varphi}^0)(v_h) = \int_\Omega g_0 v_h \, \mathrm{d}\vec{x}$ to find g_0;
$d_0 \leftarrow -g_0$;
$k \leftarrow 1$, $\mathrm{err}^k \leftarrow Tol + 1$;
while $k < N_{max}$ and $err^k > Tol$ **do**
  $j \leftarrow 0$;
  $\tau^j \leftarrow \bar{\tau}$;
  while $J^{k-1} - J^k \leq -\bar{\sigma}\tau^j {g_{k-1}}^T d_{k-1}$ **do**
    $j \leftarrow j + 1$;
    $\tau^j \leftarrow \bar{\tau}\bar{\beta}^j$;
    $\boldsymbol{\varphi}^k \leftarrow \boldsymbol{\varphi}^{k-1} + \tau^j d_{k-1}$;
    `Solve` (4.10) to find $\boldsymbol{\sigma}^k$;
    `Update` $J^k = J(\boldsymbol{\sigma}^k, \boldsymbol{\varphi}^k)$ using (4.3);
  end
  `Solve` (4.11) to find $\mathbf{p}^k$;
  `Solve` (4.13) to find g_k;
  $\mathrm{err}^k \leftarrow ||g_k||_{L^2(\Omega)} / ||g_0||_{L^2(\Omega)}$;
  $d_k \leftarrow -g_k$;
  $k \leftarrow k + 1$;
end

Algorithm 4: Scheme for the resolution of the inverse problem by use of the Armijo method

4.3 The case of distributed observation

In this section, we present the results we obtained by applying the Algorithm 4 to the inverse problem we set in section 4.2. The simulation of the direct problem we used in this section is the simulation using the parameters Table 3.1, for $\mathcal{P} = 0.5$. The numerical results of this simulation is displayed in Figure 3.3

The first numerical results test the validity of a basic problems : we set φ_{aim} being equal to the simulated function $\varphi(T_{\text{obs}})$, i.e. σ_{aim} is the finite element approximation of problem(4.2) obtained for $\varphi = \varphi_{\text{aim}}$. Furthermore, we use $\Omega_{\text{obs}} = \Omega$. Hence we used the parameters displayed in Table (4.1) and the evolution of the control function is shown at Figure (4.1) and 140 iterations of the steepest descent method in the Algorithm 4 have been done. We see that the tumour distribution in Ω can be recovered in this simple case. For the sake of practical use, we also made simulations using $\varphi_{\text{aim}} = \varphi_h^0$ and reducing γ to $\gamma = 1$. With this approach, we assume we do not know the final distribution of the tumour cells. Unfortunately the results we obtain on the control for this last test where not significant : The functional J does not decrease further than 0.64 and we observed almost no change on the control.

ε	λ	D	$\mathcal{C}$	T_{obs}	α	β	γ
0.01	0.03	1	2	10	20	0	20

Table 4.1: Parameters used for the test inverse problem with $\Omega_{\text{obs}} = \Omega$.

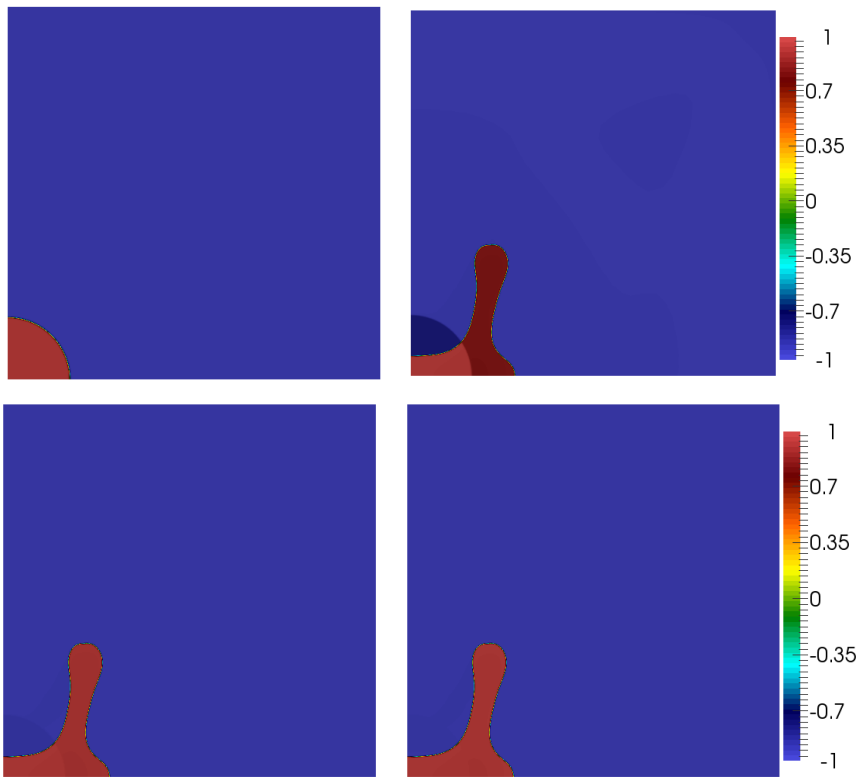

Figure 4.1: Control function for the test inverse problem using parameters in Table 4.1. From left to right and from top to bottom, iterations $0, 50, 100, 140$.

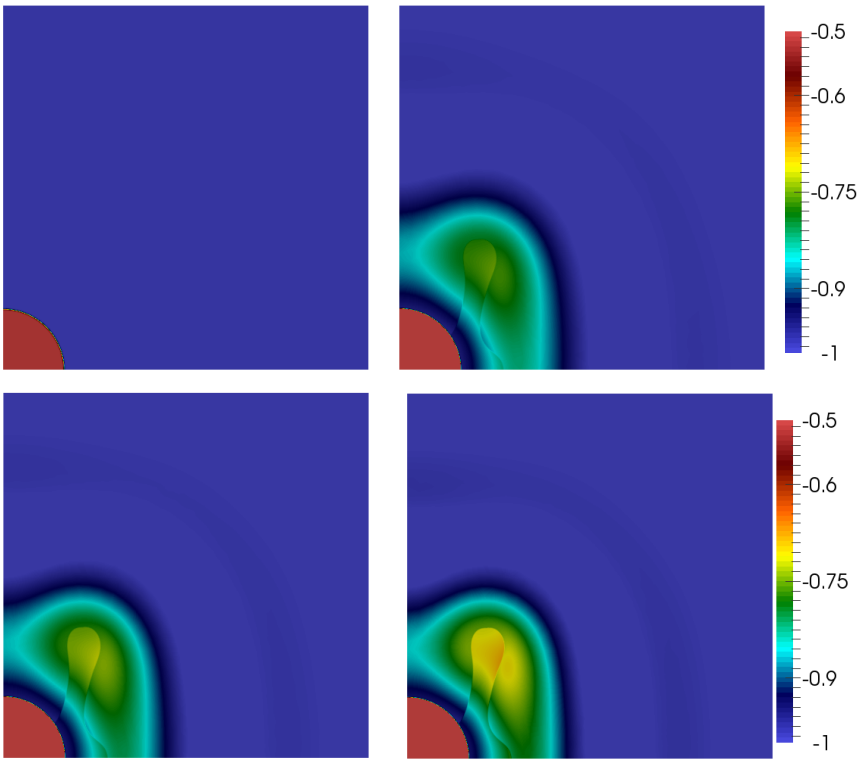

Figure 4.2: Control function for the inverse problem using parameters in Table 4.2. From left to right and from top to bottom, iterations $0, 250, 330, 500$

ε	λ	D	$\mathcal{C}$	T_{obs}	α	β	γ
0.01	0.03	1	2	10	20	$[0, 0.1]$	0

Table 4.2: Parameters used for the test inverse problem with $\Omega_{\text{obs}} = \Omega$

4.3.1 Test 1: circle-based initial control

In the next simulation, a circle-based initial control function is used and we set $\gamma = \beta = 0$. The parameters are shown on Table 4.2. The functional J_μ appears to prevent a modification of the control function towards the optimal control function, due to the fact that it minimizes a regularization term forcing the control to be either $\varphi = 1$ or $\varphi = -1$. A strategy we put in place for this simulation is to start the simulation using $\beta = 0$ and then increase it to $\beta = 0.1$ after a number $N_\beta = 500$ iterations. Unfortunately, the change of value of β makes the functional J to diverge. The results for the 500-th first iterations of the steepest descent method are shown in Figure 4.2 for the control function, in Figure 4.4 for the state function and in Figure 4.5 for the adjoint function. In Figure 4.3, we plot the evolution of the cost functional and of the L^2-norm of its approximated gradient g_k. We observe that the cost functional is strictly monotonic but appears to decrease slowly after a few iterations.

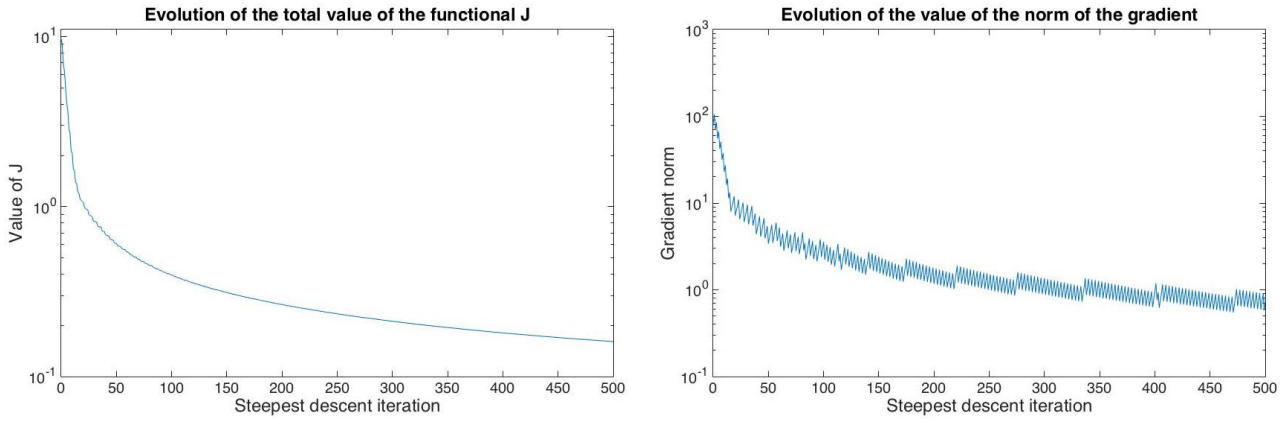

Figure 4.3: Evolution of the cost functional value (left); Evolution of $||g_k||_{L^2(\Omega)}$ (right) for the simulation with parameters setting in Table 4.2.

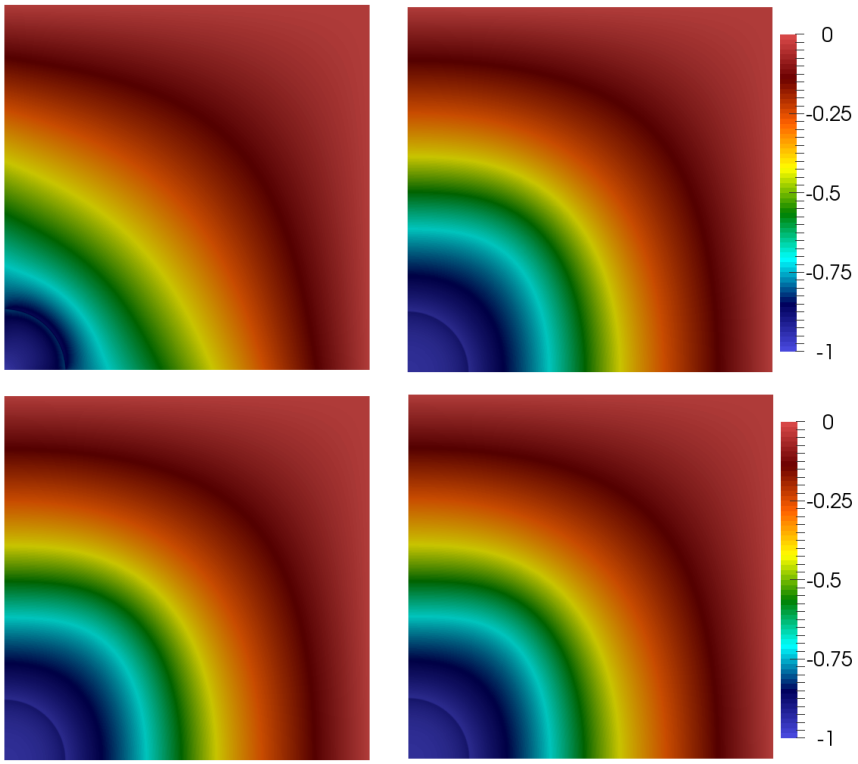

Figure 4.4: State function for the inverse problem using parameters in Table 4.2. From left to right and from top to bottom, iterations $0, 250, 330, 500$.

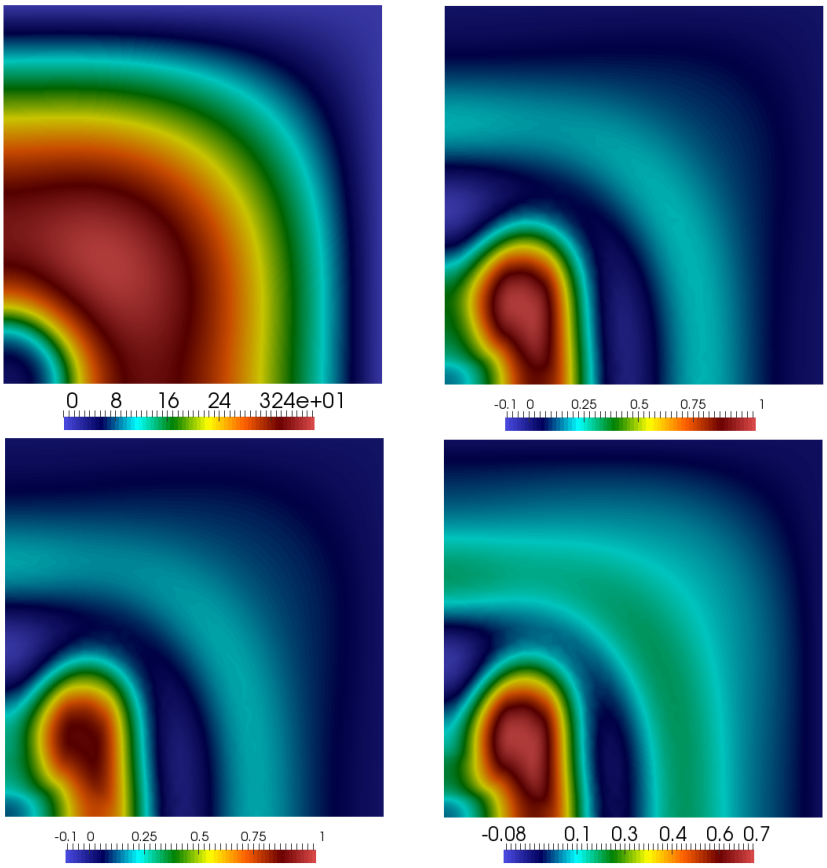

Figure 4.5: Adjoint function for the inverse problem using parameters in Table 4.2. From left to right and from top to bottom, iterations $0, 250, 330, 500$.

4.3.2 Test 2: zero-tumor initial control

Considering the first case, we could observe the interface of the optimal control rising in the control obtained at iteration 500. Unfortunately, the cost functional seems to stagnate and the adjoint function's maximum decreases to a very small value. Therefore, it might be that a local minimum of the cost functional has been reached. In order to tackle this problem, we suggest to use the exact same parameters as in Table 4.2, using an initial control $\varphi_0 = -1$ in Ω and use the same scheme as in the first case for the update of the parameter β during the simulation, after 500 iterations. This solution is supposed to prevent to fall on a possible local minimum. Unfortunately, we could still observe that the cost functional's value diverges when passing to $\beta = 0.1$. The resulting control function for this last problem is shown in Figure 4.8, before the update of the parameter β. In Figure 4.9 we show the evolution of the cost functional and its approximated gradient g_k. We observe that we can recover the interface of the optimal control for this simulation too, but not the entire optimal function.

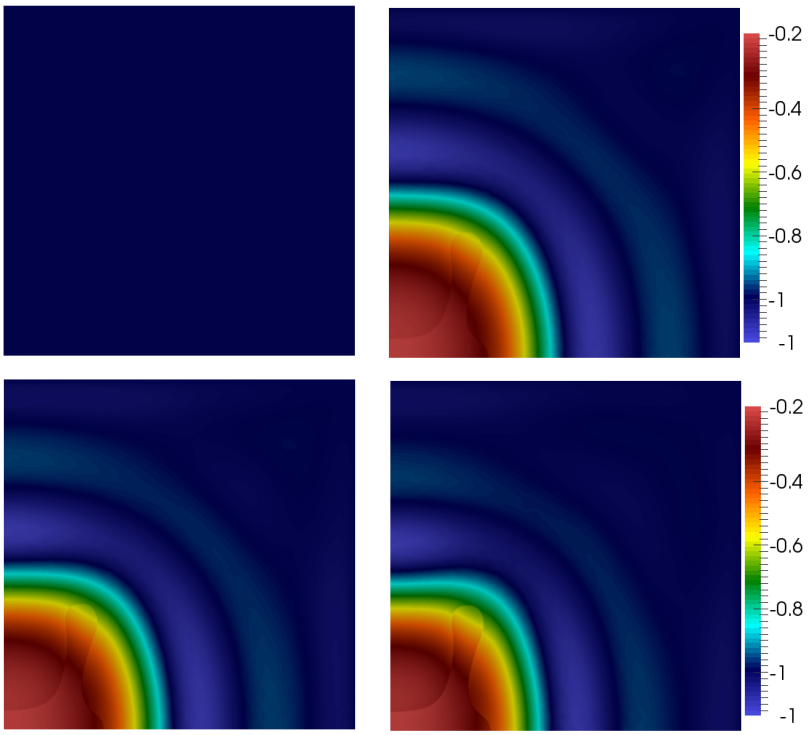

Figure 4.6: Control function for the inverse problem using parameters in Table 4.2 for the initial control $\varphi_h^0 = -1$. From left to right and from top to bottom, iterations $0, 250, 330, 500$.

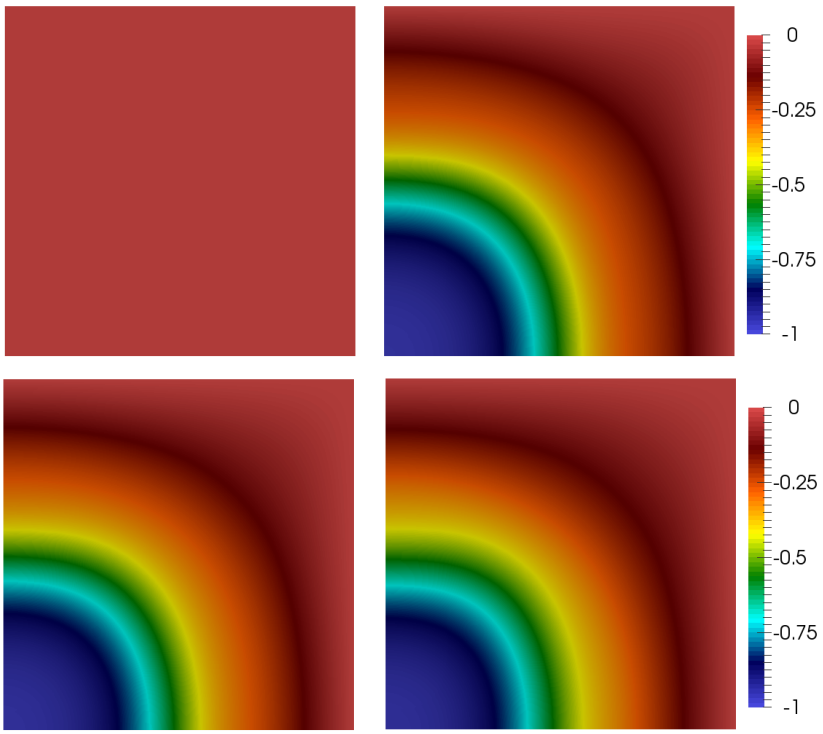

Figure 4.7: State function for the inverse problem using parameters in Table 4.2 for the initial control $\varphi_h^0 = -1$. From left to right and from top to bottom, iterations $0, 250, 330, 500$.

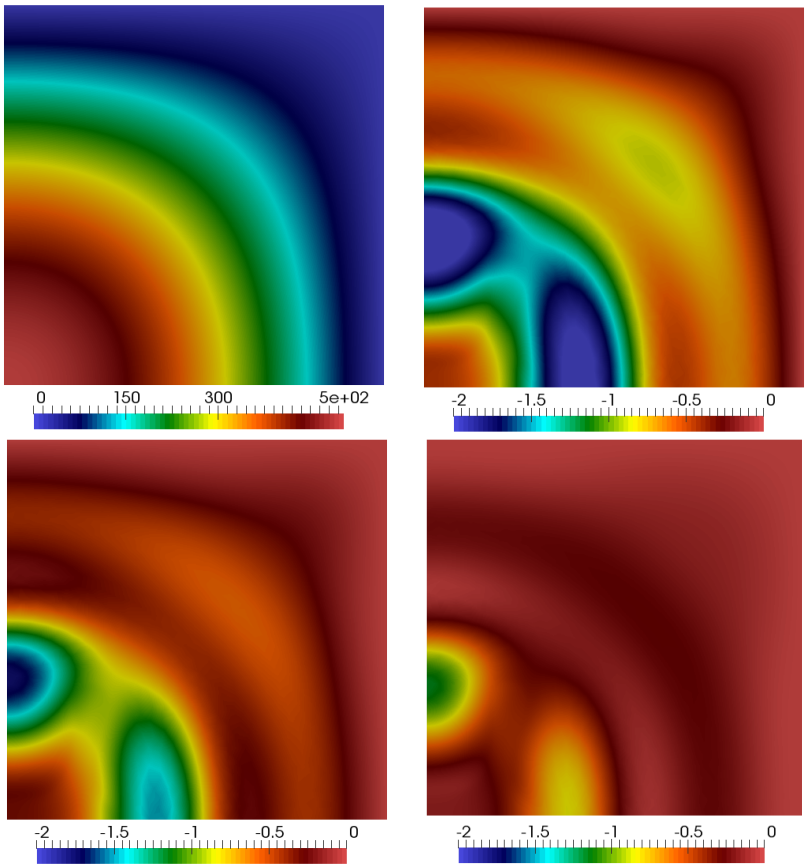

Figure 4.8: Adjoint function for the inverse problem using parameters in Table 4.2 for the initial control $\varphi_h^0 = -1$. From left to right and from top to bottom, iterations $0, 250, 330, 500$.

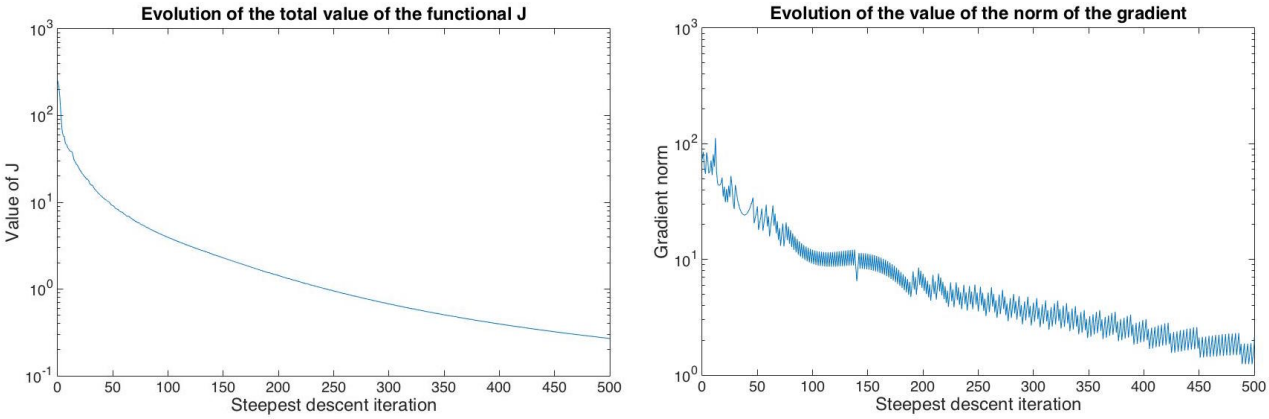

Figure 4.9: Evolution of the cost functional value (left); Evolution of $||g_k||_{L^2(\Omega)}$ (right) for the simulation with parameters setting in Table 4.2 and the initial control $\varphi^0 = -1$ in Ω.

4.4 The case of boundary observation

In section 1.4, we observed that the final mass of the tumor can be obtained from the evolution of nutrient flux across the boundary. In this section, we determine if this observation of the nutrients allow to determine the shape of the tumor inside the domain as well. Indeed, from a medical point of view, this could allow to determine the threshold of the total flux of nutrient needed by the tumor to grow. Hence, by limiting this total flux of nutrients all around the tumor region below this threshold, we would be able to see the tumor shrinking.

Let us define our optimal control problem: using the definition of section 4.1, we define $J(\varphi, \sigma) = J_2(\varphi, \sigma)$. For a target function $\sigma_{\text{target}} \in V_B$ a previous numerical simulation and the operator $C : H^1(\Omega) \to L^2(\Gamma_{\text{out}})$ defined as

$$C(\sigma) = \nabla\sigma|_{\Gamma_{\text{out}}} \cdot \nu, \text{ for } \sigma \in V_B,$$

we define our observation $z_{\text{aim}} = C(\sigma_{\text{target}}) \in L^2(\Gamma_{\text{out}})$. The control function is $u = \varphi \in H^1(\Omega)$ and the state problem is defined by the nutrient equation in (1.21). The derivation of the optimality conditions are done using the same pattern than for the distributed observation's case : the Lagrangian (4.7) is similar, using the definition of the cost functional for boundary observation. At last, only the right hand side of the adjoint equation (4.11) has to be modified.

Unfortunately, for a too large domain Ω the desired solution seems not to be reachable with the chosen initial guess φ^0. To motivate this last assumption, we define

$$\Gamma_2 = \{(12, y) \in \partial\Omega : 0 \leq y \leq 12\}, \quad \Gamma_3 = \{(x, 12) \in \partial\Omega : 0 \leq x \leq 12\}.$$

The nutrient concentration on the simulation using the parameter of the Table 3.1 for $\mathcal{P} = 0.5$ (simulation of Figure 3.3) is targeted at time $T_{\text{obs}} = 12$. In Figure 4.10, we observe that the difference between the flux on the two sides of Γ_{out} is not very different and do not take a form which could give an indication on the shape of the tumor. Hence this system is not controllable and so, this approach to solve the inverse problem is not pursued. The observation of the flux could have been more significant if the tumor cells region had taken a larger part of the domain, that is $|\{x \in \Omega : \varphi(x) = 1\}| \gg |\{x \in \Omega : \varphi(x) = -1\}|$.

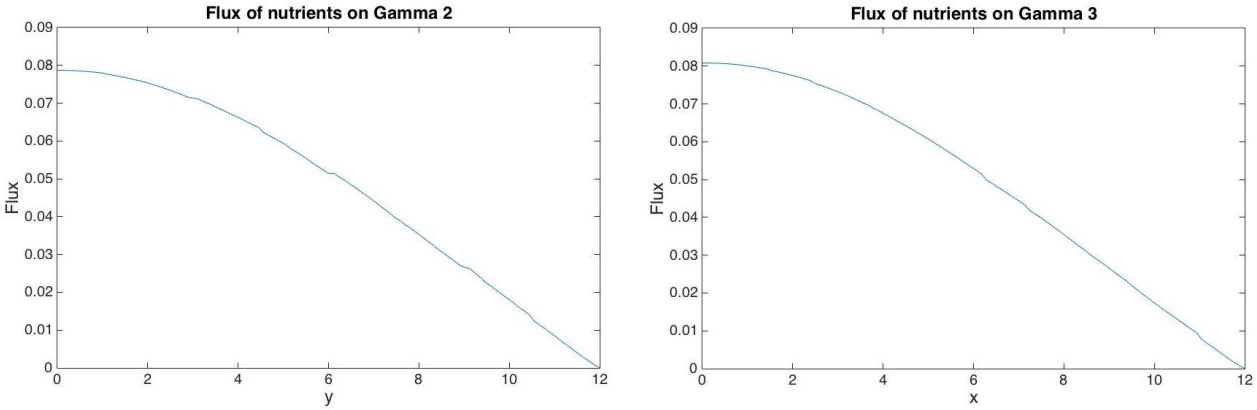

Figure 4.10: Flux of nutrients $\nabla\sigma \cdot \nu$ on Γ_{out} of the simulation in Figure 3.3 at the final time $T_{\text{obs}} = 10$. On the left the flux on Γ_2 ($y \to \partial_x\sigma(12, y)$) and on the right the flux on Γ_3 ($x \to \partial_y\sigma(x, 12)$). We observe that the difference between the two curves is not significant for an inverse problem.

4.5 Further remarks

In the case of the distributed control, even if we considered the entire domain as the observation, we can observe that the control is moving up to the optimal control, whose interface can be observed in Figure 4.2 in the first case and in Figure 4.8 in the second case. Moreover, the adjoint solution decreases until reaching a small maxmum value. Hence there are different possibilities why the inverse problem cannot find an optimal control :

- The Algorithm 4 is not efficient enough, which is likely the case, since it is quite basic. A quasi-Newton method could have been employed instead.
- The problem is not well-posed and/or there exist local minimums to the problem, yielding a control which cannot be optimal.

Conclusions

In this project we have deepen the analysis of a recently proposed tumor growth model. The main steps of developing the model have been shown and a comparison with another system of equations describing a tumour growth model provided by [2] has been made, showing similarities between the two models. The dimensional analysis of the model has been performed in order to highlight the role of relevant parameters of the model. An important link between the flux of nutrients and the mass of the tumour has been observed, and we have been able to verify it numerically. Using a Newton method, together with a semi-discretization in space by use of the finite element method and a time-discretization using a θ-method, we could approximate the Cahn-Allen equation. This allows to use the same numerical approximation method, updated in the case of a system of nonlinear parabolic equations and using Backward Euler method for the discretization in time, to fully discretize the tumour growth' system of equations. We could make several simulations of the tumour growth model, in order to compare the results with [1]. We could take the same conclusions to the ones obtained by the authors of the article concerning the role of the parameters $\mathcal{P}$ and χ_φ, and our results look indeed very similar.

Finally, attempts have been made to solve inverse problems in the form of optimal control problems, in order to recover the tumour-healthy cells function φ from an observation of the nutrient concentration, distributed on the domain or by taking its flux on the boundary. Even if we could not find an optimal control of our problem, the obtained control results are however interesting and encouraging enough to deepen the analysis of such optimal control problems. Two different possible reasons are suggested to explain this lack of accuracy in our results.

Appendix

In order to implement the numerical approximation of the Cahn-Allen equation and of the tumour growth' system of equations by exploiting the redbKIT package [5], we first test the implemention of a simple linear parabolic advection-diffusion-reaction problem. Let $\Omega = (0,1) \times (0,3) \subseteq \mathbb{R}^2$ and $T > 0$. The problem reads as follows :

Find $u : \Omega \times (0,T) \to \mathbb{R}$ such that

$$\begin{aligned} \frac{\partial u}{\partial t} - \nabla \cdot (\mu \nabla u) + \vec{b} \cdot \nabla u + \sigma u &= f \quad \text{in } \Omega \times (0,T) \\ u &= u_D \quad \text{on } \Gamma_D \times (0,T) \\ \mu \frac{\partial u}{\partial \nu} &= g \quad \text{on } \Gamma_N \times (0,T) \\ u &= u_0 \quad \text{in } \Omega \times \{0\}, \end{aligned} \tag{4.14}$$

with $\Gamma_D \cup \Gamma_N = \partial\Omega$, $\Gamma_D \cap \Gamma_N = \emptyset$.

The following constants have been used for the problem (4.14) : $T = 10, \mu = 0.01$ and

$$\vec{b}(t) = \begin{pmatrix} 1 \\ \frac{1}{4}\sin(\omega t) \end{pmatrix}, \; f(\vec{x},t) = f_0 \exp(-\frac{||\vec{x} - \vec{x}_0||_2}{\gamma^2}) h(t),$$

using $\omega = \pi$, $\sigma = 0$, $f_0 = 0.4, \gamma = 0.2$, $x_0 = \left(\frac{1}{4}, \frac{1}{4}\right)$ and

$$h(t) = \begin{cases} t & \text{if } t \leq 1 \\ 1 + \frac{1}{4}\sin(2\pi t) & \text{if } t > 1. \end{cases}$$

A second test has been implemented for $\vec{b}(t) = \begin{pmatrix} 1 \\ \frac{2}{5} + \frac{1}{4}\sin(\omega t) \end{pmatrix}$

We have set homogeneous Dirichlet and Neumann boundary conditions by taking $u_D = g = 0$; also the initial condition has bee set to $u_0 = 0$.

For the numerical approximation, we apply a semi-discretization in space by use of the finite element method and the theta method has been used for the discretization in space. The time step chosen is $dt = 0.05$ and the maximum size of the element is $h = 0.03$, resulting in 5830 elements. The Figure 4.11 shows the results we obtained for the numerical approximation of the equation (4.14)

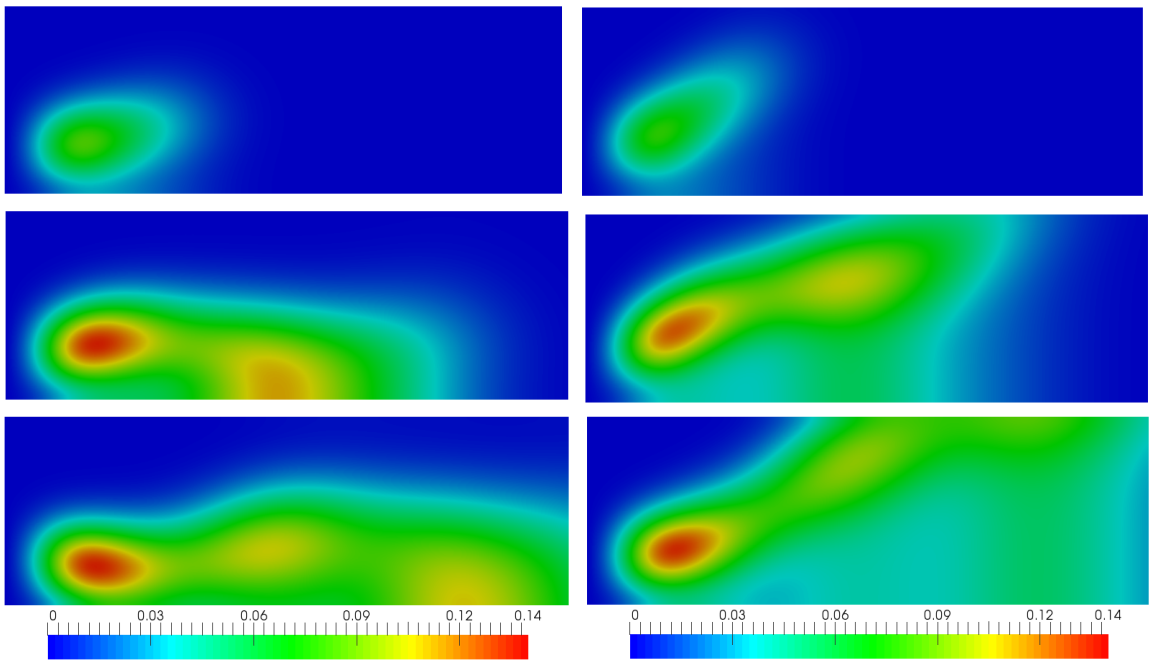

Figure 4.11: Advection-diffusion problems numerical approximations at time $t = 20, 50, 70$ from the top to the bottom. On the left, we used the first definition of the vector $\vec{b}$ for the advection and on the right the second definition.

Bibliography

[1] Harald Garcke, Kei Fong Lam, Emanuel Sitka, and Vanessa Styles. A Cahn–Hilliard–Darcy model for tumour growth with chemotaxis and active transport. *arXiv preprint arXiv:1508.00437*, 2015.

[2] Andrea Hawkins-Daarud, Kristoffer G van der Zee, and J Tinsley Oden. Numerical simulation of a thermodynamically consistent four-species tumor growth model. *International journal for numerical methods in biomedical engineering*, 28(1):3–24, 2012.

[3] Andrés Quiroga, Damián Fernández, Germán Torres, and Cristina Turner. Adjoint method for a tumor invasion Pde-constrained optimization problem in 2D using adaptive finite element method. *Applied Mathematics and Computation*, 270:358–368, 2015.

[4] Amir Gholami, Andreas Mang, and George Biros. An inverse problem formulation for parameter estimation of a reaction–diffusion model of low grade gliomas. *Journal of mathematical biology*, 72(1-2):409–433, 2016.

[5] Alfio Quarteroni, Andrea Manzoni, and Federico Negri. *Reduced Basis Methods for Partial Differential Equations: An Introduction*, volume 92 of *Unitext series*. Springer, 2016.

[6] John W Cahn. On spinodal decomposition. *Acta metallurgica*, 9(9):795–801, 1961.

[7] Paul C Fife. Models for phase separation and their mathematics. *Electronic Journal of Differential Equations*, 48(2000):1–26, 2000.

[8] Luca Dedè and Alfio Quarteroni. Isogeometric analysis for second order partial differential equations on surfaces. *Computer Methods in Applied Mechanics and Engineering*, 284:807–834, 2015.

[9] Jie Shen and Xiaofeng Yang. Numerical approximations of Allen-Cahn and Cahn-Hilliard equations. *DCDS*, 28(4):1669–1691, 2010.

[10] Fernando Concha. *Solid-Liquid Separation in the Mining Industry*. Springer, 2014.

[11] William W Symes. Caam 436 notes: Partial differential equations of mathematical physics, 2012.

[12] Michael Hinze, René Pinnau, Michael Ulbrich, and Stefan Ulbrich. *Optimization with PDE constraints*, volume 23. Springer Science & Business Media, 2008.

[13] Alfio Quarteroni. *Numerical models for differential problems*, volume 2. Springer Science & Business Media, 2010.