Advanced Numerical Treatment of Chemotaxis-driven PDEs in Mathematical Biology

Dissertation
zur Erlangung des Grades eines
Doktors der Naturwissenschaften

Der Fakultät für Mathematik der
Technischen Universität Dortmund
vorgelegt von

Robert Strehl

“*I have deeply regretted that I did not proceed far enough at least to understand something of the great leading principles of mathematics; for men thus endowed seem to have an extra sense.*”

Charles Darwin

Acknowledgements

This thesis is the result of an outstanding academic guidance and delightful non-academic atmosphere at the chair of Applied Mathematics at the TU Dortmund. Without sounding too cheesy, I would like to point out that it is hardly possible to express my deepest appreciation to all colleagues and friends who supported me professionally and personally throughout my PhD studies.

Instead, let me take this opportunity to cordially thank particular persons for their individual contributions towards the realization of this thesis. For those who are not explicitly mentioned below: I highly appreciated your support, thank you!

To begin with, I would like to acknowledge the support of Professor Stefan Turek as my primary supervisor. You provided me with the golden thread that already accompanied my very early work and continued to do so for this thesis. It were your valuable early lectures that motivated me to foster my interest in Applied Mathematics. In this context, I would also like to mention the enormously fruitful advises and dialogues with my senior colleague Dr. Andriy Sokolov. Particularly, but definitely not exclusively, in chemotaxis related subjects, I appreciated your ‘open-door’ policy very much.

I am very grateful to Professor Dmitri Kuzmin for maintaining the communication and irreplaceable consultation throughout the past years. Especially with respect to stabilization techniques, I benefited tremendously from your explanations and our conversations thereover. Also, I am very honored to have established personal contacts with world leading experts in modeling and analyzing of chemotaxis in the field of Mathematical Biology. Utmost interesting insights into analytical aspects of chemotaxis models were provided by Professor Dirk Horstmann, Professor Michael Winkler and Professor Thomas Hillen. It was an enlightening pleasure to comprehend your work, discuss open questions and hear your remarks in order to further the liaising between Numerics and Mathematical Biology.

A distinct appreciation goes to Professor Dmitri Kuzmin, Dr. Andriy Sokolov and Dr. Michael Köster for taking time to proofread this work. Your valuable comments helped me to shape and revise the structure of this thesis. Moreover, I would like to thank Sven Buijssens for valuable consultations with language concerns.

Special thanks are addressed to Christian Kühbacher, Evren Bayraktar and Dr. Abderrahim Ouazzi for sharing both mathematical and non-mathematical thoughts. Thank you!

This thesis was mainly supported by the TU Dortmund via a PhD scholarship (“Besten-

förderung”). As this support was essential for pursuing this independent research, the universitary credit is highly acknowledged.

Last but undoubtedly not least, I would like to thank my family for encouraging me to pursue my work throughout all the past years. Thank you for inspiring me with your confidence.

Dortmund, July 18, 2013

Robert Strehl

Contents

1 Introduction

In the past decades mathematical thinking has been intensively applied on natural life sciences, especially in the field of ecology, physical processes in nature and many biological phenomena in general. The common goal is to map observable features of the real physical and biological processes to an abstract mathematical model and a corresponding discrete numerical formulation in order to gain new insights in the underlying real world objectives by means of reasonable simulations. Moreover, in several cases the mathematical description of the real world system is the only possibility to provide reliable predictive analysis for the underlying process of nature, which results in templates for, e.g., industrial or medical purposes. Many of those mathematical models are described by a (system of) partial differential equation(s) (PDEs). Well established laws of nature and their mathematical counterparts have led to most of the development of suitable PDEs, e.g., heat conduction, fluid dynamics or deformation of solids. However, sometimes the development and understanding of a particular mathematical model can only be tackled by a 'trial-and-error' approach, which is a two-step procedure. In a first step, the simulated results are compared to experimental data and in a second step, these comparisons are used to modify the mathematical model, i.e., the underlying PDE. Hence, simulations of PDEs are of tremendous importance when trying to understand real world processes.

With the recent advances in experimental biology, e.g., in live imaging, scientist are now in a promising position to examine biological processes in unprecedented detail. Although the quality of measurements in experimental biology is unfortunately still not as well developed compared to the precision of measuring tools in the field of (mechanical) engineering, the experimental investigation of biological processes lately experiences a huge wealth of breaking assets. In the course of these biological accomplishments, the advent and development of *Mathematical Biology* as a novel interdisciplinary research branch emerged and provides a new perspective on biological phenomena. The mathematical formulation of biological processes and their precise analysis allows biological experimentalists to verify results retrospectively and develop prospective conjectures. However, a pure theoretical analysis of mathematical models is crucially limited. Particularly for recent models that describe multi-dimensional signaling pathways incorporating several entities, their complexity cannot be fully captured and analyzed with tools provided by theoretical analysts. The urgent need of a detailed study of complex models drove numerical analysts to consider biologically motivated systems, the foundation of *Computational Biology* as a particular discipline of

Mathematical Biology. The tremendous potential of these interdisciplinary research branches in today's science is nicely formulated in Cohen's essay *Mathematics is biology's next microscope, only better; biology is mathematics' next physics, only better* [19].

This is the field of research in which the present thesis can be understood. In essence, we consider the numerical aspects, in terms of a finite element (FE) discretization, of simulating PDEs that were introduced to model a certain biochemical process, which can be observed in many living organisms, termed *chemotaxis*. What is chemotaxis and what is the motivation behind studying the numerical properties of this phenomenon?

1.1. The biochemical concept of chemotaxis

Let us begin with the literal translation and the description of the underlying biochemical process. The word chemotaxis is originally deduced from the Greek. Loosely speaking the suffix '*taxis*' can be understood as motion or migration and '*chemo*' classifies the reason for the taxis. Thus chemotaxis describes the phenomenon of directing the migration according to a gradient of some chemical substance. The literature differentiates positive and negative chemotaxis, which simply refers to the direction upward or downward the chemical gradient. In these cases, the chemical itself is called chemoattractant or chemorepellent, respectively. The character of the reaction on the chemical gradient is called chemosensitivity, namely a large or small chemosensitivity allows a rapid or slow reaction in terms of migration. The ability of organisms to sense and direct their motion towards (or away from) a chemical gradient is an essential property.

Exemplary let us provide three paradigms in which chemotaxis plays a vitally important role. First of all, in the stage of early development of higher organisms, e.g., mammals, chemotaxis allows the mobilization and organization of stem cells that eventually leads to differentiation into, e.g., highly specialized bone, neuronal or blood cells [18, 32, 72]. A second common example is the detection and localization of food sources or prey recognition. For instance, nutrients or prey serve as a source of chemical signals (either directly or indirectly via production/secretion of chemoattractants) for simplex life forms like, e.g., bacteria, slime molds or nematodes [1, 56, 111]. The last example of how chemotaxis provides a necessary ability for organisms to react in their environment is the immune system [68, 76]. Let us focus on the human immune system. Once an inflammation arises, our immune system counters this invasion of toxic substances or harming bacteria by releasing leukocytes. The path that leukocytes take to localize the site of infection is determined by traces of chemokines that have been released by resident cells at the affected tissue. These chemokines act as attractive chemicals, hence guiding the leukocytes to the origin of the inflammation.

Biologically, chemotaxis is a process which involves a complex network of intracellular chemical signaling pathways that is activated by chemical-receptor bindings. We recommend the corresponding chapter in the book of Alberts *et al.* [3, Chapter 15] for a detailed reference for chemical signaling pathways. Here, we will only briefly recapitulate the main concept of chemotaxis.

The sensory chemical (ligand) receptors that activate the complex signaling cascade mainly happen to be located at the cell's membrane (due to hydrophilic chemical molecules). If these receptors are active and bind a corresponding (extracellular) ligand, they allow an intracellular signaling cascade to be activated, see Figure 1.1. The detailed mechanism of which regulative entities are exactly involved and how they interact with each other are highly depending on the underlying organism. Even for bacterial chemotaxis this has only marginally been explored up to the present. In [99] we read

> "*Of the estimated many millions of bacterial species which are assumed to exist in nature, less than 100 have been studied in detail.*"

The common resulting effect of these different signaling pathways is the simple (re-)mobilization of a motor for the motility, e.g., adjustment of the flagella rotation in bacteria. It was observed that a clockwise (CW) rotation of the flagella motor causes the flagella to fly apart, whereas a counter-clockwise (CCW) rotation results in a bundling of the flagella. Afterwards, the receptors adapt to the new extracellular concentration of ligands (e.g., by temporal methylation of the receptor) in order to allow further gradient detections. Figure 1.1 depicts this process very roughly. The switching in the flagella motor rotation characterizes the bacterial motion. We can consider the bacterial chemotaxis-biased random walk to be split into two states. In one state the bacteria *tumbles* due to CW rotation of its flagella. In this state the bacteria re-orientates by selecting a random new walking direction. In a second state a counter-clockwise rotation of the flagella drives the bacteria to *run* in the previously selected direction. The result is a so-called *run-and-tumble walk*, where the time between two turns (two tumbling phases) depends on the detected gradient of the chemical. A schematic sketch of this mechanism is depicted in Figure 1.2.

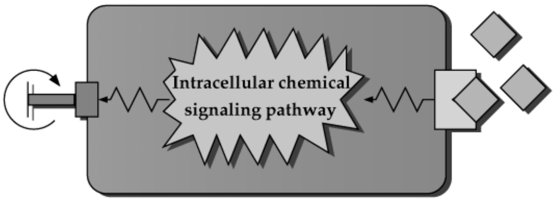
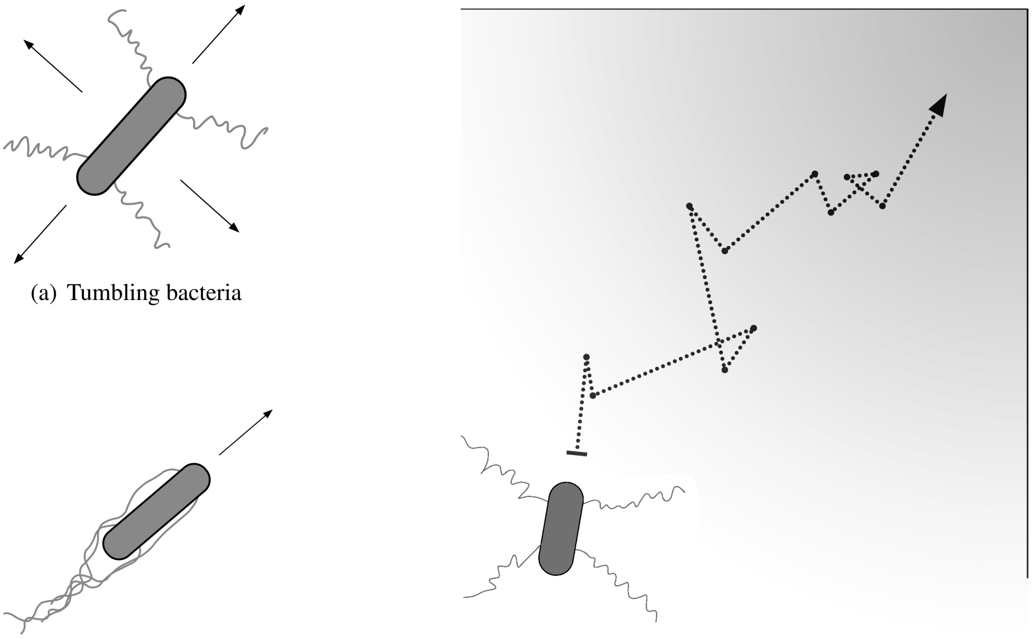


Figure 1.1: A rough sketch of a chemotaxis-induced signaling pathway of bacteria. The chemical ligands on the right activate the receptor, which initiates an intracellular signaling cascade. The result of this signaling pathway is the control of the flagella motor switching (from CW to CCW), which is depicted on the left.

A short remark regarding the detection of chemical gradients seems indicated. There is a notable differentiation how chemical gradients can be identified during the processing of the signaling pathways. Because of the simple fact of their extremely small size, bacteria sense chemical gradients in a different manner than larger organisms, e.g., the experimentally well investigated slime mold *Dictyostelium discoideum*. While slime molds (some mm in size) can measure the gradient directly by sensing the non-uniformly distributed active ligand-receptor bindings along the membrane, bacteria (of only a few μm in size) calculate the gradient by comparing the concentration along a walking path, since a non-uniform chemical concentration on the bacterial scale is already perturbed by the noise of the ubiquitous Brownian motion and hence cannot be detected directly.

1.2. The motivation for treating chemotaxis models

Now that we have classified the biochemical process on which the PDEs under consideration are based on, let us consider the motivation behind the numerical investigation of such models. The first PDE system that described a chemotaxis-driven population development goes back in time to the early 1970's. It was introduced by Keller and Segel [52] and was motivated by experiments



(a) Tumbling bacteria

(b) Running bacteria

(c) Chemotaxis-biased random walk

Figure 1.2: Schematic illustration of (bacterial) chemotaxis. **(a)** state of tumbling (CW flagella rotation), **(b)** state of running (CCW flagella rotation), In **(c)** we sketched an exemplary chemotaxis path of a bacteria, where the upper right corner is the location of an attracting chemical. Note that the runs upwards the gradient are longer than the downwards runs.

of Bonner [11] with the slime mold *Dictyostelium discoideum*, or in short '*dicty*', as it is often tenderly called by researchers. Moreover, encouraged by the research of Adler [1], Keller and Segel extended their model to chemotaxis in bacteria. Both models consist of one PDE for the cell density, commonly denoted by u, complemented by a second PDE for the chemoattractant, usually referred to as v.

Since the publishing date of the chemotaxis model is not as long ago as the publishing of the Navier-Stokes (1827/1845) or even Euler (1755) equations, it provides us a numerical field where so much can be discovered, investigated and postulated. Indeed, since the theoretical aspects of many chemotaxis models are not fully understood yet, an extensive numerical treatment of those models is highly demanded in order to gain new insights in both theory, e.g., in terms of uniqueness or boundedness of solutions, and practical applications, e.g., prediction of cell/chemical distribution for experimental assays or even clinical studies. Already for the biologically well studied mechanism of bacterial chemotaxis (cf. the brief introduction given before), Alberts *et al.* remarked the huge potential of further numerical investigations of a corresponding model for the chemical signaling pathways involved. In their book, we read [3, Chapter 15, p. 944]:

> "*Even in this relatively simple signaling network, however, computer-based simulations are required to comprehend how the system works as an integrated network. Cell signaling will provide an especially rich area of investigation for a new generation of computational biologists, as the network properties of these pathways are not understandable without powerful computational tools.*"

We will not postulate possible applications for numerical frameworks of chemotaxis PDE models in more detail since they highly depend on the governing model. Instead, let us mention the reason why chemotaxis models cannot be treated via some standard numerical scheme in the sense of a ‘black-box’ solver, rendering the present investigation redundant.

The character of chemotaxis-driven PDEs is the agglomeration of cell concentrations in limited space with possibly sharp interfaces. Moreover, the speed of agglomeration can vary in different time-scales. Standard numerical schemes are not able to cover these characteristics within suitable CPU and memory bounds. In this context, we can confer to the treatment of Navier-Stokes equations at large Reynolds numbers. After discretization, it is well known that the temporal resolution highly depends on the spatial mesh-size (rf. CFL condition). When choosing a bad resolution, e.g., to save memory and/or CPU expenses, the numerical simulation provides very poor results, if at all. A similar behavior, although arising from a different subject, can be observed for chemotaxis-dominated PDEs. A large Reynolds number corresponds to a large chemosensitivity. Moreover, because of the composition of the chemosensitivity as a function which usually depends on the chemical substance, a positive feedback, in terms of

$$\text{agglomeration of } u \quad \to \quad \text{increase of } v \quad \to \quad \text{even stronger agglomeration of } u\,,$$

enhances the agglomeration and possibly even leads to a locally unbounded increase of cell concentration. These properties already necessitate highly specialized numerical solvers that allow an efficient computation at a highly accurate resolution.

1.3. Thesis outline

Let us close this introductory part by sketching the outline of this thesis. To begin with, we will provide some notations and preliminaries for understanding the analytical results of certain PDE models and their numerical treatment via the finite element methods (FEM) that we will encounter throughout this thesis, Chapter 2. Subsequently, we will summarize the derivation of a classical model of chemotaxis and discuss the key-players of such models, so that the reader has a convenient introduction to this youthful topic of chemotaxis models, Chapter 3. The next chapter is devoted to different discretization strategies for a more general model of chemotaxis, Chapter 4. Moreover, this chapter will deal with a stabilization technique that promises to remedy common drawbacks of standard discretization schemes. The reader is kindly advised to carefully examine this chapter, since therein we provide the discretization framework for all of the numerical studies that will be subject of Chapter 5. From the practical point of view Chapter 4 and Chapter 5 represent the main parts of this thesis, since it will provide all the numerical results that are obtained during the numerical analysis. We will compare the different discretization schemes, try to quantify their results and seek for the most reliable, flexible and efficient solver for models of chemotaxis. After validating and identifying robust solvers, we will apply them on chemotaxis models that are recently discussed in the literature. The last chapter closes this thesis by providing a brief summary, encouraging some further discussions and proposing fields of further investigations, Chapter 6.

2
Preliminaries

This chapter deals with the general notation and symbols employed throughout this thesis and the main theoretical background of chemotaxis PDEs which will be the focus in this work. While the first section will provide the reader with an overview of a consistent nomenclature, the second part will give a brief summary of the state of the art from the analytical point of view and will offer explanations of terms and definitions which will be used throughout this thesis.

2.1. General notation

Table 2.1 provides an overview of most common symbols for the remainder of this thesis. Upcoming new terms and symbols will be introduced in the corresponding context in order to keep the sections self-contained and ease the understanding of particular symbols for the reader.

2.1.1. Notations for the continuous space

Let us denote by $\Omega \subset \mathbb{R}^{\mathrm{dim}}$ ($\mathrm{dim} = 1,2,3$) the computational spatial bounded domain with boundary $\partial\Omega$ and spatial variables denoted by $\mathbf{x} = (x_1, x_2, \dots) \in \Omega$. Furthermore let $I = [0, t_{end}] \subset \mathbb{R}$ with $t_{end} > 0$ be a time interval we are looking at with temporal variable $t \in I$. Hence, the time-space cylinder $I \times \Omega \ni (t, \mathbf{x})$ describes the entire domain which has to be discretized via a suitable FEM ansatz.

Let $u, v : I \times \Omega \to \mathbb{R}$ be certain ‘sufficiently smooth’ scalar functions. We will use standard notations for the gradient and the Laplacian working on the spatial variables, i.e.,

$$\nabla u \quad = \quad (\partial_{x_1} u, \dots, \partial_{x_{\mathrm{dim}}} u)^T$$

and

$$\Delta u \quad = \quad \nabla \cdot \nabla u \quad = \quad \partial_{x_1 x_1} u + \cdots + \partial_{x_{\mathrm{dim}} x_{\mathrm{dim}}} u \,.$$

Greek letters $(\alpha, \beta, \gamma, \dots)$	scalar valued variables, e.g., $\alpha \in \mathbb{R}$
Greek letters (φ, ψ)	FE-test/-trial functions
bold-faced letters $(\mathbf{u}, \mathbf{v}, \mathbf{w}, \dots)$	(FE-coefficient) vectors
calligraphic upper case letters $(\mathcal{A}, \mathcal{B}, \dots)$	block matrices
dim	underlying spatial dimension
$\mathbf{x}$	spatial variable, i.e., $\mathbf{x} = (x_1, x_2, \dots) \in \mathbb{R}^{\text{dim}}$
t	temporal variable
$\partial_\bullet$	common abbreviation for partial derivatives, e.g., $\partial_t = \partial u / \partial t$
$I\ (= [0, t_{end}])$	underlying temporal interval
δt	time step width
n	index for the time level, i.e., $\mathbf{u}^n = \mathbf{u}(n\,\delta t)$
Ω, Ω_h	original underlying open spatial domain and its discrete counterpart
δh	spatial (uniform) mesh size
N	number of spatial degrees of freedom
V, V_h	original space of the continuous solutions and its conforming discrete counterpart
φ_i	nodal basis functions, $\varphi_i \in V_h$, for $i = 1, \dots, \mathrm{N}$
h-subscripted letters $(u_h, v_h, w_h, \dots)$	FE function, i.e., $u_h = \sum_i u_i \varphi_i \in V_h$

Table 2.1: Overview of the general notation.

Moreover, throughout this thesis we will write

$$\nabla \cdot \Big(u \nabla v\Big) \quad = \quad \begin{pmatrix} \partial_{x_1} \\ \vdots \\ \partial_{x_{\text{dim}}} \end{pmatrix} \cdot \begin{pmatrix} u\,\partial_{x_1} v \\ \vdots \\ u\,\partial_{x_{\text{dim}}} v \end{pmatrix} \quad = \quad \partial_{x_1}\Big(u\,\partial_{x_1} v\Big) + \dots + \partial_{x_{\text{dim}}}\Big(u\,\partial_{x_{\text{dim}}} v\Big),$$

that is, the product of a function and a gradient $u\nabla v$ is meant component-wise. Since our work does not focus on the functional analysis aspects of the chemotaxis-driven PDEs, we will not go into detail about the functional spaces in which the solutions u and v of our governing PDEs are to be found. Hence, for the remainder of this thesis, let us assume our solutions to be in some reasonable space V.

2.1.2. Notations for the discretized space

The (full) discretization of the underlying PDE is subject of Chapter 4. Before turning to this task, let us introduce some basic notations in this paragraph. We emphasize that this thesis does not focus on elaborate mesh discretization techniques, such as adaptive time stepping and h-, p- or r-refinement of the spatial discretization. Although the author is aware that these techniques have great potential to enhance the numerical algorithms $-$ a point which will be discussed in corresponding chapters later on $-$ the author skipped their implementations in order no to overload the scope of this present work. Keeping that in mind we can define the discretization in a more convenient manner. We will follow the ideas of the *method of lines* where we first apply a spatial discretization of the governing PDE with FEM, resulting in a system of ordinary differential equations, also termed the *semi-discretized* formulation. In a second step we employ the discretization in time with simple finite differences, leading to the so-called *fully-discretized* formulation. This final system of (possibly nonlinear) equations can then be processed further by a suitable numerical scheme providing approximate solutions.

Let $\Omega_h \subset \mathbb{R}^{\text{dim}}$ be a conforming triangulation of the domain Ω with quadrilateral/hexahedral cells and $p_j \in \mathbb{R}^{\text{dim}}$, $j = 1, \ldots, \mathrm{N}$ denote the corresponding vertices of this triangulation, where N is the number of spatial degrees of freedom throughout this thesis. Furthermore, let $\delta h \ll diam(\Omega_h)$ denote the uniform spatial mesh size. Precisely speaking, the uniformity of the spatial mesh size highly depends on the underlying computational domain Ω_h. In this work, we restrict ourselves to have the same discretization for the test- and trial-space and use bilinear/trilinear conforming quadrilateral/hexahedral finite elements. The corresponding element is commonly denoted by $\mathcal{Q}_1$. The advantage of such an element is its convenient property that the degrees of freedom can be prescribed as being the function values at the corner vertices.

With this setup a simple quadrilateral/hexahedral domain, e.g., the unit square/cube $\Omega = (0,1)^{\text{dim}}$, $\text{dim} = 2,3$, can be uniformly discretized. On the other hand, discretizing a circular domain cannot be accomplished by a uniform δh. In this situation we might only focus on the maximal size $\delta h_{\max}$ or we might adjust the coarse grid to be almost uniformly discretized.

Let us denote the resulting conforming discrete space as $V_h \subset V$ and its finite dimensional set of basis functions as $\{\varphi_1, \ldots, \varphi_\mathrm{N}\}$. As it is very convenient for $\mathcal{Q}_1$ elements, we will use nodal basis functions. These functions are defined as follows

$$\varphi_i(p_j) \quad = \quad \delta_{ij}\,, \qquad \text{for all } 1 \leq i,j \leq \mathrm{N}\,,$$

where δ_{ij} denotes the Kronecker-delta.

The FE-representation $u_h(t,\mathbf{x})$ of a continuous in time and space function $u(t,\mathbf{x})$ is now provided by

$$u_h(t,\mathbf{x}) \quad = \quad \sum_i u_i(t)\,\varphi_i(\mathbf{x})\,,$$

where $\mathbf{u}(t) = (u_1(t), \ldots, u_\mathrm{N}(t))^T$ is the FE coefficient vector. For the remainder of this work, we will use this coefficient vector $\mathbf{u}(t)$ when referring to the FE-representation of the solution.

In the course of the discretization in time, we introduce a uniform time stepping with $n_{\max}$ time steps and $\delta t = t_{end}/n_{\max}$ that provides a discrete time interval. The FE solution $u_h(t,\mathbf{x})$, or in convenient formulation $\mathbf{u}(t)$, will be tracked at these distinct time instances and we will write $\mathbf{u}^n = \mathbf{u}(n\,\delta t)$.

2.2. Historical notes of chemotaxis PDEs

Before we focus on definitions and theorems on which we will rely in subsequent chapters, let us briefly recapitulate the historical accomplishments and researches in the context of chemotaxis PDEs.

As a leading remark we like to note that the following historical background was mainly extracted from the survey paper of Horstmann [46]. If missing some detailed information the reader is kindly advised to be referred to this literature and its references therein.

In 1970's Keller and Segel [52] were the first who developed a mathematical PDE model in order to describe the development of a chemotaxis-driven population, the slime mold *Dictyostelium discoideum*. Their idea was it to establish a model that fits well to the experimental data observed by, e.g., Bonner [11]. In its simplest formulation their model reads

$$\left\{\begin{array}{lcll} \partial_t u(t,\mathbf{x}) & = & \nabla\cdot\left(\nabla u(t,\mathbf{x}) - u(t,\mathbf{x})\chi\nabla v(t,\mathbf{x})\right), & \text{for } (t,\mathbf{x})\in I\times\Omega, \\ \partial_t v(t,\mathbf{x}) & = & \Delta v(t,\mathbf{x}) + u(t,\mathbf{x}) - v(t,\mathbf{x}), & \text{for } (t,\mathbf{x})\in I\times\Omega. \end{array}\right. \tag{2.2.1}$$

Herein $u(t,\mathbf{x}), v(t,\mathbf{x})$ denote the concentration of the cell population and chemoactive substance, respectively. Unless we define anything to the contrary, the coefficient χ is a certain positive scalar constants. In the first equation we face a diffusive and a chemotactical flux, which can also be considered as being of antidiffusive character (note the minus sign in front of the second term on the right-hand side). The second equation consists of (standard) diffusion-reaction terms. The reaction term represents a natural depletion of the chemoactive substance and its production by the cells. We defer a detailed description of the equation and its derivation to the upcoming Chapter 3.

The model (2.2.1) is commonly complemented by initial conditions of kind

$$u(0,\mathbf{x}) = u_0(\mathbf{x}), \quad v(0,\mathbf{x}) = v_0(\mathbf{x}), \quad \text{for all } \mathbf{x}\in\Omega \tag{2.2.2}$$

and Neumann boundary conditions of kind

$$\mathbf{n}\cdot\nabla u(t,\mathbf{x}) = 0, \quad \mathbf{n}\cdot\nabla v(t,\mathbf{x}) = 0, \quad \text{for all } (t,\mathbf{x})\in I\times\partial\Omega, \tag{2.2.3}$$

where $\mathbf{n}$ denotes the unit outward normal to $\partial\Omega$.

Keller and Segel also obtained first analytic results concerning the (linear) stability of uniform solutions which can be summarized in the following

Theorem 2.1 ([52]) *A stationary uniform solution (u^*, v^*) (in case of (2.2.1) we even have $u^* = v^*$) is unstable if $\chi u^* > 1$.*

Loosely speaking, here and hereafter we understand stability of a solution as

Definition 2.1 (following [52]) *An arbitrary (possibly nonstationary) solution (u^*, v^*) is considered stable if any small initial perturbation (fluctuation) does not grow in time and eventually pollutes the solution. A solution is considered unstable if there is at least one perturbation that finally pollutes the solution.*

For example, if, despite the initial fluctuations, the solution is exactly recovered as time evolves, we end up with *asymptotic stability*. In this case the fluctuations diminish in time. If the pollution, however, only not grows in time, say the perturbed solution's orbit remains in a neighborhood of the exact solution's orbit, then we obtain *Lyapunov stability*. We note that Lyapunov stability is indeed strictly weaker than asymptotic stability. For a brief recapitulation of an exemplary linear stability analysis we refer to the appendix A.

Returning to the result of Keller and Segel, their interpretation of their findings is that there is a critical mass (dependent on χ) that drives the cell population to overcome the diffusive character of the model. Together with the mass conservation (in terms of the L^1 norm) of the cell population and more elaborated nonlinear analysis Nanjundiah [83] conjectured that

> "*[...] the end-result of the instability is aggregation of the cells at one or more points.*"

Moreover, because of the accelerating-typed instability, caused by the positive feedback of cell agglomeration and chemical production, he concluded that these aggregates eventually form δ-singularities. About seven years later, Childress and Percus [17] reconsidered the (space-independent) conjecture of Nanjundiah and came to the conclusion that δ-singularities, in fact, cannot occur in the one-dimensional (1D) space but may emerge in higher dimensions. In the following we refer to the formation of singularities as *blow-up*.

Definition 2.2 (following [46]) *The solution (u,v) for a chemotaxis PDE blows up (is a blowing-up solution) if $||u(t,\cdot)||_{L^\infty}$ becomes unbounded in finite or infinite time,*

$$||u(t,\cdot)||_{L^\infty} \to \infty \quad \textit{as} \quad t \to t_{end} \textit{ or } t \to \infty, \textit{ respectively}.$$

The interesting question whether or not there exists (stable) non-homogeneous solutions which do not blow up are left open up to this point. The literature distinguishes between the stationary, i.e., $\partial_t u = 0 = \partial_t v$, and the time dependent problem formulation.
The stationary variant was further studied by Schaaf [91] via bifurcation methods. She considered a more general system of (2.2.1)–(2.2.3) and examined bifurcation points also for inhomogeneous solutions.

It took more than ten years to establish the next landmarks in this subject. It was Biler [9] who proved the existence of radial stationary non-homogeneous solutions − the reason to study the radial case, besides its greater simplicity, can be found in Diaz and Nagai [23], who proposed the control of arbitrary domains via symmetrization. Moreover, Biler continued considerations, e.g., about global existence and blow-up depending on parameters, based on a parabolic-elliptic variant of (2.2.1) with different boundary conditions. At this time many groups of researchers were attracted to study all kinds of variants of (2.2.1)–(2.2.3). Subsequently the non-radial case was then examined among other independent researchers by Horstmann [45] for the 2D case. Further on, increasingly more researchers revealed particular aspects of Keller-Segel's equations and it became very tedious to keep track of them in order. Instead of quoting every single contribution we rather like to sketch the main interesting results that will be of particular concern in this work. Several authors studied the asymptotic behavior with the support of a certain Lyapunov functional, e.g., as introduced in [33],

$$E(u,v) = \int\limits_{\Omega} \frac{1}{2\chi}\left(|\nabla v|^2 + v^2\right) + u \log u - u\,v \;\mathrm{d}\mathbf{x}\,.$$

For a short introduction of basic concepts of Lyapunov functionals, we defer the interested reader to the appendix B.

Dimension	Reference	Result
dim = 1	[84]	The 1D-system (2.2.1)–(2.2.3) admits (possibly stationary) globally bounded solutions for reasonable initial data.
dim = 2	[82]	For $4\pi > \chi\|\|u_0\|\|_{L^1}$ the 2D-system (2.2.1)–(2.2.3) admits a bounded Lyapunov functional and hence has a global solution.
	[33]	If, for $t \to \infty$, the 2D-solution admits $E(u,v) \to -\infty$, then it follows $\|\|u\|\| \to \infty$. In other words, if the Lyapunov functional can not be bounded globally in time, then the solution blows up.
	[33], [45]	For $4\pi > \chi\|\|u_0\|\|_{L^1}$ the global solution of the 2D-system (2.2.1)–(2.2.3) converges to a (possible non-homogeneous) steady state.
	[49]	For $4\pi < \chi\|\|u_0\|\|_{L^1}$ (necessary condition) there exist (possibly non-symmetric) solutions of the 2D-system (2.2.1)–(2.2.3) that blow up.
	[31]	Every (bounded) global solution of the nonstationary 2D-problem eventually converges to a stationary solution.
	[48]	For $4\pi > \chi\|\|u_0\|\|_{L^1}$ the 2D-system (2.2.1)–(2.2.3) admits only the unique trivial stationary solution $(u^*,v^*) = (\|\|u_0\|\|_{L^1}/\|\Omega\|, \|\|u_0\|\|_{L^1}/\|\Omega\|)$.
dim = 3	[106]	For $q > 3/2$ and $p > 3$ there exists ε such that if $\|\|u_0\|\|_{L^q} < \varepsilon$ and $\|\|v_0\|\|_{L^p} < \varepsilon$ then the solution exists globally and converges to the homogeneous equilibrium $(\|\|u_0\|\|_{L^1}/\|\Omega\|, \|\|v_0\|\|_{L^1}/\|\Omega\|)$.
	[106]	For arbitrary initial mass $\|\|u_0\|\|_{L^1}$ there exist blowing-up solutions if we consider $\Omega \subset \mathbb{R}^3$ being a ball.

Table 2.2: Overview of valuable theoretical results for the chemotaxis model (2.2.1). Note that for dim = 2 we list in chronological order and some results became obsolete.

Now the major results can be captured in the following Table 2.2.

We observe that many results are due to very recent work. Moreover, beside the lack of comprehensive steady state and stability investigations, up to our current knowledge, we recognize that the three-dimensional (3D) case is not yet covered in detail, compared to the one-dimensional (1D) and 2D counterparts. This stresses the mathematical challenge even for a 'rather simple' chemotaxis model (2.2.1)–(2.2.3), particularly from the theoretical point of view.

In order to motivate the investigation of simple-looking chemotaxis models from the theoretical point of view, we provide some open questions, contributed by personal communications with Horstmann and Winkler:

Blow-up condition (2D) From [49] we know a necessary condition for a blow-up in two dimensions, is there also additionally a sufficient condition on the initial data?

Blow-up condition (3D) Concerning the 3D case, is there a critical mass that may lead to a blow-up, such as in the case of lower dimensions? In other words is there a $K > 0$, such that $||u_0||_{L^{3/2}} \lesseqgtr K$ leads to globally bounded or blowing-up solutions, respectively?

Blow-up points There are many results for radial symmetric blow-ups. How does the situation change for a non-symmetric blowing-up solution? Are there initial conditions that drive the solution to multiple blow-up points (at the boundary or inside the domain)?

Blow-up asymptotics In the case of a blowing-up solution. How can we characterize whether or not the blow-up time is finite or infinite? Is there a way to classify the blowing-up behavior near the blowing-up time? Can we anticipate the number (and localization) of blow-up points in the case of disjointed initial data? What happens after the blow-up time?

Patterns (3D) What is the possible set of steady states in the 3D case? Yet we only know (cf. Table 2.2) that sufficiently small initial data lead to homogeneous steady states. There is no statement about possible non-homogeneous steady states.

Kinetic model (3D) When complementing the minimal model with a kinetic term for the u equation, say $\kappa u(1-u)$, cf. (3.3.1) in Section 3.3, the 2D model provides reasonable and bounded solutions [85]. However, the boundedness in 3D can yet only be proven for sufficiently large κ [107]. What is the solution's behavior for very small κ? Does there even exist a blow-up in such situations?

Obviously for more complex models – we will introduce interesting extensions to model (2.2.1) in the upcoming Chapter 3 – this given pool of theoretical questions will certainly be enriched. However, for the sake of clarity and in order not to blur the scope of this work, we restricted our historical notes to the minimal model (2.2.1).

3 Modeling chemotaxis

This chapter strives to recapitulate the derivation for the minimal model of chemotaxis. Moreover we will present some biologically motivated extensions to the model which allow for a better description of in-vitro and in-vivo observed features of chemotactic signal processing. The first section of this chapter will encourage some stochastic and physical thoughts in order to understand the main principles of the derivations. The second part will introduce some practical scalings that simplify the underlying equations. The third section, namely the biological motivated extensions of the basic model, will apply some basic knowledge of chemical reactions.

3.1. Derivation of the minimal model of chemotaxis

Basically there are two methods at hand to derive a suitable model which represents chemotactical movement of cells u induced by a certain chemical v (cf. [46, 78, 88]). On the one hand there is the microscopic approach, which basically models an entity-entity system, which can also be understood as discretizing single cells and chemical molecules, that can be derived via the limit case of a stochastic ansatz. On the other hand, the macroscopic view derives the system via Fick's law applied to the well known mass conservation law. This alternative considers the cell-chemical system in a more continuous fashion, i.e., the distribution/density of cells and chemical substances (and the corresponding fluxes) are taken into account.

To date, these two points of views or scales, which they are also commonly termed, are intensively discussed in the community. In certain cases, the modeling of cells on the macroscopic scale is inconvenient, since the total number of modeled cells are rather low, say in the range of 10,000[1]. In this scenario, the contribution of single cell dynamics will have a significant influence on the development of the entire cell compound, rendering an approach which cannot identify single cells impractical. However, for modeling the chemicals, the situation changes, simply because of the huge amount of chemical molecules under consideration. In the current belief of the community, a multi-scale approach will be the best fitting modeling framework that can capture the

[1] In contrast, the experimental assays conducted by Adler [1] that initiated the development of the Keller–Segel model involved millions of cells of *E. coli*.

interactions across cell-chemical scales. The interplay of such multi-scale processes is well known experimentally. But only recently, with the advent of promising experimental biology techniques and computational technologies, the development and the numerical consideration of such models can now be fully addressed.

3.1.1. Microscopic derivation

As mentioned above the microscopic scale shines a light on the derivation in a stochastic manner. To this end, let us first introduce the scenario which we want to model. Following Othmer and Stevens in [86], we introduce an equidistant one-dimensional continuous-time, discrete space bias random-walk approach. We consider a scenario where the orientation of an entity u (here cells) is influenced by the attractive characteristic (of the gradient) of a particular chemical substance v binding on certain cell receptors at the membrane and resulting in a biased random-walk. Let the one-dimensional domain $\Omega = [0,l]$ be discretized with the uniform step size $\delta h = l/i_{\max}$, resulting in the discrete variables $i = i\,\delta h$ for $i = 0,\ldots,i_{\max}$. Furthermore, $u_i(t)$ denotes the probability that an entity (here a single cell) is situated at the $(t, i\,\delta h)$ coordinate, where $t \in I = [0,t_{end}]$. Initially, say at $t = 0$, the cell starts at $i = 0$. Now a quasi-Markovian process leads to

$$\partial_t u_i(t) \quad = \quad T^+(v_{i-1})\,u_{i-1}(t) + T^-(v_{i+1})\,u_{i+1}(t) - \big[T^+(v_i) + T^-(v_i)\big]\,u_i(t)\,, \qquad (3.1.1)$$

where $T^{\pm}(v_j)$ reflects the probability (controlled by v) of the u-transition from the discrete location j to the right $(+)$ or left $(-)$. Note that, strictly speaking, the process described here cannot be considered Markovian if the transition probabilities $T^{\pm}(v_j)$ depend on u, which will be the case as we see later on, cf. (3.1.16). A remedy would be to consider the extended state space (u,v), which we skip for brevity reasons. Now, in order to take into account chemotaxis, we have to define suitable transition probabilities $T^{\pm}(v_j)$. Othmer and Stevens proposed some choices among which we stress two simplifications.

First of all it might already be clear from (3.1.1) that the choice $T^+(v_j) = T(v_j) = T^-(v_j)$ results in a random walk. This is clear by identifying the right-hand side of (3.1.1) by the corresponding second-order Taylor expansion

$$\partial_{xx}\Big(T(v_i)\,u_i(t)\Big) \quad = \quad \big[T(v_{i+1})\,u_{i+1}(t) - 2\,T(v_i)\,u_i(t) + T(v_{i-1})\,u_{i-1}(t)\big]/\delta h^2 + \mathcal{O}(\delta h^2)\,,$$

which transforms (3.1.1) into

$$\partial_t u_i(t) = \delta h^2\,\partial_{xx}\Big(T(v_i)\,u_i(t)\Big)\,. \qquad (3.1.2)$$

Now, after passing to the limit $\delta h \to 0$ and some scaling assumption on $T(\cdot)$, we arrive at the limiting problem

$$\partial_t u(t,x) \quad = \quad d\,\partial_{xx}\Big(T(v)\,u(t,x)\Big)\,, \qquad \text{for } (t,x) \in I \times \Omega\,,$$

where d is a constant stemming from the limiting assumption, see Remark 3.1. The multi-dimensional counterpart of this limiting problem, e.g., for $\Omega \subset \mathbb{R}^2$, reads

$$\partial_t u(t,x) \quad = \quad d\,\Delta\Big(T(v)\,u(t,x)\Big)\,, \qquad \text{for } (t,x) \in I \times \Omega\,. \qquad (3.1.3)$$

Particularly for a constant transition probability $T(\cdot)$ we end up with the common heat equation.

The second example for $T^{\pm}(\cdot)$ will be of more importance for our considerations of modeling chemotaxis phenomena. Let us assume that the cells are able to sense a local gradient of the chemical substance v. Then, we define the transition probabilities as

$$T_i^{\pm} = T^{\pm}(v_i) = \mu + \eta \left[\tau(v_{i\pm 1}) - \tau(v_i)\right], \tag{3.1.4}$$

where μ, η are positive constants and $\tau(\cdot)$ represents the ability of cells to sense the chemical, e.g., via receptors. Hence the constant μ scales the strength of the (unbiased) random walk, whereas η scales the strength of the chemotaxis-driven walk. Note that $\eta(\tau(v_{i\pm 1}) - \tau(v_i))$ can be viewed as a first-order approximation of the local gradient of τ when setting $\eta = \eta^*/\delta h$. The most simplest non-trivial receptor sensing τ can be described by a linear relation, e.g., we set $\tau(v) = v$. In order to keep the non-negativity of the transitions, here and hereafter we assume that $\mu \not\ll \eta$ and $|\tau(v_{i\pm 1}) - \tau(v_i)|$ is sufficiently small.

When using the transition probability (3.1.4), the transition process (3.1.1) reads

$$\begin{aligned} \partial_t u(t) \quad &= \quad \mu\left(u_{i-1}(t) - 2u_i(t) + u_{i+1}(t)\right) \\ &\quad -\eta\left(\left[u_{i+1}(t) + u_i(t)\right]\left[\tau(v_{i+1}) - \tau(v_i)\right] - \left[u_i(t) + u_{i-1}(t)\right]\left[\tau(v_i) - \tau(v_{i-1})\right]\right). \end{aligned} \tag{3.1.5}$$

After using Taylor expansions, passing the equation (3.1.5) to the limit $\delta h \to 0$ and assuming limiting properties for $T(\cdot)$ in a similar fashion as done before, we finally obtain the multi-dimensional equation

$$\partial_t u(t,\mathbf{x}) \quad = \quad \delta\left[\mu \Delta u(t,\mathbf{x}) - 2\eta \nabla \cdot \left(u(t,\mathbf{x})\, \partial_v \tau \nabla v(t,\mathbf{x})\right)\right], \quad \text{for } (t,\mathbf{x}) \in I \times \Omega. \tag{3.1.6}$$

Now we identify $\chi = \chi(v) := 2\eta\, \partial_v \tau$ as a chemosensitivity function and hence arrive at a more simple equation for the chemotaxis-driven evolution of the underlying organisms, e.g., cells,

$$\partial_t u(t,\mathbf{x}) \quad = \quad \delta\left[\mu \Delta u(t,\mathbf{x}) - \nabla \cdot \left(u(t,\mathbf{x})\, \chi(v)\, \nabla v(t,\mathbf{x})\right)\right], \quad \text{for } (t,\mathbf{x}) \in I \times \Omega. \tag{3.1.7}$$

Note that for a linear receptor sensing as mentioned before, i.e., $\tau(v) = v$, we end up with a constant chemosensitivity, i.e., $\chi(v) = \chi = const.$
When focusing on the evolution-equation for the chemical substance we can initially restrict ourselves to a rather simple diffusion model and basically exert a corresponding derivation as conducted for (3.1.3). For clarity reasons, this equation reads

$$\partial_t v(t,\mathbf{x}) \quad = \quad d_v \Delta v(t,\mathbf{x}), \quad \text{for } (t,\mathbf{x}) \in I \times \Omega. \tag{3.1.8}$$

Herein, $d_v > 0$ is a constant, scaling the chemical diffusion. We defer a brief introduction to chemical reaction processes for implementing reaction terms in the chemical equation to a subsequent paragraph, see Section 3.3.

We have seen how we can transform a microscopic description of a chemotaxis-driven motion (of cells) to a macroscopic scale, here a PDE model. However, as we already pointed out at the beginning of this chapter, a macroscopic perspective is sometimes inaccurate or even misleading. The crucial step in the above derivations is the limiting assumptions, $\delta h \to 0$, applied on the equations (3.1.2) and (3.1.5). Therein, we implicitly required that cells have an infinitesimal/neglectable size, which is a delicate assumption when considering chemicals being modeled at the same scale. In this work, we will nevertheless only consider PDE models for chemotaxis phenomena since they are both mathematically interesting and numerically challenging. Moreover a successful development of multi-scale models requires a deep understanding and accurate and efficient numerically handling of the chemotaxis process in general. This comprises a numerical PDE approach that provides data for examining chemotaxis in further detail.

3.1.2. Macroscopic derivation

Now we turn to the macroscopic derivation of the PDE model. To this end, we treat the functions u and v as sufficiently smooth and integrable density functions, rather than discrete entities as in the microscopic derivation. The following derivations are mainly extracted from Murray [78]. Correspondingly to this literature, we assume that the cells u yield the mass conservation law. Here, we explicitly model neither growth nor decay of the cells in the bounded domain Ω. Note that this assumption corresponds to the original pure aggregation stage in the life cycle of *Dictyostelium discoideum*, that Keller and Segel described in [52]. For an arbitrary bounded subset $\Omega' \subset \Omega$ with boundary denoted by $\partial\Omega'$, the mass (or energy) conservation law (for u) reads

$$\partial_t \int\limits_{\Omega'} u(t,\mathbf{x})\,\mathrm{d}\mathbf{x} \quad = \quad -\int\limits_{\partial\Omega'} \mathbf{F}(t,\mathbf{x})\cdot\mathbf{n}\,\mathrm{d}s, \qquad \text{for } t \in I, \tag{3.1.9}$$

where $\mathbf{F}(t,\mathbf{x})$ denotes the flux of u and $\mathbf{n}$ is the unit outward normal to $\partial\Omega'$. Verbally, equation (3.1.9) states that the rate of change of the density can fully be described by the rate of flow (flux) through the boundary of the underlying domain. If we now apply the divergence theorem and remind us of the smoothness assumption on u ($\partial_t u$ must be continuous), equation (3.1.9) can be rewritten in terms of

$$\int\limits_{\Omega'} \partial_t u(t,\mathbf{x}) + \nabla\cdot\mathbf{F}(t,\mathbf{x})\,\mathrm{d}\mathbf{x} \quad = \quad 0, \qquad \text{for } t \in I, \tag{3.1.10}$$

and because Ω' was chosen arbitrarily, we have

$$\partial_t u(t,\mathbf{x}) \quad = \quad -\nabla\cdot\mathbf{F}(t,\mathbf{x}), \qquad \text{for } (t,\mathbf{x}) \in I\times\Omega. \tag{3.1.11}$$

The final modeling task is now to choose a suitable flux term $\mathbf{F}(t,\mathbf{x})$. As already sorted out in the microscopic approach, we want to model two physically fluxes, a purely diffusive $\mathbf{F}_{\text{diffusion}}(t,\mathbf{x})$ and a rather chemotaxis-driven flux $\mathbf{F}_{\text{chemotaxis}}(t,\mathbf{x})$. The diffusive flux can be defined by Fick's law,

$$\mathbf{F}_{\text{diffusion}}(t,\mathbf{x}) \quad = \quad -d_u\nabla u(t,\mathbf{x}),$$

where $d_u > 0$ scales the diffusivity.

The chemotaxis flux offers some modeling purposes. On the one hand this flux should be proportional to the cell density, since this renders the increase of chemotaxis when cells aggregate. On the other hand, the receiving and processing of the (signal of the) chemical substance might be desired to be modeled by a function, say $\chi = \chi(v)$. Taking these considerations into account, we define the chemotaxis flux as

$$\mathbf{F}_{\text{chemotaxis}}(t,\mathbf{x}) \quad = \quad u(t,\mathbf{x})\,\chi(v)\,\nabla v(t,\mathbf{x}). \tag{3.1.12}$$

Altogether we consider the total flux to be given, simply, by the sum of the aforementioned partial fluxes, i.e.,

$$\mathbf{F}(t,\mathbf{x}) \quad = \quad \mathbf{F}_{\text{diffusion}}(t,\mathbf{x}) + \mathbf{F}_{\text{chemotaxis}}(t,\mathbf{x}). \tag{3.1.13}$$

Note the possible antagonistic roles (i.e., the ± signs) of the partial fluxes. That is, for positive $\chi(\cdot)$ we assign the chemotaxis flux a attractive character, sometimes called *chemoattraction*, while for negative $\chi(\cdot)$ we describe a repulsive effect, also called *chemorepellence*. Hence chemoattraction can be understood as an antagonist of diffusion.

We substitute the flux (3.1.13) into equation (3.1.11) and obtain under rearrangements the following equation for the cell density $u(t,\mathbf{x})$

$$\partial_t u(t,\mathbf{x}) \quad = \quad d_u \Delta u(t,\mathbf{x}) - \nabla \cdot \Big(u(t,\mathbf{x}) \chi(v) \nabla v(t,\mathbf{x}) \Big) , \qquad \text{for } (t,\mathbf{x}) \in I \times \Omega . \tag{3.1.14}$$

For the evolution equation of the chemical substance we restrict ourselves (for the moment) to a purely diffusion model as in the microscopic approach, cf. (3.1.8).

After rearrangement/renaming, both approaches eventually lead to the semi-coupled system

$$\begin{cases} \partial_t u(t,\mathbf{x}) & = & d_u \Delta u(t,\mathbf{x}) - \nabla \cdot \Big(u(t,\mathbf{x}) \chi(v) \nabla v(t,\mathbf{x}) \Big) , & \text{for } (t,\mathbf{x}) \in I \times \Omega , \\ \partial_t v(t,\mathbf{x}) & = & d_v \Delta v(t,\mathbf{x}) , & \text{for } (t,\mathbf{x}) \in I \times \Omega . \end{cases} \tag{3.1.15}$$

Up to here, the target cell equation has been fully developed, while the chemical equation misses essential reaction terms. We remind us that we wanted to model a self-enhancing chemotaxis scenario, that is cells are attracted by chemical signals which they in turn secrete by themselves. This positive feedback has to be modeled in the latter equation, e.g., by a production/source term. As a cross-reference note: this feedback leads to a coupled system, hence when returning to the microscopic approach, we recognize the (indirect) relation between the probability of the current transitions and the preceding states, i.e., the stand-alone process for u cannot be understood Markovian. Beside the source term, moreover the evolution of chemical signals are often subject to an abstract depletion term. The particular origin of this term (enzymatic degradation, consumption by alien processes/organisms, loss of effect) is perfunctorily for our current modeling. The easiest way of implementing these terms into our model (3.1.15) is by a simple linear relation, e.g., depletion by $-\alpha v$ and production/source by $+\beta u$. Alternative reaction terms are discussed in the next subsection. By virtue of a general term, say $r(v,u)$, we can write

$$\begin{cases} \partial_t u(t,\mathbf{x}) & = & d_u \Delta u(t,\mathbf{x}) - \nabla \cdot \Big(u(t,\mathbf{x}) \chi \nabla v(t,\mathbf{x}) \Big) , & \text{for } (t,\mathbf{x}) \in I \times \Omega , \\ \partial_t v(t,\mathbf{x}) & = & d_v \Delta v(t,\mathbf{x}) + r(v,u) , & \text{for } (t,\mathbf{x}) \in I \times \Omega . \end{cases} \tag{3.1.16}$$

These equations provide a good starting point for studying chemotaxis-driven processes. Nevertheless it should be noted that the original Keller-Segel model was introduced in a more general setting where possible nonlinear coefficients model the partial processes in more elaborate fashion. Besides different coefficients, Keller and Segel also consider a total of four equations, instead of only two. For a more differentiated approach they included additional equations for an enzyme, which corresponds to the chemical substance v, and a complex that is formed by chemical reactions of v and the latter enzyme. Basically this models a more sophisticated depletion of the chemical.

Remark 3.1 *Concerning the limiting assumptions in the microscopic derivation of the chemotaxis equation (3.1.3) some supplementary notes are advisable.*
First of all we remark that the transitions $T^{\pm}(\cdot)$ implicitly depend on the (discretized) time stepping, therefore we better refer to $T^{\pm}(\cdot)$ as transitions per time unit and hence assume

$$T^{\pm}(\cdot) \quad = \quad \mathcal{O}(\delta t^{-1}) .$$

The rather technical appealing limiting assumption for the so called 'Diffusion limit' to hold reads

$$\lim_{\substack{\delta h \to 0 \\ \delta t \to 0}} \frac{\delta h^2}{\delta t} \quad = \quad d > 0\,.$$

In other words, the limiting process is not homogeneous in space and time, namely the time steps shrink significantly faster than the spatial mesh size. This inhomogeneous scaling can be regarded as one reason why the well known heat equation admits an infinite speed of propagation, which is not what we might expect from the real physical process of diffusion. The infinite speed of propagation refers to the fact that an initial heat distribution, say $\vartheta(0,\mathbf{x}) \geq 0$*, with small support immediately propagates to the entire domain in terms of*

$$\vartheta(0,\mathbf{x}) = 0\,, \quad \textit{for}\ \mathbf{x} \in \Omega \setminus \mathrm{supp}(\vartheta_0) \quad \to \quad \vartheta(t,\mathbf{x}) > 0\,, \quad \textit{for any}\ t > 0\ \textit{and}\ \mathbf{x} \in \Omega\,.$$

There is an approach that tackles this paradoxon. In the context of thermodynamics, it was Cattaneo [15] who modified the well-established Fourier's law in order to solve the 'paradox of heat conduction' already in 1948. Fourier's law postulates that the heat flux, say $\mathbf{F}_{heat}(t,\mathbf{x})$*, is proportional to the negative temperature gradient*

$$\mathbf{F}_{heat}(t,\mathbf{x}) \quad = \quad -d\,\nabla\vartheta(t,\mathbf{x})\,,$$

where ϑ *denotes the temperature of a homogeneous medium. The main idea of Cattaneo was to add a small delay term to this equation, accounting to the fact that the heat flux needs (at least a small amount of) time adapting to be proportional to the negative temperature gradient. Another interpretation of Cattaneo's modification is that the heat flux depends not only on the current temperature gradient but also on its past. Both perspectives drove Cattaneo to introduce his version of the heat flux,*

$$\mathbf{F}_{heat}(t,\mathbf{x}) + \tau\partial_t\mathbf{F}_{heat}(t,\mathbf{x}) \quad = \quad -d\,\nabla\vartheta(t,\mathbf{x})\,.$$

Herein $\tau > 0$ *determines the adaptation time referred to above. Note that for* $\tau = 0$ *we reobtain Fourier's law. Together with the equation for energy conservation (cf. (3.1.10)) we obtain the so-called 'Cattaneo system'*

$$\begin{cases} \partial_t\vartheta(t,\mathbf{x}) + \nabla\cdot\mathbf{F}_{heat}(t,\mathbf{x}) \;=\; 0\,, & \textit{for}\ (t,\mathbf{x}) \in I \times \Omega\,, \\ \tau\partial_t\mathbf{F}_{heat}(t,\mathbf{x}) + \mathbf{F}_{heat}(t,\mathbf{x}) \;=\; -d\,\nabla\vartheta(t,\mathbf{x})\,, & \textit{for}\ (t,\mathbf{x}) \in I \times \Omega\,. \end{cases} \tag{3.1.17}$$

This hyperbolic system has indeed the property of providing a finite speed of propagation, hence rendering this system physically more appropriate.
Cattaneo's system can also be reformulated for chemotaxis-driven motion of cells. A straightforward substitution of the chemotaxis flux (3.1.13) in (3.1.17) leads to the following 'Cattaneo model for chemosensitive movement', which was presented by Dolak and Hillen [24],

$$\begin{cases} \partial_t u(t,\mathbf{x}) + \nabla\cdot\mathbf{F} \;=\; 0\,, & \textit{for}\ (t,\mathbf{x}) \in I \times \Omega\,, \\ \tau\partial_t\mathbf{F}(t,\mathbf{x}) + \mathbf{F}(t,\mathbf{x}) \;=\; -d_u\,\nabla u(t,\mathbf{x}) + u(t,\mathbf{x})\,\chi(v)\,\nabla v(t,\mathbf{x})\,, & \textit{for}\ (t,\mathbf{x}) \in I \times \Omega\,. \end{cases} \tag{3.1.18}$$

Hereby, the evolution equation for the chemical concentration $v(t,\mathbf{x})$ *is usually modeled as in the previous chemotaxis systems, e.g., (3.1.16). Numerical simulations of systems of kind (3.1.18) revealed that the qualitative difference between classical chemotaxis models and the Cattaneo model*

for chemosensitive movement is only recognizable for short time ranges. The asymptotic behavior of both modeling attempts seem to be very similar, see [24].
Let us remark that the derivation of the Cattaneo system can also be achieved from a microscopic perspective. In the context of chemotaxis-driven motion, the interested reader is kindly referred to Hadeler [41] and Hillen [43].

3.2. Dimensionless formulation

The most common representative of a chemotaxis model of kind (3.1.16) deals with linear reaction terms as already mentioned above, i.e., we consider

$$\begin{cases} \partial_t u &= d_u \Delta u - \nabla \cdot (u \chi \nabla v)\,, \qquad \text{for } (t,\mathbf{x}) \in I \times \Omega\,, \\ \partial_t v &= d_v \Delta v - \alpha v + \beta u\,, \qquad \text{for } (t,\mathbf{x}) \in I \times \Omega\,. \end{cases}$$

Herein, the five model parameters, d_u, χ, d_v, α and β calibrate the model more or less significantly. In order to run a proper simulation, these parameters are usually determined by experimental data sets which can be found in various experimental assays, e.g., Adler [1], Budrene and Berg [14] or Greenberg and Canale-Parola [36]. However, if we are interested in the dynamics of model (3.2.1) for a more general setting, that is if we are looking for the relation between solutions and a certain parameter, χ say, then the task would be easier for a minimal set of parameters. Furthermore for experimental unknown parameters the proper choice for simulation purposes might be very tedious. In this scenario it is more convenient to consider only certain relations between those parameters. Moreover, it is obvious that less parameters simplify the derivations of analytical results such as existence, uniqueness or stability. Hence, from the theoretical and practical point of view the elimination of redundant model parameters are highly recommended. These considerations lead to a so-called dimensionless formulation of the governing model (3.2.1).

There are indeed basic guidelines to non-dimensionalize a model of kind (3.2.1). The first step will be to scale the underlying coordinate system (including the time). For appropriately chosen parameters $x_*, t_*, u_*, v_* \in \mathbb{R}$ we define

$$\begin{aligned} \hat{\mathbf{x}} &= x_* \mathbf{x}\,, \\ \hat{t} &= t_* t\,, \end{aligned}$$

and substitute the solutions correspondingly

$$\begin{aligned} \hat{u}(\hat{x},\hat{t}) &= u_* u(t,\mathbf{x})\,, \\ \hat{v}(\hat{x},\hat{t}) &= v_* v(t,\mathbf{x})\,. \end{aligned}$$

To simplify the calculations, we restrict our derivations from now on to one spatial dimension. When substituting the above scalings in the partial derivatives of the solutions, we end up with

$$\begin{aligned} \partial_t u(t,x) &= \frac{1}{u_*} \partial_t \hat{u}(\hat{t},\hat{x}) &= \frac{t_*}{u_*} \partial_{\hat{t}} \hat{u}(\hat{t},\hat{x})\,, \\ \partial_t v(t,x) &= \frac{1}{v_*} \partial_t \hat{v}(\hat{t},\hat{x}) &= \frac{t_*}{v_*} \partial_{\hat{t}} \hat{v}(\hat{t},\hat{x})\,, \\ \partial_x v(t,x) &= \frac{1}{v_*} \partial_x \hat{v}(\hat{t},\hat{x}) &= \frac{x_*}{v_*} \partial_{\hat{x}} \hat{v}(\hat{t},\hat{x})\,, \end{aligned}$$

$$\begin{aligned}
\partial_{xx}u(t,x) &= \frac{1}{u_*}\partial_x\Big(x_*\,\partial_{\hat{x}}\hat{u}(\hat{t},\hat{x})\Big) = \frac{x_*^2}{u_*}\partial_{\hat{x}\hat{x}}\hat{u}(\hat{t},\hat{x})\,,\\
\partial_{xx}v(t,x) &= \frac{1}{v_*}\partial_x\Big(x_*\,\partial_{\hat{x}}\hat{v}(\hat{t},\hat{x})\Big) = \frac{x_*^2}{v_*}\partial_{\hat{x}\hat{x}}\hat{v}(\hat{t},\hat{x})\,,\\
\partial_x\Big(u(t,x)\,\partial_x v(t,x)\Big) &= \frac{x_*}{v_*}\partial_x\Big(u(t,x)\,\partial_{\hat{x}}\hat{v}(\hat{t},\hat{x})\Big) = \frac{x_*}{v_*}\partial_x\Big(\frac{1}{u_*}\hat{u}(\hat{t},\hat{x})\,\partial_{\hat{x}}\hat{v}(\hat{t},\hat{x})\Big)\\
&= \frac{x_*}{v_*}\Big(\frac{x_*}{u_*}\hat{u}(\hat{t},\hat{x})\,\partial_{\hat{x}}\hat{v}(\hat{t},\hat{x}) + \frac{x_*}{u_*}\hat{u}(\hat{t},\hat{x})\,\partial_{\hat{x}\hat{x}}\hat{v}(\hat{t},\hat{x})\Big)\\
&= \frac{x_*^2}{v_*\,u_*}\partial_{\hat{x}}\Big(\hat{u}(\hat{t},\hat{x})\,\partial_{\hat{x}}\hat{v}(\hat{t},\hat{x})\Big)\,.
\end{aligned}$$

Together with straightforward substitutions for the reactions terms in (3.2.1) the (one dimensional) model is reformulated as

$$\begin{cases}
\dfrac{t_*}{u_*}\partial_{\hat{t}}\hat{u}(\hat{t},\hat{x}) = d_u\dfrac{x_*^2}{u_*}\partial_{\hat{x}\hat{x}}\hat{u}(\hat{t},\hat{x}) - \dfrac{\chi x_*^2}{v_*\,u_*}\partial_{\hat{x}}\Big(\hat{u}(\hat{t},\hat{x})\partial_{\hat{x}}\hat{v}(\hat{t},\hat{x})\Big)\,, & \text{for } (t,\mathbf{x})\in I\times\Omega\,,\\[2ex]
\dfrac{t_*}{v_*}\partial_{\hat{t}}\hat{v}(\hat{t},\hat{x}) = d_v\dfrac{x_*^2}{v_*}\partial_{\hat{x}\hat{x}}\hat{v}(\hat{t},\hat{x}) + \dfrac{\beta}{u_*}\hat{u}(\hat{t},\hat{x}) - \dfrac{\alpha}{v_*}\hat{c}(\hat{t},\hat{x})\,, & \text{for } (t,\mathbf{x})\in I\times\Omega\,.
\end{cases} \tag{3.2.1}$$

After ‘normalizing’ the time derivatives on the left-hand side and demanding the coefficients in the v equation to be normalized as well, we end up with the system

$$\begin{cases}
\partial_{\hat{t}}\hat{u}(\hat{t},\hat{x}) = d_u\dfrac{x_*^2}{t_*}\partial_{\hat{x}\hat{x}}\hat{u}(\hat{t},\hat{x}) - \dfrac{\chi x_*^2}{t_*\,v_*}\partial_{\hat{x}}\Big(\hat{u}(\hat{t},\hat{x})\partial_{\hat{x}}\hat{v}(\hat{t},\hat{x})\Big)\,, & \text{for } (t,\mathbf{x})\in I\times\Omega\,,\\[2ex]
\partial_{\hat{t}}\hat{v}(\hat{t},\hat{x}) = \underbrace{d_v\dfrac{x_*^2}{t_*}}_{\overset{!}{=}1}\partial_{\hat{x}\hat{x}}\hat{v}(\hat{t},\hat{x}) + \underbrace{\dfrac{\beta\,v_*}{t_*\,u_*}}_{\overset{!}{=}1}\hat{u}(\hat{t},\hat{x}) - \underbrace{\dfrac{\alpha}{t_*}}_{\overset{!}{=}1}\hat{v}(\hat{t},\hat{x})\,, & \text{for } (t,\mathbf{x})\in I\times\Omega\,.
\end{cases} \tag{3.2.2}$$

We set the scaling coefficients for the dimensionless parameters correspondingly to the last equation and introduce two new model parameters $\hat{\delta},\hat{\chi}\in\mathbb{R}$, which leads to

$$\begin{aligned}
t_* &= \alpha\,,\\
x_* &= \sqrt{\alpha/d_v}\,,\\
v_* &= 1\,,\\
u_* &= \beta/\alpha\,,\\
\hat{d} &:= d_u/d_v\,,\\
\hat{\chi} &:= \chi/d_v\,.
\end{aligned}$$

After dropping the hat notation ($\hat{\cdot}$) these definitions finally lead to the dimensionless system of model (3.2.1)

$$\begin{cases}
\partial_t u(t,x) = d\,\partial_{xx}u(t,x) - \partial_x\Big(u(t,x)\,\chi\,\partial_x v(t,x)\Big)\,, & \text{for } (t,\mathbf{x})\in I\times\Omega\,,\\
\partial_t v(t,x) = \partial_{xx}v(t,x) + u(t,x) - v(t,x)\,, & \text{for } (t,\mathbf{x})\in I\times\Omega\,.
\end{cases} \tag{3.2.3}$$

It is straightforward to deduce the counterpart of system (3.2.3) for multiple dimensions, e.g., $\Omega\subset\mathbb{R}^2$. For the remainder of this work, we will refer to the following model as the minimal model of chemotaxis in its dimensionless form,

$$\begin{cases} \partial_t u(t,x) &= \nabla \cdot \big(d \nabla u(t,x) - u(t,x) \chi \nabla v(t,x) \big), \qquad \text{for } (t,x) \in I \times \Omega, \\ \partial_t v(t,x) &= \Delta v(t,x) + u(t,x) - v(t,x), \qquad \text{for } (t,x) \in I \times \Omega. \end{cases} \tag{3.2.4}$$

Note that we encountered this model already in Section 2.2.

We observe that our system (3.2.4) only incorporates two model parameters, namely d and χ. As already mentioned above, these parameters can be interpreted as certain relations of the original parameters in (3.2.1). In detail, we refer to $d = d_u/d_v$ as the diffusion rate, which obviously determines the ratio of the diffusion rates of u and v. Furthermore, χ now indicates the ratio of the original chemosensitivity and the diffusion rate of the chemicals.

For the remainder of this work we will ease the reading of upcoming model equations by omitting the time and space variables if they are not particularly part of the main focus.

3.3. Some model extensions

As mentioned earlier, extensions to the minimal model of chemotaxis were in fact already considered in the original paper of the ‘founding fathers’ of chemotaxis models. Not surprisingly that some of them have lately been revived by recent scientists. Therefore, in this section we will have a look on some particular interesting extensions that are subject of very recent works and this current investigation. In the context of more elaborate modeling of certain processes, e.g., in terms of non-trivial coefficients, let us already now refer to the short survey about the Michaelis-Menten theory and Monod kinetics in the appendix C.

3.3.1. About Growth Kinetics

Referring to the cell equation in (3.1.16), we did not take into account some proliferation terms, namely we always assumed mass conservation in a simple modeled aggregation phase of the underlying organism such as *Dictyostelium discoideum*. However the explicit modeling of growth terms is of paramount interest when it comes to model highly invasive processes like angiogenesis and bacteria proliferation, or if we model more than one generation time of the life cycle of the organism. For example, in the latter case it was nicely demonstrated by Budrene and Berg [14] that bacteria form astonishing patterns when they were exposed to certain stresses, cf. Figure 3.1.

In the context of a PDE model, a very common approach for modeling growth is derived by introducing a Fisher-type term. It is well known that the Fisher equation admits traveling wave solutions which describe a logistic growth. In terms of the cell concentration u, Fisher's equation can be written in the following form

$$\partial_t u \quad = \quad \Delta u + \kappa u (1 - u/K).$$

Herein, $\kappa > 0$ denotes a constant growth rate and $K > 0$ is the carrying capacity, namely, the maximum cell concentration admitted by the environment (caused by certain resource limits). A careful look on the growth term reveals that the cell accumulation now yields an increase by $+\kappa u$ in the early stage and a decrease by $-\kappa u^2/K$ in a later stage, modeling the competition for the critical

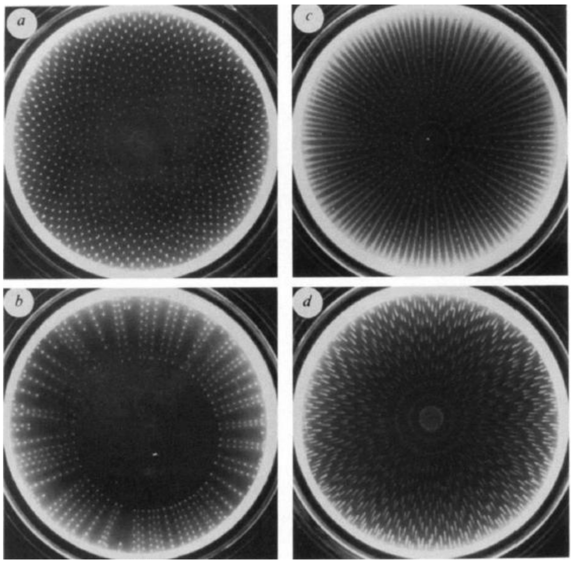


Figure 3.1: Patterns formed by *Escherichia coli* in semi-solid agar. Original experimental observations of Budrene and Berg [14], used with permission from Howard C. Berg, Department of Molecular and Cellular Biology, Harvard University.

resource.

By simply adding the Fisher-term to our cell equation in (3.1.16), we end up with a corresponding chemotaxis model incorporating cell-growth,

$$\begin{cases} \partial_t u &= d_u \Delta u - \nabla \cdot (u \chi(v) \nabla v) + \kappa u (1 - u/K) \,, \\ \partial_t v &= d_v \Delta v + r(v,u) \,. \end{cases} \tag{3.3.1}$$

Let us remark that this logistic growth can be related to a more comprehensive theory of (bacterial) growth, the *Monod model*, see appendix C. Here, we like to focus on this relation in a simplified setting. To this end, let $\kappa = 1 = K$, hence the Fisher-term reads $(1-u)\,u$, subject to $0 < u \leq 1$. We introduce a variable $s > 0$ for the limiting resource/substrate concentration, e.g., a nutrient. We will now relate the limiting resource to the present cell concentration. If we set $s = K_s(1/u - 1)$, where $K_s > 0$ denotes a constant, a simple calculus reveals

$$\frac{s}{s + K_s} \;=\; 1 - u \,.$$

The left-hand side of this equation is comparable to what Monod formulated as a growth rate μ for bacteria cultures, given that $\mu_{\max} = 1$. Such kind of relations for certain coefficients also lead to further extensions of (3.1.16) which we will see in the subsequent paragraphs. Before turning to other extensions, let us briefly discuss a justification for the definition of the limiting resource s. If $u \to 0$ (however never vanishing completely), we note $s \to \infty$ and if $u \to 1$, the limiting resource yields $s \to 0$. In view of a limiting resource, this is reasonable: it describes the increase or decline of the limiting resource as the cells cease or reach their carrying capacity (here $K = 1$), respectively.

3.3.2. About chemosensitivities

We introduced the chemotaxis flux as $\mathbf{F}_{\text{chemotaxis}} = -u\chi(v)\nabla v$, cf. (3.1.12). To define $\chi(v)$, early-stage experiments already motivated Keller and Segel [52] and later Lapidus and Schiller [65] to propose chemosensitivities that promote fluxes due to (non-trivial) rational functions of type $\chi(v) = 1/v$ (cf. the *Weber-Fechner law*) and $\chi(v) = 1/(1+v)^2$, respectively. The latter is based on observations that the chemotactic response declines at low chemical concentrations and saturates at high concentrations, in contrast to the derivative of the logarithm of the concentration as in the former case – note that $\nabla \log v = \nabla v/v$. In fact the phenomenon that is modeled by these terms is also called signal-dependent (chemo-)sensitivity in the literature, e.g., [44], and corresponds to a simple model for a receptor-signal binding (we already assumed beforehand that the chemicals bind to certain receptors located on the cell membrane, rf. Chapter 1).

Consider the following reaction

$$R_f + V \underset{k_{-1}}{\overset{k_1}{\rightleftharpoons}} R_b\,,$$

where R_f, R_b denote free and bounded receptors, respectively, and V is a molecule of the chemical active substance. Assuming a constant amount of receptors, the potential of chemotaxis can be described by the current number of bounded receptors R_b. Therefore, if v exceeds a threshold, a saturation of bounded receptors emerges and hence, the chemotaxis flux cannot be considered solely proportional to the (negative) gradient of v anymore. After employing a steady-state hypothesis about the bounded receptors in the above reaction, we end up with a concentration of R_b that is highly related to a Michaelis-Menten-type relation, cf. [44].

Another notable chemosensitivity can be obtained by a *volume-filling* approach. Its fundamental idea is that cells carry a certain volume (nonzero and finite). By assuming that cells do not penetrate/overlap, the occupation of a certain area limits the chemotaxis-driven attraction. In fact, also the general motility (including diffusion) is limited by these volume-filling effects. An illustration of volume filling is given in Figure 3.2. Therein, the circles represent cells moving to the right, along the chemical gradient. Cells A, B and C are located in regions of increasingly packed cell compounds. Cell A can freely move, whereas the movement of Cell B is already moderately limited, Cell C can finally hardly move since surrounding cells block individual movement. A detailed derivation of this type of chemosensitivity can be found in [88]. Since we consider individual cell volumes the derivation is conveniently carried out from the microscopic point of view. We modify the transition (3.1.4) in terms of

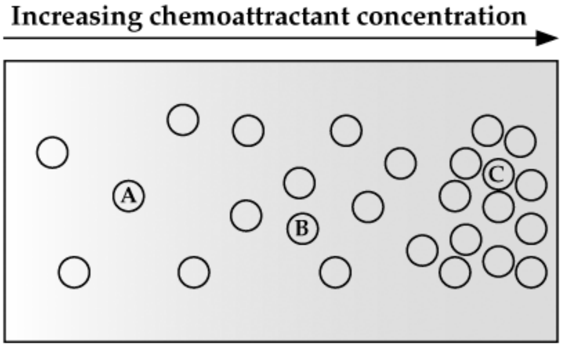


Figure 3.2: Illustration of volume-filling, modified from [88].

$$T_i^{\pm} = T_i^{\pm}(v) = q(u_{i\pm1})\left[\mu + \eta(v_{i\pm1} - v_i)\right]. \tag{3.3.2}$$

Note that for notational simplicity we set $\tau(v_j) = v_j$. The newly introduced function $q(u_j)$ describes the probability of unoccupied space. When considering a maximum of accumulated cells in a region $u_{\max}$ we assume the function to yield

$$q(u_{\max}) = 0 \quad \text{and} \quad q(u) \geq 0\,, \quad \text{for } 0 \leq u \leq u_{\max}\,.$$

After passing to the continuous limit as before we end up with the following equation for u,

$$\partial_t u \;\; = \;\; \nabla\Big(d_u\left[q(u) - u\,q'(u)\right]\nabla u - u\chi\,q(u)\,\nabla v\Big)\,. \tag{3.3.3}$$

A reasonable and convenient choice of a linear occupation function reads $q(u) = 1 - u/u_{\max}$. This choice has the nice property of a vanishing diffusive contribution $q(u) - u\,q'(u) = 0$ and therefore is very commonly used. We remark that when $u_{\max} \to \infty$ we arrive at the classical chemosensitivity, which also appeals to common sense.

In the course of further experimental results, several authors proposed alternative chemosensitivities, the interested reader is referred to the given literature for further studies. A selected listing is sketched in Table 3.1.

Exemplary reference	Chemosensitivity	Background
Keller and Segel [52]	$\chi(v) = \chi = const$	constant sensitivity
Keller and Segel [52]	$\chi(v) = \frac{\chi}{v}$	logarithmic law (Weber-Fechner)
Hillen and Painter [44]	$\chi(v) = \frac{\chi\kappa}{(\kappa+v)^2}\,(\star)$	receptor kinetic law

Table 3.1: Overview of particular chemosensitivities discussed in the preceding paragraph. $(\star)$ The constant κ is sometimes called the dissociation constant for the receptor-attractant interaction, in our formulation we set $\kappa = 1$.

3.3.3. About chemical reaction rates

Up to now we did not define a particular reaction term $r(v,u)$ for the chemical substance in (3.1.16). For reasons of comprehensibility we split the reaction term into the explicit contributions of production, say $r_+(v,u)$, and degradation, say $r_-(v,u)$, of the chemical, i.e.,

$$r(v,u) \;\; = \;\; r_+(v,u) - r_-(v,u)\,.$$

We begin with a discussion of a proper production term. The most common definition, which also appears in the minimal model (3.2.4), is $r_+(v,u) = \beta\,u, \quad \beta > 0$. This approach however fails when reconsidering the signal pathway of the chemical-receptor binding more carefully. Reasonable arguments promote a fall-off in the production rate at high cell/chemical concentrations. Indeed, in many cases the chemical-receptor binding induces an internal signal pathway for a saturating chemical production, viz., the cells' chemical-production rate decreases at sufficiently high chemical concentrations. In fact, this kind of saturating effect is observed in various pathways besides chemotaxis, e.g., hormone secretion. Referring to this perspective, a suitable production rate is given by a Michaelis-Menten-like relation. Exemplary, Tyson *et al.* [101] employed a production rate given by

$$r_+(v,u) \quad = \quad \frac{\beta u^2}{\kappa + u^2},$$

where κ is a constant. Note that in this case the chemical production saturates with increasing cell concentration and not directly by the concentration of bounded/active chemical-receptor bindings. A similar relation, although originally established in a different context, was proposed by Moser [77]. The following table, Table 3.2, provides three production rates that are commonly studied in the context of evolution equations for bacteria or simple cells.

Exemplary reference	Production rate	Background
Nanjundiah [83]	$r_+(v,u) = \beta u$	proportional to cell concentration
Hillen and Painter [44]	$r_+(v,u) = \frac{\beta u}{1+\kappa u}$	Michaelis-Menten like
Tyson *et al.* [101]	$r_+(v,u) = \frac{\beta u^2}{\kappa+u^2}$	Moser-like

Table 3.2: Overview of particular production rates.

The dilution or consumption rates are assumed to be proportional to the chemical concentration itself, i.e., $r_-(v,u) = \alpha v, \quad \alpha > 0$. This is a common and plausible setting in many cases. However if the chemical is neither consumed nor degraded, e.g., under the assumption of sufficient nutrients for the cells [101], we can also consider explicit zero-degradation, i.e., $\alpha = 0$. A classical alternative of pure consumption by the cells is also modeled by $r_-(v,u) = \alpha uv$. This relation is also used in the context of predator-prey systems introduced by Lotka and Volterra, in the presence of chemotaxis motivated equations it was employed by, e.g., [88]. A fourth example of chemical degradation can be again derived by the Michaelis-Menten kinetics in a similar fashion as before, [52]. For reasons of clarity let us list the above degradation terms in a table , Table 3.3.

Exemplary reference	Degradation rate	Background
Tyson *et al.* [101]	$r_-(v,u) = 0$	no uptake/consumption
Nanjundiah [83]	$r_-(v,u) = \alpha v$	proportional to chemical concentration
Painter and Hillen [88]	$r_-(v,u) = \alpha u v$	consumption by cells, cf. predator-prey systems
Keller and Segel [52]	$r_-(v,u) = \frac{\alpha v}{\kappa+v}$	following Michaelis-Menten kinetics

Table 3.3: Overview of particular degradation terms.

3.3.4. About the parabolic-elliptic simplification

All of the models considered so far comprise unsteady equations for u and v, wherein the temporal scale is similar. However, sometimes it is more appropriate to consider different temporal scales. In particular, this occurs if the run-and-tumble-walk of bacteria is considerably slow compared to the diffusion of simple chemoactive molecules. This plausible scenario is found in the literature by explicitly scaling the diffusion, e.g., [73], or even by a stationary equation for the chemical v, which was the basis for first detailed fundamental theoretical analysis of chemotaxis models, cf. [42, 50, 81]. Note that a different scaling of the diffusion processes can cause Turing instabilities and hence, scaling approaches that are similar to those in [73] can also lead to chemotaxis models that generate such instabilities.

4

Discretization of a general chemotaxis model

Because of the variety of the chemotaxis model introduced in the preceding chapter, it seems convenient to formulate the discretization for a rather general model of chemotaxis. This section is therefore dedicated to the formulation of a suitable general model and its FE discretization. We like to stress that the well-posedness of this following general model in terms of existence and uniqueness of solutions is not among the topics of this work and the curious reader is kindly referred to the corresponding literature provided in Chapter 2. Indeed we promote the general formulation mostly with respect to the application point of view, at the well known cost of mathematical incompleteness, strictly speaking.

In this work, the actual numerical treatment of such a continuous model formulation via PDEs will be tackled by FEM, those basic notations have already been depicted in the preliminaries, Chapter 2.1. After a straightforward calculation of the weak formulation of the underlying PDE we will obtain the standard Galerkin discretization in space. The discretization in time will then be established by the common theta-scheme, which includes a first-order fully explicit scheme (*forward Euler*) for $\theta = 0$ and a first-order fully implicit scheme (*backward Euler*) for $\theta = 1$ as well as the second-order so called *Crank-Nicolson* scheme for $\theta = 0.5$.

4.1. Formulation of a general model

The governing general model is taken from [97] and reads as follows,

$$\left\{ \begin{array}{rcl} \partial_t u &=& \nabla \cdot \big(d_u \nabla u - u \chi(v) \nabla v \big) + g(u) \, u \, , \\ \partial_t v &=& d_v \Delta v - \alpha v + s(u) \, u \, . \end{array} \right. \tag{4.1.1}$$

Another more general formulation can be found in [44]. In the subsequent section we will derive the FE formulation for our general model (4.1.1), whereas it is more convenient to formulate parts of the discretization schemes only for particular models. The contributions of the single terms are referred to the physical processes and the relations to the variants in Section 3 can be associated very easily. For reasons of clarity we depict the relations for three designated models explicitly.

Minimal model

The minimal model of chemotaxis as it was studied by, e.g., Nanjundiah in [83] reads

$$\begin{cases} \partial_t u &= \nabla \cdot \big(\nabla u - u\chi\nabla v\big)\,, \\ \partial_t v &= d_v \Delta v - v + u\,. \end{cases} \tag{4.1.2}$$

This system can easily be obtained from the general model (4.1.1) by choosing the following coefficients

$$d_u = 1, \quad \chi(v) = \chi\,(= const), \quad g(u) = 0, \quad \alpha = 1, \quad s(u) = 1.$$

We already discussed this model in Chapter 2 and hence the reader may recall this chapter as reference.

Aggregation model

The following model mimics the ability of the population u to form multiple aggregates that finally merge while a blowing up solution is prevented. The modifications of the corresponding terms have already been considered by, e.g., Tyson *et al.* [101] and Hillen together with Painter [44], cf. Section 3.3. The model reads

$$\begin{cases} \partial_t u &= \nabla \cdot \big(d_u \nabla u - \chi \dfrac{u}{(1+v)^2} \nabla v\big)\,, \\ \partial_t v &= d_v \Delta v + \dfrac{u^2}{1+u^2}\,. \end{cases} \tag{4.1.3}$$

We note the absence of a depletion term in the v equation which models a constant saturated nutrients level for u. We arrive at this model by considering the coefficients

$$\chi(v) = \chi/(1+v)^2, \quad g(u) = 0, \quad \alpha = 0, \quad s(u) = u/(1+u^2).$$

By this particular choice of $s(u)$ we obtain a nonlinearity also for the v equation which influences the overall nonlinear convergence of the underlying solvers. We will spotlight this remark in the corresponding chapter of numerical observations later on.

Kinetic model

The third exemplary model is originally based on the experimental observations of certain bacterial populations, e.g., by Budrene and Berg [14]. Modifying the Fisher-term introduced in Chapter 3 we will provide a higher-order proliferation term (motivated by the work of Mimura *et al.* [73]). A typical model that admits evolving patterns can be stated as

$$\begin{cases} \partial_t u &= \nabla \cdot \big(d_u \nabla u - \chi u \nabla c\big) + u(1-u)(u-a), \\ \partial_t v &= \Delta v - \alpha v + u. \end{cases} \tag{4.1.4}$$

This model arises by selecting

$$\chi(v) = \chi, \quad g(u) = (1-u)(u-a), \quad d_v = 1, \quad s(u) = 1,$$

where $0 \leq a < 1$ is a constant. Remark that a Fisher-like term with a carrying capacity of $K = 1$ is recovered by choosing $a = 0$. A choice of $0 < a < 1$ however is a more convenient parameter since it can be viewed as modeling a threshold that the species must overcome (in terms of $a < u$) in order to proliferate (until the population reaches the prescribed carrying capacity, $u = 1$). Figure 4.1 plots a sequence of the proliferation term with increasing values of a. The value of a determines the threshold for a positive growth contribution. Furthermore, let us note that some proliferation

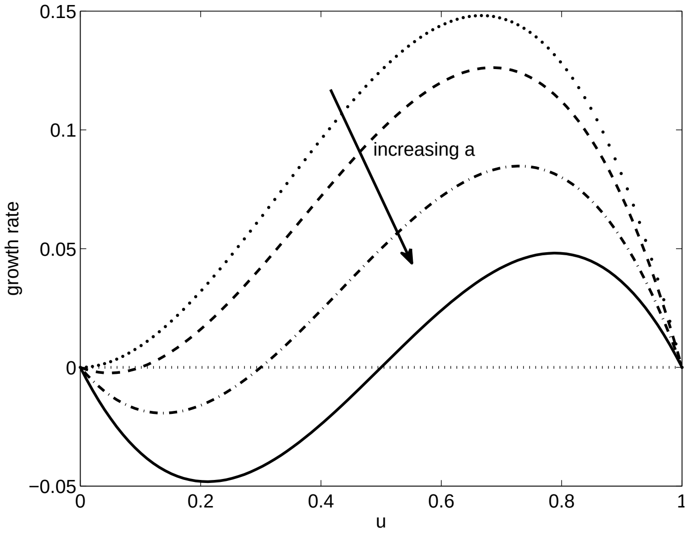


Figure 4.1: Plots of the growth rate for (4.1.4) for four values $a = 0, 0.1, 0.3, 0.5$.

models explicitly require different scales for the involving physical processes. For example, when we solely model a particular stage in the growth phase of, e.g., bacteria (cf. the last paragraph in the appendix C), we can safely assume that the scales of bacteria diffusion and chemotaxis, d_u, χ, respectively, are remarkably lower than the scales of the proliferation and the dynamics of the chemical. Following Mimura *et al.* [73] we can choose a order of ε^2 and ε ($\varepsilon \ll 1$) for d_u and χ, respectively.

For the numerical treatment of the general model (4.1.1), it can be complemented by usual prescribed initial values (2.2.2) and certain boundary conditions, e.g., homogeneous Neumann boundary conditions (2.2.3) or total flux boundary conditions of the form

$$\mathbf{n} \cdot \big(d_u \nabla u - u\chi(v)\nabla v\big) = 0, \qquad \mathbf{n} \cdot \nabla v = 0 \qquad \text{on} \quad \partial\Omega, \tag{4.1.5}$$

where $\mathbf{n}$ denotes the unit outward normal to the boundary $\partial\Omega$.

4.2. Finite element formulation

In order to derive proper fully discretized FE schemes for our general chemotaxis PDE model at hand (4.1.1) together with (2.2.2) and (2.2.3) or (4.1.5), we will start to introduce the corresponding FE formulations for the spatial discretization step-by-step. As already mentioned in the beginning of this chapter, we will consider the conventional Galerkin method in this work.

Let us begin our derivations by stating the weak formulation of (4.1.1). We assume that the solutions u and v exist in some (sufficiently smooth) space V. We multiply both sides of the equations by a suitable test function, say $\psi \in V_h \subset V$, and integrate over the underlying computational domain Ω

$$\begin{cases} \int_\Omega \psi \partial_t u \, \mathrm{d}\mathbf{x} &= \int_\Omega \psi \Big[\nabla \cdot \big(d_u \nabla u - u \chi(v) \nabla v \big) + g(u)\, u \Big] \, \mathrm{d}\mathbf{x} \\ \int_\Omega \psi \partial_t v \, \mathrm{d}\mathbf{x} &= \int_\Omega \psi \Big[d_v \Delta v - \alpha v + s(u)\, u \Big] \, \mathrm{d}\mathbf{x} \,. \end{cases} \tag{4.2.1}$$

After integrating by parts (of certain terms) we reformulate the weak form as

$$\begin{aligned} \int_\Omega \psi \partial_t u \, \mathrm{d}\mathbf{x} &= -\int_\Omega \nabla \psi \cdot \big(d_u \nabla u - u \chi(v) \nabla v \big) - \psi g(u)\, u \, \mathrm{d}\mathbf{x} + \int_{\partial\Omega} \psi \big(d_u \nabla u - u \chi(v) \nabla v \big) \cdot \nu \, \mathrm{d}s \\ \int_\Omega \psi \partial_t v \, \mathrm{d}\mathbf{x} &= -\int_\Omega d_v \nabla \psi \cdot \nabla v + \psi \big(\alpha v - s(u)\, u \big) + \int_{\partial\Omega} d_v \psi \nabla v \cdot \nu \, \mathrm{d}s \,, \end{aligned}$$

where the boundary integrals cancel out by making use of the prescribed boundary conditions (2.2.3) or (4.1.5). Hence, the final weak form reads

$$\begin{cases} \int_\Omega \psi \partial_t u \, \mathrm{d}\mathbf{x} &= -\int_\Omega \nabla \psi \cdot \big(d_u \nabla u - u \chi(v) \nabla v \big) - \psi g(u)\, u \, \mathrm{d}\mathbf{x} \\ \int_\Omega \psi \partial_t v \, \mathrm{d}\mathbf{x} &= -\int_\Omega d_v \nabla \psi \cdot \nabla v + \psi \big(\alpha v - s(u)\, u \big) \, \mathrm{d}\mathbf{x} \,. \end{cases} \tag{4.2.2}$$

The next step towards a fully discretization concerns the spatial discretization of the weak form. With the spatial discretization of the approximate solutions $u_h = u_h(t)$ and $v_h = v_h(t)$ which has been introduced in Chapter 2, i.e.,

$$\begin{cases} u_h(t) = \sum_j u_j(t) \varphi_j \\ v_h(t) = \sum_j v_j(t) \varphi_j \,, \end{cases} \tag{4.2.3}$$

we can cast (4.2.2) into a semi-discretized form. Since we employ the conventional Galerkin method, we assume that the test function-space and the space spanned by the basis functions of u_h and v_h, sometimes termed the trial function-space, are the same. This allows for testing the semi-discretized weak formulation with the basis functions, i.e., we set $\psi = \varphi_i$, and therefore, we can present the semi-discretized weak formulation by substituting several occurrences of the approximative solutions by their linear combinations (4.2.3),

$$\begin{cases} \sum_j \int_\Omega \varphi_i \varphi_j \frac{\mathrm{d}\mathbf{u}_j}{\mathrm{d}t} \, \mathrm{d}\mathbf{x} &= -\sum_j \int_\Omega \nabla \varphi_i \cdot \big(d_u \nabla \varphi_j - \varphi_j \chi(v_h) \nabla v_h \big) \mathbf{u}_j - \varphi_i g(u_h)\, \varphi_j \mathbf{u}_j \, \mathrm{d}\mathbf{x} \,, \\ \sum_j \int_\Omega \varphi_i \varphi_j \frac{\mathrm{d}\mathbf{v}_j}{\mathrm{d}t} \, \mathrm{d}\mathbf{x} &= -\sum_j \int_\Omega d_v \nabla \varphi_i \cdot \nabla \varphi_j \mathbf{v}_j + \varphi_i \big(\alpha \varphi_j \mathbf{v}_j - s(u_h)\, \varphi_j \mathbf{u}_j \big) \, \mathrm{d}\mathbf{x} \,. \end{cases} \tag{4.2.4}$$

Note that certain approximative solutions are not substituted, because they give rise to nonlinearities. Indeed, when we consider a matrix-vector form of (4.2.4) the non-substituted terms require a matrix formulation with non-constant (nonlinear) coefficients. With the following matrices which correspond to the continuous counterparts of the corresponding physical processes in the underlying model (4.1.1), e.g., diffusion, chemotaxis, proliferation and reaction terms,

$$\begin{cases} \mathbf{M}_{ij} &= \int_\Omega \varphi_i \varphi_j \, \mathrm{d}\mathbf{x} \,, \\ \mathbf{L}_{ij} &= \int_\Omega \nabla \varphi_i \cdot \nabla \varphi_j \, \mathrm{d}\mathbf{x} \,, \\ \Big[\mathbf{K}_1(v_h) \Big]_{ij} &= \int_\Omega \chi(v_h) \, (\nabla v_h \cdot \nabla \varphi_i) \, \varphi_j \, \mathrm{d}\mathbf{x} \,, \\ \Big[\mathbf{G}(u_h) \Big]_{ij} &= \int_\Omega g(u_h) \, \varphi_i \varphi_j \, \mathrm{d}\mathbf{x} \,, \\ \Big[\mathbf{S}(u_h) \Big]_{ij} &= \int_\Omega s(u_h) \, \varphi_i \varphi_j \, \mathrm{d}\mathbf{x} \,, \end{cases} \tag{4.2.5}$$

the following matrix formulation of (4.2.4) can be derived

$$\begin{cases} \mathbf{M}\dfrac{\mathrm{d}\mathbf{u}}{\mathrm{d}t} & = & -\left[d_u\mathbf{L}-\mathbf{K}_1(v_h)-\mathbf{G}(u_h)\right]\mathbf{u} \\ \mathbf{M}\dfrac{\mathrm{d}\mathbf{v}}{\mathrm{d}t} & = & -\left[d_v\mathbf{L}+\alpha\mathbf{M}\right]\mathbf{v}+\mathbf{S}(u_h)\,\mathbf{u}\,. \end{cases} \tag{4.2.6}$$

Sometimes it is more convenient to rewrite the above matrix form of the weak formulation in a block matrix form

$$\mathcal{M}\frac{\mathrm{d}\mathbf{w}}{\mathrm{d}t} \;=\; \mathcal{B}(\mathbf{w})\,\mathbf{w}\,, \tag{4.2.7}$$

where $\mathbf{w}=(\mathbf{u},\mathbf{v})^T$ denotes the block solution vector, and the left-hand side block mass matrix and right-hand side block matrix are defined as

$$\begin{aligned} \mathcal{M} &= \begin{bmatrix} \mathbf{M} & 0 \\ 0 & \mathbf{M} \end{bmatrix}, \\ \mathcal{B}(\mathbf{w}) &= \begin{bmatrix} -d_u\mathbf{L}+\mathbf{K}_1(\mathbf{v})+\mathbf{G}(\mathbf{u}) & 0 \\ \mathbf{S}(\mathbf{u}) & -d_v\mathbf{L}-\alpha\mathbf{M} \end{bmatrix}. \end{aligned}$$

Let us remark that above and for the remainder of this work it is often more convenient to use the FE coefficient vectors of the discrete solutions, $\mathbf{u},\mathbf{v}$, as arguments for the matrices with non-constant coefficients (if clear from the context, we omit the arguments).

Remark 4.1 *In (4.2.4) we have chosen a particular linearization of the chemotaxis term* $\int_\Omega \nabla\varphi_i\cdot(u_h\chi(v_h)\nabla v_h)\,\mathrm{d}\mathbf{x}$. *We linearized this term in u and obtained a discrete chemotaxis operator dependent on v_h, namely $\mathbf{K}_1(v_h)$, cf. (4.2.5). However, if we face a constant chemosensitivity, $\chi(v)=\chi$, we basically have two options to cast the above integral into a proper matrix-vector formulation. Besides the one adopted above, we can also think about linearizing the integral in v which leads to a discrete chemotaxis operator that depends on u_h, namely*

$$\left[\widehat{\mathbf{K}}_1(u_h)\right]_{ij} \;=\; \int_\Omega \chi\,(\nabla\varphi_j\cdot\nabla\varphi_i)\,u_h\,\mathrm{d}\mathbf{x}\,.$$

These two alternatives can be sketched as follows,

$$\widehat{\mathbf{K}}_1(u_h)\,\mathbf{v} \quad \xleftarrow{\textit{alternative}} \quad \int_\Omega \nabla\varphi_i\cdot(u_h\chi\nabla v_h)\,\mathrm{d}\mathbf{x} \quad \xrightarrow{\textit{current}} \quad \mathbf{K}_1(v_h)\,\mathbf{u}\,.$$

Is there a difference between those two approaches? If so, what is the most reasonable choice?

The answer to these questions is provided in the work of DeBlois [20]. Therein he studied this assignment in the context of Navier-Stokes equations, i.e., how to linearize the convection given by $(\mathbf{u}\cdot\nabla)\mathbf{u}$, where $\mathbf{u}\in\mathbb{R}^{dim}$ is a dim-dimensional velocity field. When following his argumentations, we clearly favor the linearization via $\mathbf{K}_1(v_h)\,\mathbf{u}$. In fact, DeBlois even provided particular examples where an alternative linearization leads to wrong solutions.

4.3. Temporal discretization

Given the fact that our model (4.1.1) is nonstationary we need to discretize (4.2.7) in time to obtain the final resulting discretization scheme. As we already pointed out in the beginning of this chapter, the treatment of the temporal discretization of the time derivative in (4.2.7) will be accomplished by the theta-scheme, whose application to (4.1.1) reads

$$\mathcal{M}\frac{\mathbf{w}^{n+1}-\mathbf{w}^{n}}{\delta t} \;=\; \theta\,\mathcal{B}(\mathbf{w}^{n+1})\,\mathbf{w}^{n+1}+(1-\theta)\,\mathcal{B}(\mathbf{w}^{n})\,\mathbf{w}^{n}\,.$$

After re-sorting the terms by their temporal index we end up with the (block) system

$$\left[\mathcal{M}-\theta\,\delta t\mathcal{B}(\mathbf{w}^{n+1})\right]\mathbf{w}^{n+1} \;=\; \left[(1-\theta)\,\delta t\,\mathcal{B}(\mathbf{w}^{n})+\mathcal{M}\right]\mathbf{w}^{n}\,, \tag{4.3.1}$$

which can be easily cast into the more general (and convenient) form

$$\mathcal{A}(\mathbf{w}^{n+1})\,\mathbf{w}^{n+1} \;=\; \mathbf{b}(\mathbf{w}^{n}) \tag{4.3.2}$$

with the notations

$$\begin{aligned}\mathcal{A}(\mathbf{w}^{n+1}) &= \mathcal{M}-\theta\,\delta t\,\mathcal{B}(\mathbf{w}^{n+1})\,,\\ \mathbf{b}(\mathbf{w}^{n}) &= \left[(1-\theta)\,\delta t\,\mathcal{B}(\mathbf{w}^{n})+\mathcal{M}\right]\mathbf{w}^{n}\,.\end{aligned}$$

The semi-discretized system (4.2.7) and the fully-discretized system (4.3.2) will be of special interest in the next sections. While the latter will often be referred to in order to derive particular iteration schemes (see the following up section), the semi-discretized system will be the focus in the introduction of the stabilization technique in Section 4.5.

4.4. Formulation of the iteration schemes

In the following, we will present the detailed iteration schemes that are subject in this work. Therefore, let us briefly give a little roadmap for the different upcoming methods. The particular choice of the iteration scheme depends on the users interests, e.g., in accuracy, computational resources, robustness, and the underlying model, e.g., complexity, coupling and order of nonlinearity. Particularly for chemotaxis models, Strehl *et al.* [97] initiated some preliminary comparative numerical analysis for certain schemes. Mainly, we can distinguish nonlinear and linear schemes. The latter require an encompassing nonlinear iteration, whereas the former linearize the given system a priori, i.e., these schemes work without an explicit nonlinear loop. In this current work we will only focus on one representative linearization scheme, the so called *Linearization via Extrapolation*, which will be presented in the following. Three particular nonlinear schemes will be introduced afterwards.

4.4.1. Linearization via extrapolation in time

Given a nonlinear system in the general form of (4.3.2) we mainly have two options to cope with the nonlinearity introduced by the argument of the system matrix $\mathcal{A}$. Either we tackle it by some kind of nonlinear iteration methods, which will be the focus of the proceeding section, or we approximate $\mathcal{A}(\mathbf{w}^{n+1})$ by some linear counterpart. In the course of a linearization via extrapolation (in time) the basic idea is to consider Taylor expansions of the nonlinear time-continuous terms contained in the system matrix, e.g., in our case these are the chemotaxis, growth and chemical production terms of the general PDE model (4.1.1). Let $\vartheta : \mathbb{R}_+ \to \mathbb{R}, t \mapsto \vartheta(t)$ be a twice differentiable function (in time) and let $\{t_0, t_1, t_2, \dots\}$ with an uniform step width $\delta t = t_n - t_{n-1} > 0$ be a discrete subset of $\mathbb{R}_+$. We will now approximate $\vartheta(t_{n+1})$ in terms of Taylor expansions. In order to do so, we expand ϑ centered at t_n and evaluate it at t_{n+1} and t_{n-1}:

$$
\begin{array}{rrcl}
+\left[\begin{array}{r} T_{t_n}^{\vartheta}(t_{n+1}) \qquad\qquad \\ T_{t_n}^{\vartheta}(t_{n-1}) \end{array}\right. & & \begin{array}{c} = \\ = \end{array} & \begin{array}{l} \vartheta(t_n) + \vartheta'(t_n)(t_{n+1} - t_n) + \mathcal{O}(\delta t^2) \\ \vartheta(t_n) + \vartheta'(t_n)(t_{n-1} - t_n) + \mathcal{O}(\delta t^2) \end{array} \\
\hline
\Rightarrow \quad T_{t_n}^{\vartheta}(t_{n+1}) + T_{t_n}^{\vartheta}(t_{n-1}) & & = & 2\,\vartheta(t_n) + \mathcal{O}(\delta t^2)
\end{array}
$$

Since $T_{t_n}^{\vartheta}(t_{n+1})$ and $T_{t_n}^{\vartheta}(t_{n-1})$ are second-order approximations of $\vartheta(t_{n+1})$ and $\vartheta(t_{n-1})$, respectively, we can deduce

$$
\vartheta(t_{n+1}) \quad = \quad 2\,\vartheta(t_n) - \vartheta(t_{n-1}) + \mathcal{O}(\delta t^2). \tag{4.4.1}
$$

Thus, we obtain a second-order linearization (note that the right hand side of (4.4.1) is independent of t_{n+1}) if the time stepping δt is sufficiently small (to ensure the convergence of the Taylor series expansion $T_{t_n}^{\vartheta}$).

We can now easily adopt this technique to the governing nonlinear system given above by substituting all occurrences of terms depending on $\mathbf{w}^{n+1}$ by the corresponding linear extrapolation $\mathbf{w}_{lin}^{n+1} = 2\mathbf{w}^n - \mathbf{w}^{n-1}$. Together with the matrices defined in (4.2.5) the linearized system matrix for the general model can be formulated as

$$
\mathcal{A}(\mathbf{x}_{lin}^{n+1}) \quad = \quad \begin{bmatrix} \mathbf{M} + \theta\,\delta t\,\{d_u \mathbf{L} - \mathbf{K_1}(\mathbf{w}_{lin}^{n+1}) - \mathbf{G}(\mathbf{w}_{lin}^{n+1})\} & 0 \\ -\theta\,\delta t\,\mathbf{S}(\mathbf{w}_{lin}^{n+1}) & \mathbf{M} + \theta\,\delta t\,\{d_v \mathbf{L} + \alpha \mathbf{M}\} \end{bmatrix}. \tag{4.4.2}
$$

Let us remark that this technique eliminates the strong implicit coupling between the two components of the solution vector. Hence, it can also be interpreted and implemented as a two-step solution method. Furthermore, we acknowledge that for the initial time step, i.e., $n = 0$, we can safely define $\mathbf{w}^{n-1} := \mathbf{w}^n = \mathbf{w}^0$. This way, the initial step of the linearization via extrapolation is equivalent to one step of the first-order Picard linearization, where the nonlinear iteration is only exerted once, cf. the upcoming Picard's linearization (4.4.9) in Section 4.4.3.

As a summary, the linearization via extrapolation is accompanied by two main advantages. First of all, it does not require a costly nonlinear iteration, it only needs to additionally store one solution vector. Secondary, this linearization is of second order, in contrast to a pure Picard linearization. However the overall numerical benefits, in terms of efficiency, have to be carefully revised, particularly for large time steps or in the case of dominating nonlinearities. This will be particularly addressed in the chapter of the numerical results, Chapter 5. To comply with our aim of being a first comprehensive guideline of numerical treatment for chemotaxis models, Algorithm 4.1 sketches the scheme of linearization via extrapolation. For the remainder of this work we will to this scheme as "LIN".

4.4.2. Nonlinear Richardson scheme

Before stating the nonlinear schemes in detail, Figure 4.2 already depicts a short overview of these schemes. The Newton-like scheme will not be discussed in full detail in this current work. This

Algorithm 4.1 Linearization via extrapolation in time (LIN)

```
Initialization: $\mathbf{w}^{-1} := \mathbf{w}^0$
for time step $n < n_{\max}$ do
    Build RHS: $\mathbf{b}(\mathbf{w}^n)$
    Compute linearization: $\mathbf{w}_{lin}^{n+1} = 2\,\mathbf{w}^n - \mathbf{w}^{n-1}$
    Build system matrix: $\mathcal{A}(\mathbf{w}_{lin}^{n+1})$
    Solve the linear system $\mathcal{A}(\mathbf{w}_{lin}^{n+1})\,\mathbf{w}^{n+1} = \mathbf{b}(x^n)$
end for
```

method naturally arises when the exact Jacobian should not or simply cannot be calculated. In the case of a resulting lack of a matrix-vector formulation of the underlying Jacobian-vector product, this was already considered in [97]. Let us remark that the strong decoupled Picard linearization method was already introduced in [96].

In order to linearize the governing nonlinear system (4.3.2), we employ most commonly either a Picard linearization or Newton's method. A general nonlinear Richardson scheme for system (4.3.2) can be formulated in the usual way (the superscript m denotes the nonlinear iterate)

$$\mathbf{x}_{m+1} \quad = \quad \mathbf{x}_m + \mathcal{P}^n(\mathbf{x}_m)^{-1}\mathbf{res}^n(\mathbf{x}_m), \tag{4.4.3}$$

where we define the so-called (nonlinear) *residual* as

$$\mathbf{res}^n(\mathbf{x}_m) \quad = \quad \mathbf{b}(\mathbf{x}^n) - \mathcal{A}(\mathbf{x}_m)\,\mathbf{x}_m\,, \tag{4.4.4}$$

and $\mathcal{P}^n(\mathbf{w}_m)$ represents the iteration matrix evaluated in the nonlinear iterate $\mathbf{w}_m$, corresponding to either the Picard linearization, i.e., $\mathcal{P}^n(\mathbf{w}_m) = \mathcal{A}(\mathbf{w}_m)$, or Newton's method, i.e., $\mathcal{P}^n(\mathbf{w}_m)$ being (at least close to) the exact Jacobian of $\mathcal{A}(\mathbf{w}_m)$, say $\mathcal{P}^n(\mathbf{w}_m) = \mathcal{J}(\mathbf{w}_m) \approx \mathrm{jac}\big(\mathcal{A}(\mathbf{w}_m)\big)$. Sometimes we drop the explicit arguments in the terms and simply write $\mathbf{b}^n = \mathbf{b}(\mathbf{w}^n)$ or $\mathbf{res}_m^n = \mathbf{res}^n(\mathbf{w}_m)$ as abbreviation.

To circumvent the costly task to invert $\mathcal{P}^n$ in equation (4.4.3), we use the common workaround to express (4.4.3) in two steps involving the solution of a linear system in terms of

$$\begin{aligned} i) \quad \mathcal{P}^n(\mathbf{x}_m)\,\mathbf{y} \quad &= \quad \mathbf{res}^n(\mathbf{x}_m)\,, \\ ii) \qquad \mathbf{x}_{m+1} \quad &= \quad \mathbf{x}_m + \mathbf{y}\,. \end{aligned} \tag{4.4.5}$$

Hence, after these transformations we end up with the task to solve linear systems of kind (4.4.3) multiple times to obtain the solution for the original nonlinear system at the $(n+1)^{\text{th}}$ time step, i.e., a proper nonlinear termination criterion accepts $\mathbf{w}_M$, for a certain iteration number M, as new solution, $\mathbf{w}^{n+1} := \mathbf{w}_M$.

For the remainder of this thesis, especially in Chapter 5, we will refer to the monolithic Picard's iteration as "PIC". This iteration scheme can be sketched as in the following algorithm, Algorithm 4.2.

Newton's method

In contrast to the rather simple Picard linearization it might be helpful to discuss Newton's method for the governing system (4.4.3) already here at this point. It is well known from the literature,

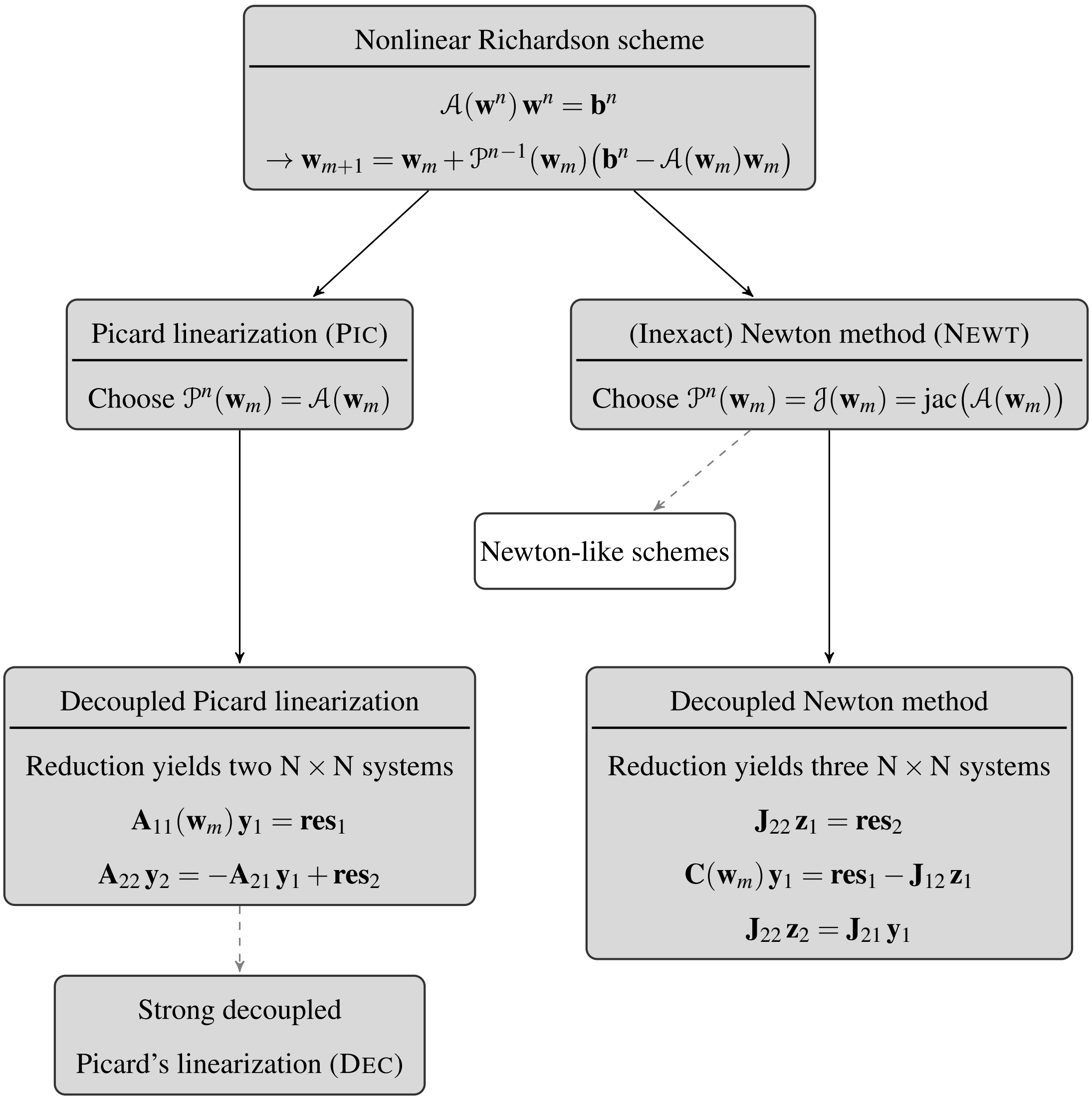

Nonlinear Richardson scheme
$\mathcal{A}(\mathbf{w}^n)\,\mathbf{w}^n = \mathbf{b}^n$
$\to \mathbf{w}_{m+1} = \mathbf{w}_m + \mathcal{P}^{n-1}(\mathbf{w}_m)\big(\mathbf{b}^n - \mathcal{A}(\mathbf{w}_m)\mathbf{w}_m\big)$
Picard linearization (PIC)
Choose $\mathcal{P}^n(\mathbf{w}_m) = \mathcal{A}(\mathbf{w}_m)$
(Inexact) Newton method (NEWT)
Choose $\mathcal{P}^n(\mathbf{w}_m) = \mathcal{J}(\mathbf{w}_m) = \mathrm{jac}\big(\mathcal{A}(\mathbf{w}_m)\big)$
Newton-like schemes
Decoupled Picard linearization
Reduction yields two N × N systems
$\mathbf{A}_{11}(\mathbf{w}_m)\,\mathbf{y}_1 = \mathbf{res}_1$
$\mathbf{A}_{22}\,\mathbf{y}_2 = -\mathbf{A}_{21}\,\mathbf{y}_1 + \mathbf{res}_2$
Decoupled Newton method
Reduction yields three N × N systems
$\mathbf{J}_{22}\,\mathbf{z}_1 = \mathbf{res}_2$
$\mathbf{C}(\mathbf{w}_m)\,\mathbf{y}_1 = \mathbf{res}_1 - \mathbf{J}_{12}\,\mathbf{z}_1$
$\mathbf{J}_{22}\,\mathbf{z}_2 = \mathbf{J}_{21}\,\mathbf{y}_1$
Strong decoupled
Picard's linearization (DEC)

Figure 4.2: Roadmap of nonlinear iteration schemes. The definitions of the involved vectors and matrices are provided in the corresponding paragraphs.

Algorithm 4.2 Monolithic Picard linearization (PIC)

```
Given the nonlinear system $\mathcal{A}(\mathbf{w}^n)\mathbf{w}^n = \mathbf{b}^n$
Initialize $\mathbf{w}^0 = \mathbf{w}^{n-1}$
while $m \leq m_{\max}$ and not converged do
    Build system block matrix $\mathcal{A}(\mathbf{w}_m)$
    Calculate block residual: $\mathbf{res}_m^n = \mathbf{b}^n - \mathcal{A}(\mathbf{w}_m)\mathbf{w}_m$
    Solve $\mathcal{A}(\mathbf{w}_m)\,\mathbf{y} = \mathbf{res}_m^n$
    Update solution: $\mathbf{w}_{m+1} = \mathbf{w}_m + \mathbf{y}$
end while
```

e.g., [22], that today there is a variety of different kinds of Newton methods for particular efficiency purposes. In our work we will restrict ourselves to so-called *inexact ordinary Newton* and *inexact Newton-like* methods. They are characterized by inexactly solving the underlying linear subsystem (4.4.5), e.g., with an iterative linear Krylov-space solver such as BICGSTAB, rather than with a direct solver, where the iteration matrix $\mathcal{P}^n(\mathbf{w}_m)$ is the exact or only an approximative

Jacobian, respectively. The latter case is employed to reduce the computational work per iteration, e.g., by dropping 'weak' terms in the exact Jacobian (*sparsing*) or by a 'close-by' Jacobian $\mathcal{P}^n(\mathbf{w}_m) = \mathrm{jac}\big(\mathcal{A}(\mathbf{z})\big)$ for all nonlinear iterations, where $\mathbf{z} \neq \mathbf{w}_m$ is fixed. However there are certainly situations where particular choices for the coefficients in the general model (4.1.1) hinder a matrix representation of the exact Jacobian system, cf. [97], which actually requires Newton-like schemes. In the concluding chapter, Chapter 6, we will briefly discuss these issues again.

Because the Jacobian is highly dependent on the particular underlying model, we will only provide it explicitly for the minimal model (4.1.2). The chemotaxis term introduces an additional FE-matrix in the associated Jacobian, namely

$$\Big[\mathbf{K}_2(u)\Big]_{ij} \;=\; \int_\Omega (\nabla\varphi_i \cdot \nabla\varphi_j)\, u \,\mathrm{d}\mathbf{x}.$$

The exact Jacobian evaluated in $\mathbf{w}_m$ can now be written as

$$\mathrm{jac}\big(\mathcal{A}(\mathbf{x}_m)\big) = -\begin{bmatrix} \mathbf{M} + \theta\,\delta t\,\big[\mathbf{L} - \mathbf{K}_1(\mathbf{v}_m)\big] & -\theta\,\delta t\,\chi\,\mathbf{K}_2(\mathbf{u}_m) \\ -\theta\,\delta t\,\mathbf{M} & \mathbf{M} + \theta\,\delta t\,\big[d_v\mathbf{L} + \mathbf{M}\big] \end{bmatrix}. \tag{4.4.6}$$

Note that in contrast to the system matrix $\mathcal{A}(\mathbf{w}_m)$ the (1,2)-block is non-zero and hence the iteration matrix loses its block triangular shape.

In Algorithm 4.3 we provided a pseudocode for Newton's method which is the basis for our implemented inexact Newton (Newton-like) iteration, referred to as "NEWT" for the remainder of this thesis. Note that if we issue a direct solver in line 5 we actually end up with the exact Newton method. Moreover, if we use an approximation of the ordinary Jacobian in line 7, we arrive at Newton-like methods.

Algorithm 4.3 Monolithic Newton method (NEWT)

```
1: Given the nonlinear system $\mathcal{A}(\mathbf{w}^n)\mathbf{w}^n = \mathbf{b}^n$
2: Initialize $\mathbf{w}_0 = \mathbf{w}^{n-1}$
3: while $m \leq m_{\max}$ and not converged do
4:     Build system block matrix $\mathcal{A}(\mathbf{w}_m)$
5:     Build block jacobian $\mathcal{J}(\mathbf{w}_m) = \mathrm{jac}\big(\mathcal{A}(\mathbf{w}_m)\big)$    ▷ approximation leads to Newton-like methods
6:     Calculate block residual: $\mathbf{res}_m^n = \mathbf{b}^n - \mathcal{A}(\mathbf{w}_m)\mathbf{w}_m$
7:     Solve $\mathcal{J}(\mathbf{w}_m)\,\mathbf{y} = \mathbf{res}_m^n$    ▷ direct solver leads to exact Newton methods
8:     Update solution: $\mathbf{w}_{m+1} = \mathbf{w}_m + \mathbf{y}$
9: end while
```

4.4.3. Decoupled approach

Instead of a monolithic treatment of the system as introduced in the preceding section, sometimes it is more convenient to decouple the discrete equations in (4.4.5). Indeed this is a very common approach when facing only weakly coupled nonlinear PDEs, special shapes of iteration matrices or striving to limit the computational expenses. In the framework of saddle-point problems the most common decoupled approach involves the so-called *Schur complement*, cf., e.g., the surveying paper of Benzi *et al.* [8]. For chemotaxis PDEs simplified versions of decoupling can already be found in [96–98]. Here we will derive a decoupled scheme that follows the idea of the Schur-complement. For reasons of comprehensibility we drop obvious indices and arguments and identify the underlying iteration matrix with

$$\mathcal{P}^n(\mathbf{w}_m) = \begin{bmatrix} \mathbf{A}_{11} & \mathbf{A}_{12} \\ \mathbf{A}_{21} & \mathbf{A}_{22} \end{bmatrix}.$$

Provided some block transformations, we can rewrite this system matrix by the following triangular factorization

$$\begin{bmatrix} \mathbf{A}_{11} & \mathbf{A}_{12} \\ \mathbf{A}_{21} & \mathbf{A}_{22} \end{bmatrix} = \begin{bmatrix} I & \mathbf{A}_{12}{\mathbf{A}_{22}}^{-1} \\ 0 & I \end{bmatrix} \begin{bmatrix} \mathbf{A}_{11} - \mathbf{A}_{12}{\mathbf{A}_{22}}^{-1}\mathbf{A}_{21} & 0 \\ 0 & \mathbf{A}_{22} \end{bmatrix} \begin{bmatrix} I & 0 \\ {\mathbf{A}_{22}}^{-1}\mathbf{A}_{21} & I \end{bmatrix}.$$

Hence, if the entire block matrix and the submatrix $\mathbf{A}_{22}$ are regular, the solution of (4.4.5) admits the following representation

$$\begin{aligned} \mathbf{y} &= \mathcal{P}^n(\mathbf{w}_m)^{-1}\mathbf{res} \\ &= \begin{bmatrix} \mathbf{I} & \mathbf{0} \\ -{\mathbf{A}_{22}}^{-1}\mathbf{A}_{21} & \mathbf{I} \end{bmatrix} \begin{bmatrix} \mathbf{C}^{-1} & \mathbf{0} \\ \mathbf{0} & {\mathbf{A}_{22}}^{-1} \end{bmatrix} \begin{bmatrix} \mathbf{I} & -\mathbf{A}_{12}{\mathbf{A}_{22}}^{-1} \\ \mathbf{0} & \mathbf{I} \end{bmatrix} \mathbf{res}, \end{aligned} \tag{4.4.7}$$

where we set $\mathbf{C} = \mathbf{A}_{11} - \mathbf{A}_{12}{\mathbf{A}_{22}}^{-1}\mathbf{A}_{21}$, which is a standard Schur complement reduction. By expanding ($\mathbf{y} = (\mathbf{y}_1, \mathbf{y}_2)^T, \mathbf{res} = (\mathbf{res}_1, \mathbf{res}_2)^T$) we obtain

$$\begin{aligned} \mathbf{y}_1 &= \mathbf{C}^{-1}(\mathbf{res}_1 - \mathbf{A}_{12}{\mathbf{A}_{22}}^{-1}\mathbf{res}_2), \\ \mathbf{y}_2 &= -{\mathbf{A}_{22}}^{-1}\mathbf{A}_{21}\mathbf{C}^{-1}(\mathbf{res}_1 - \mathbf{A}_{12}{\mathbf{A}_{22}}^{-1}b_2) + {\mathbf{A}_{22}}^{-1}\mathbf{res}_2. \end{aligned}$$

Hence, instead of solving the system (4.4.5) simultaneously, it is sufficient to solve the following equations

$$\begin{cases} \mathbf{A}_{22}\,\mathbf{z}_1 = \mathbf{res}_2, \\ \mathbf{C}\,\mathbf{y}_1 = \mathbf{res}_1 - \mathbf{A}_{12}\,\mathbf{z}_1, \\ \mathbf{A}_{22}\,\mathbf{z}_2 = \mathbf{A}_{21}\,\mathbf{y}_1, \end{cases} \tag{4.4.8}$$

where $\mathbf{z}_1$ and $\mathbf{z}_2$ are auxiliary solutions. The original solution $\mathbf{y}_2$ is re-obtained by setting $\mathbf{y}_2 = \mathbf{z}_1 - \mathbf{z}_2$.

The crucial part is the form of $\mathbf{C}$, since it explicitly contains (i) ${\mathbf{A}_{22}}^{-1}$ and (ii) the nonlinear characteristic chemotaxis contributions. For particular models/situations the shape of $\mathbf{C}$ can be readily simplified. Exemplary let us have a look on the minimal model with very low chemical diffusion, i.e., $d_v \ll 1$. In this case $\mathbf{A}_{22}$ degenerates to $\mathbf{A}_{22} \approx -(1+\theta\delta t)\mathbf{M}$ and therefore $\mathbf{C} \approx \mathbf{A}_{11} + \theta\delta t/(1+\theta\delta t)\mathbf{A}_{12}$ which simplifies the computations since ${\mathbf{A}_{22}}^{-1}$ does not have to be calculated explicitly anymore.

Remark 4.2 *The provided triangular factorization of the block system matrix is not the only option for deriving a Schur-complement-like decoupled system. Instead of inverting the submatrix*

$\mathbf{A}_{22}$, *we can also consider a factorization of kind*

$$\begin{bmatrix}\mathbf{A}_{11} & \mathbf{A}_{12}\\ \mathbf{A}_{21} & \mathbf{A}_{22}\end{bmatrix} = \begin{bmatrix}\mathbf{I} & \mathbf{0}\\ \mathbf{A}_{21}{\mathbf{A}_{11}}^{-1} & \mathbf{I}\end{bmatrix}\begin{bmatrix}\mathbf{A}_{11} & \mathbf{0}\\ \mathbf{0} & \mathbf{A}_{22}-\mathbf{A}_{21}{\mathbf{A}_{11}}^{-1}\mathbf{A}_{12}\end{bmatrix}\begin{bmatrix}\mathbf{I} & {\mathbf{A}_{11}}^{-1}\mathbf{A}_{12}\\ \mathbf{0} & \mathbf{I}\end{bmatrix}.$$

This is indeed a very popular Schur complement reduction which, amongst others, can be found in [8]. However, in order to be well-defined we have to require regularity of $\mathbf{A}_{11}$ *which is sometimes a very crucial assumption.*

For completeness, let us remark that the factorizations that involve inversion of the $\mathbf{A}_{12}$ *and* $\mathbf{A}_{21}$ *blocks could also be considered for our application. However, in common literature this is often not the case because of dimension concerns. For instance, consider the discretization of the Navier-Stokes equation where* $\mathbf{A}_{12}$ *is not necessarily a square-matrix.*

The task to choose the most suitable factorization and hence the proper Schur complement-like reduction highly depends on the underlying (chemotaxis) model and definitely give rise to further research. For the Picard linearization we will see in the next paragraph that inversion of the $\mathbf{A}_{12}$ *block is not feasible, simply because it vanishes for a Picard linearization. However, for the scope of this work, we focus on the Schur complement reduction provided in (4.4.7).*

After this general formulation of a decoupling of the system via the Schur complement reduction, we take a look on the specific nonlinear iterations. As mentioned above, we shall present Picard's linearization and Newton's method.

Picard's linearization

From the expression of the iteration matrix $\mathcal{P}^n(\mathbf{w}_m) = \mathcal{A}(\mathbf{w}_m)$ we recognize that the solution of (4.4.5) admits the more convenient representation

$$\begin{aligned}\mathbf{y} &= \mathcal{P}^n(\mathbf{w}_m)^{-1}\,\mathbf{res}\\ &= \begin{bmatrix}\mathbf{I} & \mathbf{0}\\ -{\mathbf{A}_{22}}^{-1}\mathbf{A}_{21} & \mathbf{I}\end{bmatrix}\begin{bmatrix}{\mathbf{A}_{11}}^{-1} & \mathbf{0}\\ \mathbf{0} & {\mathbf{A}_{22}}^{-1}\end{bmatrix}\mathbf{res}\,,\end{aligned}$$

and thus, the solving procedure can be rewritten as

$$\begin{cases}\mathbf{A}_{11}\,\mathbf{y}_1 = \mathbf{res}_1\,,\\ \mathbf{A}_{22}\,\mathbf{y}_2 = -\mathbf{A}_{21}\,\mathbf{y}_1 + \mathbf{res}_2\,.\end{cases} \tag{4.4.9}$$

We remark that in our considerations $\mathbf{A}_{22}$ is often well conditioned (at least under mild assumption on the mesh), such that solving for $\mathbf{y}_2$ is much cheaper than inverting $\mathbf{A}_{11}$. Additionally we might think of simplifying (4.4.9) even more by weakening the nonlinearities. Remember that the equations (4.4.9) must be solved for every nonlinear iteration. However, the system matrix of the second equation gives rise to solve this system only once outside of the nonlinear iteration loop. This *strong decoupling* has already been proposed and approved by Strehl *et al.* in [96]. In the light of this strong decoupling, the second solution block $\mathbf{v}^{n+1}$ will be provided by solving the original system (4.3.2) with an explicit treatment of the first solution block $\mathbf{u}^n$,

$$\mathbf{A}_{22}\,\mathbf{v}^{n+1} = \mathbf{b}_2(\mathbf{w}^n) - \mathbf{A}_{21}(\mathbf{u}^n)\,\mathbf{u}^n\,.$$

Herein, we explicitly pointed out the nonlinearity of the (2,1) block matrix entry. In order to solve for the first solution block $\mathbf{u}^{n+1}$, we employ a nonlinear iteration which only consists of solving for $\mathbf{y}_1$ and updating $\mathbf{u}_m$ (both representing only the first solution block) in terms of

$$\begin{array}{rcl} \mathbf{A}_{11}(\mathbf{w}_m)\,\mathbf{y}_1 & = & \mathbf{res}_1\,, \\ \mathbf{u}_{m+1} & = & \mathbf{u}_m + \mathbf{y}_1\,. \end{array}$$

For convenience, we stressed the nonlinearity of the (1,1) block matrix entry. This iteration scheme will certainly reduce the computational expense in each nonlinear iteration, because only one single N × N system needs to be solved. Note that this nonlinear iteration degenerates if the matrix is only nonlinear in the second solution block, say $\mathbf{A}_{11}(\mathbf{w}_m) = \mathbf{A}_{11}(\mathbf{v}_m)$. This is the case in generic situations, e.g., when considering the minimal model of chemotaxis (4.1.2) or the aggregation model (4.1.3). Hence, this strong decoupling only issues a non-trivial nonlinear iteration if the underlying model gives rise to u-nonlinearities in the equation for the cell concentration, e.g., stemming from kinetic terms as introduced in the kinetic model (4.1.4).

Indeed, considerations like these are the reason for the (at least theoretical) success of investigating decoupled solvers. Numerical and practical benefits and drawbacks will be discussed in detail in the chapter of the numerical results, Chapter 5. For the remainder of this thesis we will refer to this strong decoupled iteration scheme as "DEC". Let us conclude our current theoretical thoughts with a sketch of the algorithm for the strong decoupled Picard linearization of an abstract iteration matrix of the above kind, Algorithm 4.4.

Algorithm 4.4 Strong decoupled Picard linearization (DEC)

```
Given the nonlinear system $\begin{bmatrix} \mathbf{A}_{11}(\mathbf{w}^n) & 0 \\ \mathbf{A}_{21}(\mathbf{u}^n) & \mathbf{A}_{22} \end{bmatrix} \begin{bmatrix} \mathbf{u}^n \\ \mathbf{v}^n \end{bmatrix} = \begin{bmatrix} b_1(\mathbf{w}^n) \\ b_2(\mathbf{w}^n) \end{bmatrix}$
Solve $\mathbf{A}_{22}\,\mathbf{v}^{n+1} = b_2(\mathbf{w}^n) - \mathbf{A}_{21}(\mathbf{u}^n)\,\mathbf{u}^n$
Initialize $\mathbf{w}_0 = \mathbf{x}^{n-1}$
while $m \leq m_{\max}$ and not converged do
    Build current matrix $\mathbf{A}_{11}(\mathbf{x}_m)$
    Calculate residual: $\mathbf{res}_1 = b_1(\mathbf{x}^n) - \mathbf{A}_{11}(\mathbf{x}_m)\mathbf{u}_m$
    Solve $\mathbf{A}_{11}(\mathbf{x}_m)\,\mathbf{y} = \mathbf{res}_1$
    Update u iteration: $\mathbf{u}_{m+1} = \mathbf{u}_m + \mathbf{y}$
end while
```

Newton's method

For simplicity, we shall only discuss the ordinary Newton method, as approximations of the Jacobian require little more work. For an exemplary derivation of a second-order approximation of the Jacobian originating from the aggregation model (4.1.3), which will be applied in our upcoming numerical simulations, the interested reader is kindly referred to [97]. In order to introduce the ordinary Newton method we consider the system obtained from the minimal model. In the case of non-trivial/non-zero chemosensitivity and chemical consumption all blocks of the Jacobian are non-zero. Hence the structure of the decoupling in (4.4.8) is maintained. When we recapture the explicit nonlinearities of (4.4.8) we observe that only the first block row holds nonlinear contributions in terms of $\mathbf{A}_{11}(\mathbf{v}_m)$ and $\mathbf{A}_{12}(\mathbf{u}_m)$. Therefore only the second equation in (4.4.8) is explicitly nonlinear. In correspondence to the consideration for the Picard linearization this gives rise to simplify the computational calculus. However in the case of a full block system matrix the strong decoupling seems doubtful since the equations (4.4.8) are accompanied by a ‘two-way coupling’. The solution $\mathbf{z}_1$ is required for solving for $\mathbf{y}_1$, which in turn is explicitly needed for computing $\mathbf{z}_2$. Repeating the remark given in the discourse of the general decoupled scheme above, the main drawback of the decoupled Newton method is the explicit requirement of the inverse of $\mathbf{A}_{22}$ contained in the underlying system matrix $\mathbf{C}$.

4.5. Stabilization technique

A robust numerical solver for conservative physical differential equations can be mainly characterized by five design principles, namely

1. consistency,
2. algorithmic stability,
3. convergence,
4. mass conservation,
5. positivity preservation.

While the first three properties describe the necessary requirements for the pure numerical scheme (commonly consistency and stability imply the convergence of the numerical scheme), the last two principles link the quality of the scheme to the underlying physical system to model. Indeed, the last two points are essential when it comes to interpreting the provided solutions in a proper physical (and biological) setting.

Under suitable discretization restrictions, we can safely assume the convergence of a proper implementation of the standard discretization scheme presented in the preceding section. In what follows, we will focus on presenting one promising way of admitting also mass conservation and positivity preservation in order to obtain (converging) physical meaningful solutions.

Numerical solutions of standard discretization schemes as introduced in the preceding section suffer from ill-conditioned operators by adopting non-physical behavior such as negative function values or strong oscillations. One promising approach of tackling this issue was introduced by Kuzmin *et al.* [59, 62]. Therein, an *Algebraic Flux Correction* (AFC) was developed which is based on the idea of preserving the positivity of the solutions. In terms of a classical one-step method for the temporal discretization and non-negative basis functions this can be formulated as

$$\mathbf{x}^n \geq 0 \quad \Rightarrow \quad \mathbf{x}^{n+1} \geq 0\,.$$

Moreover, in the course of the development of the AFC methodology for models of incompressible (divergence-free) flow-fields without source/sink terms, the property of positivity preservation was accompanied by the consistent numerical property of being *Local Extremum Diminishing* (LED). For the readers convenience let us therefore very briefly recapitulate the basic definitions of LED frameworks.

Definition 4.1 (LED) *In the absence of source/sink terms, a discretized scheme is called LED, if local maxima/minima do not increase/decrease with time.*

One-dimensional LED schemes can also be proven to be a subset of general *Total Variation Diminishing* (TVD) schemes which strives to diminish the ‘wiggliness’ of transient solutions in terms of

$$TV(\mathbf{u}^{n+1}) \quad \leq \quad TV(\mathbf{u}^n), \quad \text{with } TV(\mathbf{u}) = \int_\Omega |\partial_x \mathbf{u}|\, dx.$$

For detailed references the interested reader is kindly referred to Jameson [51] and the references therein.

We remark that the chemotaxis velocity is not divergence-free, which gives rise to a zero-order reaction term in the equation for u. Moreover, interesting applications of kinetic terms, e.g., model (4.1.4), induce (cell) growth and death which do not yield mass conservation in general. In Section 4.5.4 we will see that the AFC concept is still applicable and turns out to be a promising approach.

In the following we will describe the design principles of an AFC scheme with a particular focus on the embedding of such a scheme into the final resulting discretized nonlinear system for our general model of chemotaxis (4.1.1).

4.5.1. The concept of AFC

In our context AFC can practically be considered as a blending strategy of the so-called *high-order* scheme, i.e., the previously introduced standard Galerkin scheme, and a numerically dissipative counterpart, usually referred to as *low-order* scheme or formulation. Because the latter deteriorates the overall second order accuracy of standard $\mathcal{Q}_1$ Galerkin schemes, we will adopt this nomenclature for the two differently accurate schemes. Moreover, an intermediate accuracy will be referred to as being of *mixed order* for the remainder of this work.

To give a first very brief introductory to AFC, its concept can be sketched as a two-step methodology. In a first step, the original high-order scheme is transformed into a low-order scheme that introduces as much numerical dissipation as it is required to ensure ‘stable solutions’, a more precise criterion is given shortly. In a second step, the overdissipative parts of the low-order scheme that cause excessive smearing of the solution profile are canceled by blending the low-order with the high-order solution pointwise.

A major design criterion for a proper stabilized transient discretization scheme for (4.3.2) is the preservation of positivity of the numerical solution. Remember that for chemotaxis models, the solution $\mathbf{x} = (\mathbf{u}, \mathbf{v})$ represents the population of cells and the density of chemical agents, respectively, so that positivity preservation is naturally recommended. The following theorem provides a handy condition for the monotonicity of matrices and is also often used to prove *discrete maximum principles* (DMP).

Theorem 4.1 (M-matrix property, cf. [103]) *If $\mathcal{A} = \{a_{ij}\}$ is an irreducibly diagonal dominant matrix with $a_{ii} > 0$ for all i and $a_{ij} \leq 0$ for all $j \neq i$, then $\mathcal{A}$ is nonsingular and its inverse yields $\mathcal{A}^{-1} \geq 0$ component-wise.*

Hence for the system (4.3.2), a sufficient condition for positivity preservation reads

Corollary 4.1 *Let $\mathbf{b}(\cdot) \geq 0$ for all nonnegative arguments and $\mathcal{A}$ satisfy the conditions of Theorem 4.1, then the system (4.3.2) yields positivity preservation.*

Notably, we recognize that, under mild assumptions on the discretization, system (4.3.2) satisfies the above conditions in the case of a dominant diffusion term and a *lumping of the mass matrix* $\mathbf{M} = \{m_{ij}\}$ in terms of

$$\mathbf{M}_L \quad := \quad \operatorname{diag}\{m_i\}, \quad m_i = \sum_j m_{ij} \,. \tag{4.5.1}$$

This stems from the fact that the discrete diffusion operator (consider for instance the common five-point stencil) already satisfies the M-matrix property for the standard Galerkin approach.

However generally speaking, the M-matrix properties are often violated by standard Galerkin discretizations. In our specific case of chemotaxis-driven PDEs, the main 'troublemaker' can be identified by the discrete contributions of chemotaxis, i.e., $\mathbf{K}_1, \mathbf{K}_2$. Moreover, in a fully discretized system the M-matrix properties is also violated by large time steps. It is well known that even if all coefficients of the semi-discrete scheme (4.2.7) have the correct sign, a CFL-like condition must hold for the fully discretized system to possess the M-matrix property [61, section 3.3.3].

So the very first step towards a full AFC scheme is to transform the underlying system (4.4.5) into a positivity preserving discretization of low-order. Beside the DMP condition we also require the transformation to hold the law of mass conservation, for that the main physical nature of the solution should be maintained. In particular Kuzmin *et al.* suggested artificial diffusion to fulfill the requirements of DMP and mass conservation. Furthermore their transformation yields the LED criterion. To recapture high-order where possible, we apply a limited amount of compensating fluxes which will still yield the DMP, mass conservation and LED criterion. The compensating antidiffusive fluxes can be either incorporated into the nonlinearity (nonlinear AFC-schemes) or they can be linearized, explicitly computed and applied in a post-processing fashion. In this work we will focus on the explicit application of compensating antidiffusion, since it is an efficient variant and has already been approved in the context of chemotaxis models [96, 98]. The limiter coefficients are defined by the so-called *Zalesak limiter* in a symmetric fashion such that mass conservation is also preserved in the limiting step.

In the following course of demonstrating the above methodology it is helpful to distinguish between the semi- and the fully-discretization of the problem at hand. We will therefore follow the ideas of Kuzmin in [60] by first take a look on the spatial discretization, the semi-discretized system, before going on to the temporal discretization, the fully discretized system. Hence, we take some steps back from the underlying system (4.4.5) and identify its corresponding semi-discretized system by the general formulation (4.2.7). Moreover, for the sake of simplicity, we will derive all schemes in the case of only one single variable, i.e., we consider a semi-discretized system of kind

$$\mathbf{M}\frac{\mathrm{d}\mathbf{u}}{\mathrm{d}t} = \mathbf{B}(\mathbf{u})\,\mathbf{u}, \tag{4.5.2}$$

Herein the matrices are explicitly no block matrices and $\mathbf{u}$ corresponds to one single solution variable only.

4.5.2. The low-order formulation

Before we turn to the low-order scheme for the semi-discretized system (4.5.2), some introductory words are advisory, since the following derivations use terms that might be misleading for certain readers. As we will outline in the numerical state of art in Chapter 5 positivity preserving schemes have already been developed by several authors. Mainly, these authors used *Upwinding techniques* to approximate the fluxes generated by the chemotaxis term in an adequate manner. In this context let us remark that Upwinding is equivalent to adding *artificial diffusion*. Since we will make use of both of these terms, let us briefly recapture the reason for this ambivalent usage, which is taken from Brookes and Hughes [13].

For simplicity, let us consider a straightforward finite difference (FD) discretization of the one dimensional scalar stationary convection-diffusion model

$$u''(x) \quad = \quad c\,u'(x), \qquad \text{for}\, x \in \Omega = (0,l)\,, \tag{4.5.3}$$

where $c > 0$ is the constant velocity. Note that the derivatives in this ordinary differential equation are denoted by primes, u' and u''. Let the (spatial) discretization be uniform with step width $\delta h = l/i_{\max}$, $i_{\max}$ denoting an integer, and the discretized solutions be indexed by the discretized interval under consideration, i.e., $u(i\,\delta h)$ in the continuous space corresponds to u_i in the discrete space. The standard Upwind discretization for the convective part, i.e., along the positive stream, reads

$$u'(i\,\delta h) \quad \approx \quad (u_i - u_{i-1})/\delta h\,,$$

and the standard central difference scheme for the convection and the diffusion read

$$\begin{aligned} u'(i\,\delta h) \quad &\approx \quad (u_{i+1} - u_{i-1})/(2\delta h)\,, \\ u''(i\,\delta h) \quad &\approx \quad (u_{i+1} - 2u_i + u_{i-1})/\delta h^2\,. \end{aligned}$$

Now we can relate the approach of artificial diffusion with the one of Upwinding for the convection term by means of

$$\begin{array}{rcll} & \left[\right. & (u_{i+1} - u_{i+1})/(2\delta h) & \text{(central difference for the convection)} \\ -(\delta h/2)\cdot & \left[\right. & (u_{i+1} - 2u_i + u_{i-1})/\delta h^2 & \text{(additional artificial diffusion)} \\ \hline & = & (u_i - u_{i-1})/\delta h & \text{(Upwinding)} \end{array} \tag{4.5.4}$$

Thus, if we employ the central differences for the convection term and add as much diffusion as $\delta h/2$ to this discretized operator, then we obtain the standard Upwind discretization of the convection term. Note that the introduced artificial diffusion has to be shifted to the right hand side of (4.5.3), hence, there is a negative sign in the relation (4.5.4).

In terms of this interpretation, the terms of Upwind schemes and artificial diffusion approaches will be used interchangeable for the remainder of this work.

Let us now turn back to the low-order formulation and derive the low-order scheme for the semi-discretized system (4.5.2). It is a very common practice to approximate the (consistent) mass matrix $\mathbf{M}$ by its diagonal counterpart in terms of the lumped mass matrix, cf. (4.5.1). Although this lumping can notably affect the accuracy of the scheme, the favorable practical properties of a diagonalized mass matrix, such as its simple inverse, mostly dominate. Moreover we will see that the loss in phase accuracy, which is caused by the lumped mass matrix, will be countered by the subsequently introduced flux correction. For the formulation of the low-order scheme, the diagonalization of the consistent mass matrix is of particular interest, since it cancels out entries that cause a threat to the M-matrix properties and provides an easy proof for the positivity-preservation of the semi-discretized scheme. Let us additionally remark that diagonal mass matrices also arise naturally, e.g., for the nonconforming Crouzeix-Raviart element or for a special choice of the underlying quadrature rule. Here we will only consider the 'artificial' diagonalization of the mass matrix via (4.5.1). For detailed discussions on this kind of mass lumping, the interested reader is referred to the short surveying paper of Wendland and Schulz [105] and their references therein.

Besides the lumping we take care of the 'trouble-making' entries, $b_{ij} < 0$ for $j \neq i$ with $\mathbf{B}(\mathbf{u}) = \{b_{ij}\}$, by adding a symmetric artificial diffusion operator $\mathbf{D}(\mathbf{u}) = \{d_{ij}\}$. We drop the nonlinear dependencies when they can easily deduced from the context. For convenient reasons, this operator should provide a zero row/column sum. Kuzmin *et al.* proposed the following coefficients

$$d_{ij} = max\{-b_{ij}, 0, -b_{ji}\} \geq 0, \quad j \neq i \qquad \text{and} \quad d_{ii} = -\sum_{j\neq i} d_{ij}. \tag{4.5.5}$$

Let us first remark that a possible nonlinearity of $\mathbf{B}$ also imposes a nonlinearity in $\mathbf{D}$. We note further that this definition is just enough to eliminate negative entries, hence, we have $\widetilde{b}_{ij} := b_{ij} + d_{ij} \geq 0$. However, we acknowledge that the symmetry causes overdiffusive entries which has to be rectified. Before turning to this assignment, let us quickly prove that the above choice of the modified transport operator $\widetilde{\mathbf{B}} = \mathbf{B} + \mathbf{D}$ indeed also yields the LED criterion.

Together with the upper definitions the semi-discretized system is transformed into

$$\mathbf{M}_L \frac{\mathrm{d}\mathbf{u}}{\mathrm{d}t} = \widetilde{\mathbf{B}}(\mathbf{u})\mathbf{u}, \tag{4.5.6}$$

Under the assumption that $\mathbf{B}(\mathbf{u})$ has zero row sum (e.g., we assume incompressibility of a corresponding flow-field) the i^{th} row reads

$$m_i \frac{\mathrm{d}\mathbf{u}_i}{\mathrm{d}t} = \sum_{j\neq i} \widetilde{b}_{ij}(\mathbf{u}_j - \mathbf{u}_i). \tag{4.5.7}$$

Let $\mathbf{u}_i$ be a local maximum/minimum, i.e., $\mathbf{u}_j - \mathbf{u}_i \leq 0$ or ≥ 0 for all $j \neq i$, respectively (note that we implicitly assume that the FE matrices have only local stencils, i.e., $b_{ij} = 0$ if the nodes i and j are not nearest neighbors). With $\widetilde{b}_{ij} \geq 0$ for all $j \neq i$ we obtain

$$\begin{aligned} & m_i^{-1} \widetilde{b}_{ij}(\mathbf{u}_j - \mathbf{u}_i) \leq 0 \quad (\text{or } \geq 0), \quad \forall j \neq i \\ \Rightarrow \quad & \frac{\mathrm{d}\mathbf{u}_i}{\mathrm{d}t} \leq 0 \quad (\text{or } \geq 0). \end{aligned}$$

That is, beside the positivity, we see that local extrema cannot be amplified by the scheme. Unfortunately a theorem by Godunov [35] states that linear *monotonicity-preserving* schemes, which particularly contain our approach (4.5.6), can at most be first-order accurate, hence it is termed low-order scheme. To re-obtain second-order accuracy where possible, nonlinear limiters are required (even if the governing PDE model is linear).

Remark 4.3 *In correspondence with a simple one dimensional scalar example of Kuzmin and Turek in [63] we see that the definition of the discrete upwinding term (4.5.5) leads to the standard upwind method and hence is of first-order accuracy.*
To the best of our current knowledge, however, there is no notably theoretical foundation of the (first-order) accuracy of the definition (4.5.5) by means of the resulting semi-discretized scheme (4.5.6) in the context of a general discretization in space, e.g., FE in two dimensions.

4.5.3. The explicit flux correction

Now that we have a semi-discretized low-order scheme of kind (4.5.6) at hand, this current subsection is devoted to the challenge of defining a suitable antidiffusive correction to re-obtain high order where possible. In essence the (common) idea is to add a proper correction term $\bar{\mathbf{f}}$ to the right hand side of (4.5.6). A straightforward choice for this term is derived from the difference between the residuals of (4.5.6) and the original high-order Galerkin scheme (4.5.2), i.e.,

$$\mathbf{f} = (\mathbf{M}_L - \mathbf{M}) \frac{\mathrm{d}\mathbf{u}}{\mathrm{d}t} - \mathbf{D}(\mathbf{u})\mathbf{u}. \tag{4.5.8}$$

Together with the symmetry of the above matrices and an edge-wise decomposition, the residual error yields the following component-wise representation

$$f_i \;=\; \sum_{j\neq i} f_{ij} \qquad \text{with } f_{ij} = -f_{ji},$$

where the raw antidiffusive fluxes from node j to node i are denoted by f_{ij} and can be written as

$$f_{ij} \;=\; \left[m_{ij}\frac{\mathrm{d}}{\mathrm{d}t} + d_{ij}\right](u_i - u_j). \tag{4.5.9}$$

In order to control the amount of raw antidiffusive flux that will return to the scheme, we limit the single contributions of (4.5.9) by symmetric solution-dependent correction factors $\alpha_{ij} \in [0,1]$, i.e., the AFC scheme reads

$$\mathbf{M_L}\frac{\mathrm{d}\mathbf{u}}{\mathrm{d}t} \;=\; \widetilde{\mathbf{B}}\mathbf{u} + \bar{\mathbf{f}} \qquad \text{with } \overline{f}_i = \sum_{j\neq i} \alpha_{ij} f_{ij}. \tag{4.5.10}$$

Let us stress that indeed the values of α_{ij} balance the (artificial) diffusivity of the scheme, e.g., for $\alpha_{ij} = 0$ we arrive at the low-order scheme and for $\alpha_{ij} = 1$ we re-obtain high order. Additionally, with a solution-dependent choice of these correction factors, we have a scheme at hand that is (in correspondence to Godunov) potentially more than first-order accurate, since it is nonlinear overall.

In the course of the temporal discretization of (4.5.10) with the theta-scheme, we will reveal another key-property of AFC schemes. Therefore let us briefly provide the high- and low-order fully discretized schemes,

$$\left[\mathbf{M} - \theta\,\delta t\,\mathbf{B}^{n+1}\right]\mathbf{u}^{n+1} \;=\; \left[\mathbf{M} + (1-\theta)\,\delta t\,\mathbf{B}^{n}\right]\mathbf{u}^{n}, \tag{4.5.11}$$

$$\left[\mathbf{M}_L - \theta\,\delta t\,\widetilde{\mathbf{B}}^{n+1}\right]\mathbf{u}^{n+1} \;=\; \left[\mathbf{M}_L + (1-\theta)\,\delta t\,\widetilde{\mathbf{B}}^{n}\right]\mathbf{u}^{n}. \tag{4.5.12}$$

Here we use the abbreviation $\mathbf{B}^n = \mathbf{B}(\mathbf{u}^n)$. Referring to (4.5.8) and (4.5.9) the corresponding raw antidiffusive fluxes (here explicitly with respect to the old and current solution) admit the following decomposition

$$\begin{aligned} f_{ij}(\mathbf{u}^{n+1},\mathbf{u}^n) \;=\;& m_{ij}(u_i^{n+1} - u_j^{n+1}) - m_{ij}(u_i^n - u_j^n) \\ &+ \delta t\left[\theta\, d_{ij}^{n+1}(u_i^{n+1} - u_j^{n+1}) + (1-\theta)\, d_{ij}^n(u_i^n - u_j^n)\right]. \end{aligned} \tag{4.5.13}$$

Hence, the fully discretized AFC scheme is formulated as

$$[\mathbf{M}_L - \theta\,\delta t\,\widetilde{\mathbf{B}}^{n+1}]\mathbf{u}^{n+1} \;=\; [\mathbf{M}_L + (1-\theta)\,\delta t\,\widetilde{\mathbf{B}}^{n}]\mathbf{u}^{n} + \bar{\mathbf{f}}(\mathbf{u}^{n+1},\mathbf{u}^n). \tag{4.5.14}$$

Here, we easily recognize that this scheme is implicit in time even if we choose $\theta = 0$. Moreover, this implies that even in the case of linear system matrices, i.e., $\mathbf{B}^n = \mathbf{B}$, we eventually have to solve a nonlinear system. This is a main drawback of this formulation. However, amongst others, there is a linearization technique for these kinds of AFC schemes that does not necessitates a nonlinear treatment of the antidiffusive fluxes (4.5.13). In [60], Kuzmin introduced an explicit correction of a "transported and diffused" end-of-step solution, as in the case of classical diffusion-antidiffusion methods. His principle idea was to compute one low-order solution per nonlinear iteration, e.g., following (4.5.12), and correct its nonlinear converged counterpart, say $\mathbf{u}^L$, outside of the nonlinear iteration via

$$\mathbf{M}_L\mathbf{u}^{n+1} \;=\; \mathbf{M}_L\mathbf{u}^L + \delta t\,\bar{\mathbf{f}}(\mathbf{u}^L,\mathbf{u}^n). \tag{4.5.15}$$

Herein, the corresponding raw antidiffusive fluxes are defined as,

$$f_{ij} \quad = \quad m_{ij}(\dot{u}_i^L - \dot{u}_j^L) + d_{ij}^L(u_i^L - u_j^L), \tag{4.5.16}$$

where we use the following approximation for the time derivative $\dot{\mathbf{u}}^L = (\dot{u}_1^L, \ldots, \dot{u}_{\mathrm{N}}^L)$ stemming from the semi-discretized system (4.5.6),

$$\dot{\mathbf{u}}^L \quad = \quad {\mathbf{M}_L}^{-1}[\mathbf{B}^L \mathbf{u}^L]\,. \tag{4.5.17}$$

Note that since $\mathbf{M}_L$ is a diagonal matrix, this value can be directly computed without actually solving a linear system.

We saw that the whole process of solving the presented AFC scheme for a nonlinear model of type (4.5.11) can be accomplished by nonlinearly solving the low-order counterpart (4.5.12) and correcting the obtained overdiffusive solution only once per time step. The additional numerical costs are hence mainly determined by the computation of the artificial diffusion $\mathbf{D}$ once per nonlinear iteration and by the construction of the antidiffusive fluxes $\alpha_{ij} f_{ij}$ once per time step. Note that indeed the nonlinear solution of the low-order scheme (4.5.12) is usually less expensive than the original high-order Galerkin scheme (4.5.11), because the latter scheme does not satisfy the conditions of Corollary 4.1 in general.

Let us notice that in the light of Corollary 4.1 and the fully discretized low-order scheme (4.5.12) the application of the one step theta-scheme can lead to a CFL-like restriction on the time stepping. In fact, we remark that the scheme does not generally satisfy the conditions of the corollary if $\theta < 1$, since the right-hand side of (4.5.12) is not guaranteed to be positive. In this case a proper CFL-like condition must hold for the corresponding discretization, cf. [61, theorem 3.29].

Antidiffusive flux limiter

The goal of a proper choice of the correction factors $\alpha_{ij} \in [0, 1]$ is to allow as less artificial diffusion as possible in terms of positivity-preservation. We recall that $\alpha_{ij} = 0$ and $\alpha_{ij} = 1$ refer to the low-order and high-order reconstructions, respectively. In the following we present a symmetric limiting strategy that was proposed by Zalesak and approved to keep the solutions from exceeding local maxima and minima, the interested reader is kindly referred to, e.g., Kuzmin *et al.* [59, 62] for details.

0. In a pre-limiting step we cancel the fluxes that would impose further diffusivity and tend to flatten the solution profiles instead of steepening them. The former is clearly undesirable, hence, we nullify all such fluxes that point in the same direction as the solution fluxes, i.e.,

$$f_{ij} = 0, \qquad \text{if } f_{ij}(u_j^L - u_i^L) > 0\,.$$

1. In the first step we calculate all positive and negative antidiffusive fluxes into node i,

$$P_i^+ = \sum_{j \neq i} \max\{0, f_{ij}\}, \quad P_i^- = \sum_{j \neq i} \min\{0, f_{ij}\}\,.$$

2. In order to keep the solutions from exceeding local maxima and minima, we compute the distance to them,

$$Q_i^+ = \max\left\{0, \max_{j \neq i}(u_j^L - u_i^L)\right\}\,, \quad Q_i^- = \min\left\{0, \min_{j \neq i}(u_j^L - u_i^L)\right\}\,.$$

3. Correspondingly, to limit the net increase to node i, we calculate the maximal correction factors,
$$R_i^+ = \min\left\{1, \frac{m_i Q_i^+}{\delta t P_i^+}\right\}, \quad R_i^- = \min\left\{1, \frac{m_i Q_i^-}{\delta t P_i^-}\right\}.$$

4. With taking care of the symmetry of the final correction factors we define,
$$a_{ij} = \begin{cases} \min\{R_i^+, R_j^-\}, & \text{if } f_{ij} > 0\,, \\ \min\{R_i^-, R_j^+\}, & \text{otherwise.} \end{cases}$$

Remark 4.4 *Notably, all the derivations and modifications that we introduced above are of algebraic character, i.e., they work on the involved discrete matrices and vectors directly. Whereas many alternative stabilization approaches modify the governing model on the continuous level by adding highly problem dependent stabilization parameters, AFC does not require any additional stabilization parameters. The paradigm of AFC does not depend on the origin of the underlying (non-)linear system. In other words, AFC is very flexible in terms of also being applicable to different discretizations schemes such as finite volumes (FV), FD or discontinuous Galerkin methods (DG). In these cases the limiting techniques, which also crucially depend on the particular underlying model, requires some modifications.*

4.5.4. Application to chemotaxis

Here, we focus on the application of AFC for the two nonlinear Richardson schemes and leave the corresponding straightforward formulation for the linearization via extrapolation to the reader. The application of AFC for the decoupled scheme (4.4.8) can also be readily derived from the upcoming formulations for the nonlinear Richardson schemes.

The subsequent considerations take into account that the application of the AFC scheme can also be directly addressed to the governing fully discretized system rather than its semi-discretized counterpart. We will therefore first of all discuss a proper transformation of system (4.4.5) into its low-order equivalent. After obtaining the corresponding overdiffusive, nonlinearly converged solution we restore the high order, where possible, via the explicit flux correction. For the readers convenience Figure 4.3 depicts these steps. Again for reasons of simplicity we will restrict the upcoming derivations to the minimal model of chemotaxis.

In our case, the velocity field is given by the gradient of the chemoattractant, i.e., $\vec{\mathbf{v}} = \nabla \mathbf{v}$, and does not yield incompressibility in general. Particularly in terms of equation (4.5.6), this implies that we cannot assume $\mathbf{B}(\mathbf{u})$ having a zero row sum and hence, we cannot cast our discrete system into the form (4.5.7), i.e., our entire numerical scheme is not LED. However, it can still be positivity-preserving if a suitable CFL-like condition holds for the explicit and implicit part. A more detailed consideration of our application reveals that, in fact, the discretized chemotaxis flux can be decomposed in a LED and a non-LED part, which allows us to cast the flux at least partially into equation (4.5.7), cf. [39].

$$\begin{aligned}\Big(\mathbf{B}(\mathbf{u})\mathbf{u}\Big)_i &= \sum_j b_{ij} u_j \\ &= \underbrace{\sum_{j \neq i} b_{ij}(u_j - u_i)}_{\text{LED part}} + \underbrace{\sum_j b_{ij} u_i}_{\text{non-LED part}}\end{aligned}$$

If the row sum of $\mathbf{B}(\cdot)$ is negative, the non-LED part poses hazards to the overall LED property, whereas a positive row sum is harmless. From the modelling point of view, the non-LED part is the reason for local accumulation of the chemotaxing entity and must not be neglected. For the remainder of this chapter we must therefore keep in mind that the LED property is only partially preserved for chemotaxis problems. The positivity, however, can still be entirely preserved when a suitable CFL-like condition holds. This condition must hold for both the explicit and the implicit Euler, because of the possibly varying signs of entries of the discrete chemotaxis operator, i.e., $\mathbf{K}_1$.

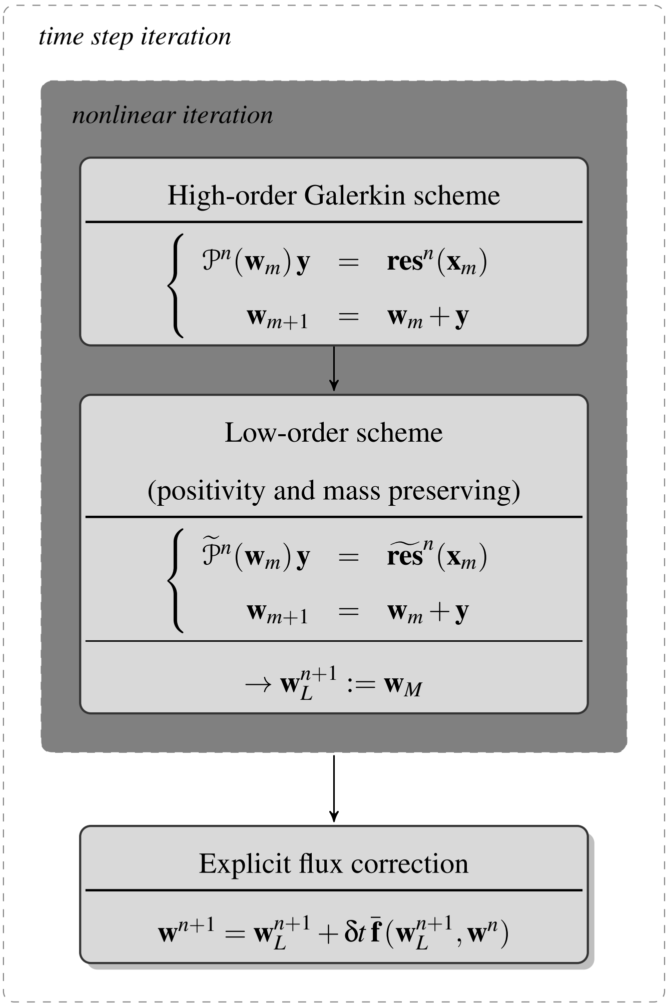

time step iteration
nonlinear iteration
High-order Galerkin scheme
$\left\{\begin{array}{rcl}\mathcal{P}^n(\mathbf{w}_m)\,\mathbf{y} &=& \mathbf{res}^n(\mathbf{x}_m)\\ \mathbf{w}_{m+1} &=& \mathbf{w}_m+\mathbf{y}\end{array}\right.$
Low-order scheme
(positivity and mass preserving)
$\left\{\begin{array}{rcl}\widetilde{\mathcal{P}}^n(\mathbf{w}_m)\,\mathbf{y} &=& \widetilde{\mathbf{res}}^n(\mathbf{x}_m)\\ \mathbf{w}_{m+1} &=& \mathbf{w}_m+\mathbf{y}\end{array}\right.$
$\rightarrow \mathbf{w}_L^{n+1} := \mathbf{w}_M$
Explicit flux correction
$\mathbf{w}^{n+1} = \mathbf{w}_L^{n+1} + \delta t\,\bar{\mathbf{f}}(\mathbf{w}_L^{n+1},\mathbf{w}^n)$

Figure 4.3: Schematic outline of the AFC for a time-dependent system.

Low-order formulation

When applying the concept of AFC to our governing chemotaxis system (4.4.5), it is useful to distinguish the schemes according to the nonlinear treatment, e.g., either via the Picard linearization or Newton's method. Definitely, the latter alternative needs special handling (note that the artificial diffusion operator is not differentiable) and therefore we start with the former variant.

Preliminary studies revealed that under mild assumptions on the geometric properties of the mesh (no sharp angles, moderate aspect ratios) the most DMP and LED criterion violating terms act on the first block row in (4.4.5), i.e., the u equation. It is therefore sufficient to only manipulate this block row in terms of adding artificial diffusion.

Low-order formulation for Picard's linearization

With recalling the iteration matrix for the Picard linearization

$$\mathcal{P}^n(\mathbf{w}_m) = \mathcal{A}(\mathbf{w}_m)\,,$$

we define the (approximative) low-order system of (4.4.5) by

$$\widetilde{\mathcal{A}}(\mathbf{w}_m)\,\mathbf{y} \;=\; \widetilde{\mathbf{res}}^n(\mathbf{x}_m)\,, \tag{4.5.18}$$

where the transformed iteration matrix and residual read

$$\begin{aligned}\widetilde{\mathcal{A}}(\mathbf{x}_m) &= \begin{bmatrix} \mathbf{M}_L + \theta\,\delta t\,\left[\mathbf{L} - (\mathbf{K}_1(\mathbf{v}_m) + \mathbf{D})\right] & 0 \\ -\theta\,\delta t\,\mathbf{M} & \mathbf{M} + \theta\,\delta t\,\left[d_v\,\mathbf{L} + \mathbf{M}\right] \end{bmatrix}, \\ \widetilde{\mathbf{res}}^n(\mathbf{x}_m) &= \widetilde{\mathbf{b}}(\mathbf{x}^n) - \widetilde{\mathcal{A}}(\mathbf{x}_m)\,\mathbf{x}_m \end{aligned}$$

with

$$\widetilde{\mathbf{b}}(\mathbf{x}^n) = \begin{bmatrix} \mathbf{M}_L\,\mathbf{u}^n - (1-\theta)\,\delta t\,\left[\mathbf{L} - (\mathbf{K}_1(\mathbf{v}^n) + \mathbf{D})\right]\mathbf{u}^n \\ \mathbf{M}\,\mathbf{v}^n - (1-\theta)\,\delta t\,\left[d_v\,\mathbf{L}\,\mathbf{v}^n + \mathbf{M}\,\mathbf{v}^n - \mathbf{M}\,\mathbf{u}^n\right] \end{bmatrix}.$$

The artificial diffusion operator is constructed in correspondence to $\mathbf{K}_1(\cdot)$

$$d_{ij} = d_{ij}(\cdot) = max\{-k_{1_{ij}}(\cdot), 0, -k_{1_{ji}}(\cdot)\} \geq 0, \quad j \neq i \qquad \text{and} \quad d_{ii} = -\sum_{j \neq i} d_{ij}.$$

From this formulation of the low-order system the reader can easily derive the corresponding system for the decoupled scheme (4.4.8) since the modifications only concern the first block row, i.e., only the second equation in (4.4.8) requires corresponding modifications.

Low-order formulation for Newton's method

For Newton's method, the proper definition of a low-order iteration matrix is a bit more delicate. Basically, we firstly have to transform the high-order system matrix $\mathcal{A}(\mathbf{w}_m)$ into its low-order counterpart $\widetilde{\mathcal{A}}(\mathbf{w}_m)$. Secondly, we have to approximate its low-order Jacobian. Note that this will naturally lead to a Newton-like method (rather than the ordinary Newton method), since the discrete upwinding process is not differentiable in the classical sense. This renders the assembly of the (approximated) Jacobian a computationally challenging task.

Let us turn to the derivation of the low-order scheme in more detail. As the underlying Jacobian can only be approximated, let us denote by

$$\widetilde{\mathcal{J}}(\mathbf{w}_m) = \begin{bmatrix} \widetilde{\mathbf{J}_{11}}(\mathbf{w}_m) & \widetilde{\mathbf{J}_{12}}(\mathbf{w}_m) \\ \widetilde{\mathbf{J}_{21}}(\mathbf{w}_m) & \widetilde{\mathbf{J}_{22}}(\mathbf{w}_m) \end{bmatrix}$$

the second-order divided difference approximation of the low-order Jacobian, i.e.,

$$\left\{\left[\widetilde{\mathcal{J}}(\mathbf{w}_m)\right]_{ij}\right\} \approx \frac{\left[\widetilde{\mathcal{A}}(\mathbf{w}_j^+)\mathbf{w}_j^+\right]_i - \left[\widetilde{\mathcal{A}}(\mathbf{w}_j^-)\mathbf{w}_j^-\right]_i}{2\sigma}$$

with $\mathbf{w}_j^{\pm} = \mathbf{w}_m \pm \sigma\,\mathbf{e}_j$, $\mathbf{e}_j$ being the j^{th} unit vector and σ being some relative deviation parameter whose particular choice can crucially effect the performance of the Newton-like method.
Particularly for our system it is more comprehensive to rewrite the approximation of the Jacobian as

$$\left[\widetilde{\mathcal{J}}(\mathbf{w}_m)\right]_{ij} = \left[\frac{\widetilde{\mathcal{A}}(\mathbf{w}_j^+) - \widetilde{\mathcal{A}}(\mathbf{w}_j^-)}{2\sigma}\mathbf{w}_m\right]_i + \left[\frac{\widetilde{\mathcal{A}}(\mathbf{w}_j^+) + \widetilde{\mathcal{A}}(\mathbf{w}_j^-)}{2}\mathbf{e}_j\right]_i . \tag{4.5.19}$$

It is useful to look a little more carefully on this expression, since the second block row of $\widetilde{\mathcal{A}}(\cdot)$ is indeed linear. Since N denotes the number of degrees of freedom per variable, our entire system is a $2\mathrm{N} \times 2\mathrm{N}$ system. The first term in (4.5.19) vanishes for $i > \mathrm{N}$ and the second one reduces to $\left[(-\theta\,\delta t\,\mathbf{M} \quad \mathbf{M} + \theta\,\delta t\,\{\mathbf{L}+\mathbf{M}\})\,\mathbf{e}_j\right]_i$. Carefully rewriting every single block contribution to the block Jacobian we end up with the following blocks

Contributions to $\widetilde{\mathbf{J}_{11}}(\mathbf{w}_m)$ For the indices $1 \le i, j \le \mathrm{N}$ the only non-zero contribution is given by the second term in (4.5.19), i.e.,

$$\begin{aligned}\left[\widetilde{\mathbf{J}_{11}}(\mathbf{w}_m)\right]_{ij} &= \left[\frac{\widetilde{\mathbf{A}_{11}}(\mathbf{w}_j^+) + \widetilde{\mathbf{A}_{11}}(\mathbf{w}_j^-)}{2}\right]_{ij} \\ &= \left[\mathbf{M}_L + \theta\,\delta t\left(\mathbf{L} - 0.5\widetilde{\mathbf{K}_1}(\mathbf{w}_j^+) - 0.5\widetilde{\mathbf{K}_1}(\mathbf{w}_j^-)\right)\right]_{ij} .\end{aligned} \tag{4.5.20}$$

Here and hereafter we define

$$\widetilde{\mathbf{K}_1}(\mathbf{w}) = \mathbf{K}_1(\mathbf{w}) + D,$$

where D is the discrete Upwind operator introduced above. Particularly for our designated chemotaxis operator $\mathbf{K}_1(\cdot)$ we can even simplify the above expression since $\mathbf{K}_1(\cdot)$ is only nonlinear in the second block row $j > \mathrm{N}$, i.e., $\mathbf{K}_1(\mathbf{w}) = \mathbf{K}_1(\mathbf{v})$. Therefore equation (4.5.20) can be rewritten as

$$\begin{aligned}\left[\widetilde{\mathbf{J}_{11}}(\mathbf{w}_m)\right]_{ij} &= \left[\mathbf{M}_L + \theta\,\delta t\left(\mathbf{L} - 0.5\widetilde{\mathbf{K}_1}(\mathbf{w}_j^+) - 0.5\widetilde{\mathbf{K}_1}(\mathbf{w}_j^-)\right)\right]_{ij} \\ &= \left[\mathbf{M}_L + \theta\,\delta t\left(\mathbf{L} - \widetilde{\mathbf{K}_1}(\mathbf{w}_m)\right)\right]_{ij} .\end{aligned} \tag{4.5.21}$$

Contributions to $\widetilde{\mathbf{J}_{12}}(\mathbf{w}_m)$ For the indices $1 \le i \le \mathrm{N}$ and $\mathrm{N} < j \le 2\mathrm{N}$ the only non-zero contribution stems from the first term, i.e.,

$$\begin{aligned}\left[\widetilde{\mathbf{J}_{12}}(\mathbf{w}_m)\right]_{i(j-\mathrm{N})} &= \left[\frac{\widetilde{\mathcal{A}}(\mathbf{w}_j^+) - \widetilde{\mathcal{A}}(\mathbf{w}_j^-)}{2\sigma}\mathbf{w}_m\right]_i \\ &= -\frac{\theta\,\delta t}{2\sigma}\left[\left(\widetilde{\mathbf{K}_1}(\mathbf{w}_j^+) - \widetilde{\mathbf{K}_1}(\mathbf{w}_j^-)\right)\mathbf{u}_m\right]_i .\end{aligned} \tag{4.5.22}$$

We remark that in the absence of Upwinding, i.e., $\widetilde{\mathbf{K}_1}(\cdot) = \mathbf{K}_1(\cdot)$ the above expression simplifies to

$$\begin{aligned}
\left[\widetilde{\mathbf{J}_{12}}(\mathbf{w}_m)\right]_{i(j-\mathrm{N})} &= -\frac{\theta\,\delta t}{2\sigma}\left[\left(\widetilde{\mathbf{K}_1}(\mathbf{w}_j^+) - \widetilde{\mathbf{K}_1}(\mathbf{w}_j^-)\right)\mathbf{u}_m\right]_i \\
&= -\theta\,\delta t\left[\mathbf{K}_1(\mathbf{e}_j)\,\mathbf{u}_m\right]_i \\
&= -\chi\,\theta\,\delta t\left[\mathbf{K}_2(\mathbf{u}_m)\right]_{ij}.
\end{aligned}$$

In other words, the approximation of the high-order Jacobian coincides with the exact Jacobian.

Contributions to $\widetilde{\mathbf{J}_{21}}(\mathbf{w}_m)$ Since the (2,1)-block is linear, there is only the contribution stemming from the system matrix itself, i.e.,

$$\widetilde{\mathbf{J}_{21}}(\mathbf{w}_m) = \widetilde{\mathbf{J}_{21}} = -\theta\delta t\mathbf{M}\,. \tag{4.5.23}$$

Contributions to $\widetilde{\mathbf{J}_{22}}(\mathbf{w}_m)$ As in the previous block we simply obtain

$$\widetilde{\mathbf{J}_{22}}(\mathbf{w}_m) = \widetilde{\mathbf{J}_{22}} = \mathbf{M} + \theta\,\delta t\,(d_v\,\mathbf{L} + \mathbf{M})\,. \tag{4.5.24}$$

At this stage of AFC application we have a corresponding low-order scheme at hand. As indicated in, e.g., Figure 4.3, the only remaining task is to choose the explicit flux correction via (4.5.15).

Applying explicit flux limiting

By virtue of the general discussions in Section 4.5.3 we do not employ implicit AFC schemes for chemotaxis models in this work. We would rather defer this challenge to ongoing future research. Here we state the formula for the explicit AFC scheme as introduced above. Note that therefore the flux correction applies to both nonlinear Richardson schemes in the same manner.

Since we imposed artificial diffusion in terms of equation (4.5.6) only on the first solution block u we also promote the flux limiting to be only applied on this part of the solution. Let us denote the nonlinearly converged low-order (intermediate) solution by $\mathbf{w}_L^{n+1}$ (independently of the choice of the underlying nonlinear solution method). The explicit flux correction of $\mathbf{u}_L^{n+1}$ reads

$$\mathbf{u}^{n+1} = \mathbf{u}_L^{n+1} + \delta t\,\mathbf{M}^{-1}\,\bar{\mathbf{f}}(\mathbf{u}_L^{n+1}, \mathbf{u}^n).$$

The fluxes are defined as in (4.5.15)–(4.5.17).

Practical concerns of applying AFC for Newton's method

From the theoretical point of view the aforementioned AFC-scheme improves the solution in terms of preserving positivity and non-oscillatory profiles, particularly for ill-conditioned chemotaxis systems. However, we like to point out that a naive implementation of this scheme is not competitive with respect to computational expense. In the next paragraph we will present the main reason for this postulate.

Algorithms that embed nested loops can often be optimized or at least significantly accelerated by shifting the main workload to the outermost iteration or, in the best case, outside any loop. In the context of algorithmic schemes for solving PDEs, possible pre-calculations of vectors, right-hand sides and matrices are of particular interest.

Here, since we implemented an explicit AFC-scheme, the main computational costs will be spent in calculating the discrete Upwind operator $\mathbf{D}$. Because of the nonlinearity of the chemotaxis operator $\mathbf{K}_1 = \mathbf{K}_1(\mathbf{w}) = \mathbf{K}_1(\mathbf{v})$ we have to calculate the contribution of $\mathbf{D}$ in every nonlinear iteration. The main drawback is that this Upwind operator cannot be easily split up in order to do some global pre-calculations before entering the nonlinear loop. In fact from the definition of $\mathbf{K}_1$ we know

$$\begin{aligned}
\Big[\mathbf{K}_1(\mathbf{v} \pm \sigma\,\mathbf{e}_j)\Big]_{kl} &= \Big[\int_\Omega \Big(\nabla(\mathbf{v} \pm \sigma\,\mathbf{e}_j)\cdot\nabla\varphi_j\Big)\,\varphi_i\Big]_{kl} \\
&= \Big[\int_\Omega \Big(\nabla\mathbf{v}\cdot\nabla\varphi_j\Big)\,\varphi_i\Big]_{kl} \pm \sigma\Big[\int_\Omega \Big(\nabla\mathbf{e}_j)\cdot\nabla\varphi_j\Big)\,\varphi_i\Big]_{kl} \\
&= \Big[\mathbf{K}_1(\mathbf{v})\Big]_{kl} \pm \sigma\Big[\mathbf{K}(\mathbf{e}_j)\Big]_{kl}. \qquad (4.5.25)
\end{aligned}$$

In general, however, the Upwind operator $\mathbf{D}$ yields

$$\begin{aligned}
\Big[\mathbf{D}\Big]_{kl} = \Big[\mathbf{D}(\mathbf{v} \pm \sigma\,\mathbf{e}_j)\Big]_{kl} &= \max\Big\{-\Big[\mathbf{K}_1(\mathbf{v} \pm \sigma\,\mathbf{e}_j)\Big]_{kl}, 0, -\Big[\mathbf{K}_1(\mathbf{v} \pm \sigma\,\mathbf{e}_j)\Big]_{kl}\Big\} \\
&= \max\Big\{-\Big[\mathbf{K}_1(\mathbf{v}) \pm \sigma\,\mathbf{K}_1(\mathbf{e}_j)\Big]_{kl}, 0, -\Big[\mathbf{K}_1(\mathbf{v}) \pm \sigma\,\mathbf{K}_1(\mathbf{e}_j)\Big]_{kl}\Big\} \\
&\neq \max\Big\{-\Big[\mathbf{K}_1(\mathbf{v})\Big]_{kl}, 0, -\Big[\mathbf{K}_1(\mathbf{v})\Big]_{kl}\Big\} \qquad (4.5.26) \\
&\quad \pm\sigma\,\max\Big\{-\Big[\mathbf{K}_1(\mathbf{e}_j)\Big]_{kl}, 0, -\Big[\mathbf{K}_1(\mathbf{e}_j)\Big]_{kl}\Big\}.
\end{aligned}$$

Precisely speaking we tend to over- or underestimate the contributions of Upwind when applying the expression on the right-hand side of the inequality in (4.5.26) (also depending on the ± sign) in order to shift some workload outside the nonlinear iteration. We see that the approximation of the low-order Jacobian seems to be a very costly task. As we can see from the expressions in (4.5.21)–(4.5.24) the main computational effort will be expensed by the $(1,1)$-block, since only this term crucially depends on the perturbation $\pm\sigma\,\mathbf{e}_j$. In fact, this block has to be build in a loop over all $\mathbf{e}_j$, i.e., in terms of FE language, in a loop over all degrees of freedom of the current FE-space (note that the block solution has overall 2N degrees of freedom). Algorithm 4.5 sketches the computations for assembling the approximated Jacobian in one particular, say m^{th}, nonlinear iteration step, i.e., $\widetilde{\mathfrak{J}}(\mathbf{w}_m)$.

Note that in the light of (4.5.25) we save some computation time by assembling $\mathbf{K}_1(\mathbf{w})$ outside the j-loop, see line 1. Moreover the assembly of $\mathbf{K}_1(\mathbf{e}_j)$ $(j > \mathrm{N})$ in line 7 can be readily improved by comprehensively studying its definition and the underlying FE-space. Let us recall the main assumption of our underlying FE discretization, i.e., our FE trial and test function-space coincide

Algorithm 4.5 Computing the approximated low-order Jacobian $\widetilde{\mathfrak{J}}(\mathbf{w})$ (given $\mathbf{w}$)

```
Require: Let us assume that all linear matrices, i.e., $\mathbf{M}, \mathbf{M}_L$ and $\mathbf{L}$, and all parameters are passed to this routine
Assemble $\mathbf{K}_1(\mathbf{w})$
Calculate $\widetilde{\mathbf{K}_1}(\mathbf{w}) = \mathbf{K}_1(\mathbf{w}) + \mathbf{D}(\mathbf{w})$
Build jacblock(1,1): $\left[\widetilde{\mathfrak{J}}(\mathbf{w})\right]_{i\leq \mathrm{N}}^{j\leq \mathrm{N}} = \left[\mathbf{M}_L + \theta\,\delta t\left(\mathbf{L} - \widetilde{\mathbf{K}_1}(\mathbf{w})\right)\right]_{ij}$
Build jacblock(2,1): $\left[\widetilde{\mathfrak{J}}(\mathbf{w})\right]_{i>\mathrm{N}}^{j\leq \mathrm{N}} = -\theta\,\delta t\,\mathbf{M}_{ij}$
Build jacblock(2,2): $\left[\widetilde{\mathfrak{J}}(\mathbf{w})\right]_{i>\mathrm{N}}^{j>\mathrm{N}} = \left[\mathbf{M} + \theta\,\delta t\,(d_v\mathbf{L} + \mathbf{M})\right]_{ij}$
for $j = \mathrm{N}+1, 2\mathrm{N}$ do
    Assemble $\mathbf{K}_1(\mathbf{e}_j)$
    Calculate $\widetilde{\mathbf{K}_1}(\mathbf{w}_j^{\pm}) = \mathbf{K}_1(\mathbf{w}) \pm \mathbf{K}_1(\mathbf{e}_j) + \mathbf{D}(\mathbf{w}_j^{\pm})$
    Build jacblock(1,2): $\left[\widetilde{\mathfrak{J}}(\mathbf{w})\right]_{i\leq \mathrm{N}}^{j} = \frac{\theta\,\delta t}{2\sigma}\left[\left(\widetilde{\mathbf{K}_1}(\mathbf{w}_j^{+}) - \widetilde{\mathbf{K}_1}(\mathbf{w}_j^{-})\right)\mathbf{u}\right]_{ij}$
end for
```

and are defined by $\mathcal{Q}_1$, i.e., bilinear-quadrilateral elements whose degrees of freedom are the function values at the corner vertices (cf. Chapter 2).

As we recall from (4.2.5) we have

$$\left[\mathbf{K}_1(\mathbf{e}_k)\right]_{ij} \quad = \quad \int_{\Omega} \chi\,(\nabla\varphi_k \cdot \nabla\varphi_i)\,\varphi_j\,\mathrm{d}\mathbf{x}.$$

From the assumptions about the FE-space above, we remark that the integral vanishes for most of the i-j-combinations, simply because for a given index k the support of the integrand is already covered by at most four neighboring elements of node k. Figure 4.4 illustrates these non-zero contributions on a simple mesh. In other words at most only nine degrees of freedom contribute

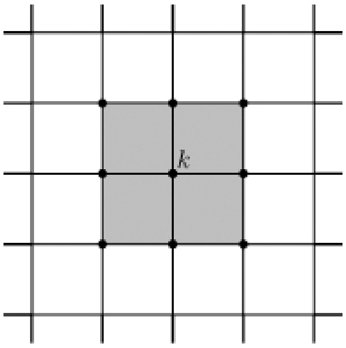


Figure 4.4: Exemplary grid section depicting the non-zero contributions to matrix $\mathbf{K}_1(\mathbf{e}_k)$ for a given node k. Only the gray-shaded elements and their corresponding nodes, marked by a dot, contribute non-zero entries.

to $\mathbf{K}_1(\mathbf{e}_k)$. Let $\mathcal{N}_k$ denote these degrees of freedom, then we have

$$\Big[\mathbf{K}_1(\mathbf{e}_k)\Big]_{ij} \;=\; 0\,, \quad \text{for } i,j \notin \mathcal{N}_k\,. \tag{4.5.27}$$

This consideration not only saves enormous memory (precisely speaking there are only 81 non-zero entries), but also allows for a fast matrix assembly. We therefore substitute $\mathbf{K}_1(\mathbf{e}_j)$ by $\mathbf{K}_1^{\mathcal{N}_j}(\mathbf{e}_j)$ in line 7, wherein $\mathcal{N}_j$ denotes the restriction of the matrix to the contributions of the 81 entries.

In addition, if we use a uniform and fixed grid, we can also make use of template matrices. For example let k_* be a particular inner node and let $\mathbf{K}_1(\mathbf{e}_{k_*})$ be assembled and stored globally. Then the assembly of subsequent matrices $\mathbf{K}_1(\mathbf{e}_k)$, where k refers to an inner node, can be enhanced by copying corresponding entries of the template matrix $\mathbf{K}_1(\mathbf{e}_{k_*})$. In other words, the matrix entries of $\mathbf{K}_1(\mathbf{e}_k)$, for k referring to an inner node, do not require explicit calculations anymore.

Remark 4.5 *Firstly, concerning the stabilization of the chemotaxis term in the low-order formulation (4.5.18) let us remark that in the case of the general chemotaxis model (4.1.1) the formulation yet ignores possible negative contributions from the discrete proliferation term* $\mathbf{G}(\cdot)$ *and the chemical production term* $\mathbf{S}(\cdot)$*. This is intended by the author, since source and sink terms usually require more special care due to their physical nature, e.g., missing mass conservation. The treatment of sources and sinks in the context of AFC stabilization methods is still a vital topic of current research and hence cannot be satisfactorily discussed in this present work.*

Secondly, we remind ourselves that in our formulation (4.5.18) we do not modify the second row-block. However, for consistency reasons we recommend to use mass lumping for the corresponding terms, which, in turn, entails also an explicit flux correction for the second solution block. In fact, another selling point for employing mass lumping is the simplification of the resulting system. Note that as the time stepping keep decreasing, the crucial task is to invert the mass matrices, and if those are diagonalized, the challenge of inverting can indeed be easily accomplished. Let us state the modifications for the mass lumping of the second row-block for the minimal model of chemotaxis. After substituting all consistent mass matrices in this row-block with the lumped counterparts $\mathbf{M}_L$*, the correcting flux reads*

$$\mathbf{f}^v \;=\; (\mathbf{M}_L - \mathbf{M})\frac{\mathrm{d}\mathbf{v}}{\mathrm{d}t} - (\mathbf{M}_L - \mathbf{M})\mathbf{v} + (\mathbf{M}_L - \mathbf{M})\mathbf{u}.$$

Similar to the flux in (4.5.8) the residual error can be casted into the following representation

$$f_i^v \;=\; \sum_{j\neq i} f_{ij}^v \qquad \textit{with } f_{ij}^v = -f_{ji}^v,$$

where the raw antidiffusive fluxes from node j *to node* i *are denoted by* f_{ij}^v *and can be written as*

$$f_{ij}^v \;=\; m_{ij}\Big[\frac{\mathrm{d}}{\mathrm{d}t}(v_i - v_j) - (v_i - v_j) + (u_i - u_j)\Big].$$

The fully discretized AFC scheme can now be easily derived in a similar fashion as for the first block row.

5 Numerical analysis

This main chapter deals with the numerical analysis of the FEM iteration strategies which were developed in the preceding chapter, Chapter 4. Prior to the investigation of these iteration schemes, we will briefly shed light on the multigrid solver which is one of the improvements of already existing schemes in preceding papers of the author, [96–98]. As the main objective of this chapter, these bundles of methodologies will be applied on particular nonlinear models for chemotaxis, whereby we mainly focus on the influence of the chemosensitivity on the stability of the solution process, i.e., convergence of the solver.

Before we begin our numerical study we will provide the reader with a short state of the art of numerical investigations which can be found in present literature, Section 5.1. Afterwards we will present the numerical analysis of the solver for the underlying linear subsystems, Section 5.3, a numerical comparison of all iteration schemes, Section 5.4, and numerical results that reveal certain limitations of the iteration schemes, Section 5.5. In all of these investigations we will not consider any stabilization techniques. This will be solely the subject of Section 5.6. At the end of this chapter, we will provide a summary of the main numerical findings, Section 5.7.

5.1. The numerical state of the art

Even though this work offers an unprecedented in-depth study of solver methodologies for chemotaxis related systems of nonlinear equations, (particularly very recently) there have been some notable numerical assets. Basically all well-established spatial discretization frameworks have been considered for chemotaxis related PDEs: FV, FD, DG and finally FEM. This brief survey gives references for all of the above frameworks.

Tyson *et al.* [102] introduced a FV based fractional step method of so called *Strang splitting-type*, which is at most of second order. However this operator splitting approach is only largely applicable for operators of ‘pure’ character, i.e., problems arise when the advective part (here chemotaxis) is not of pure hyperbolic character anymore. For the governing model in [102] (pattern formation in bacterial growth) the assumption of a hyperbolic advection is reasonable, but for the minimal model of chemotaxis this fails in generic situations as, e.g., Chertock and Kurganov pointed out in [16]. In fact, Tyson *et al.* figured out this problem in numerical test cases when they stated that

> "*[...] the advection-diffusion and reaction-diffusion pairings are stable while the advection-reaction pairing is highly unstable. [...] We investigated changing the methods used for reaction and advection steps, but these did not eliminate the problem.*"

In this context, Chertock and Kurganov [16] developed a positivity preserving central upwind scheme based on FV. This scheme has been successfully applied on the minimal model, a kinetic model and a haptotaxis model. Very recently, an ongoing research of Bencheva [7] strive to apply this scheme also to a model of stem cell migration. These schemes, however, give only rise to first order accuracy.

In the context of a hyperbolic-parabolic chemotaxis model a FD based operator splitting has also been addressed by Gerisch *et al.* [34]. The limitations of the operator splitting as mentioned earlier remain also for this work. Another FD based scheme was promoted by Wise *et al.* in [108]. Therein they applied an adaptive multigrid algorithm on a complex tumor growth model with an overall of five variables, which obviously reflects one motivation of keeping the total number of degrees of freedom as small as possible, e.g., by spatial adaptation strategies. Unfortunately divergence of the embedded multigrid solver in case of a fully implicit treatment primary causes a first order accuracy of the overall algorithm.

The recent gain in popularity of DG also hit the chemotaxis community. Epshteyn and Kurganov [29] adapted the FV upwind scheme of [16] to fit in the context of DG. Their main idea was it to rewrite the original model, i.e., the minimal model of chemotaxis, into a form which provides a pure hyperbolic advection term. This extends the model to a system of four variables. One of the main shortcomings of FV, FD and DG approaches is the practical restriction of the algorithms to be only capable of mostly 'academic' computational domains. Caused by the nature of these approaches highly tedious mesh-dependent calculations are required to capture 'realistic' domains. A soon appearing paper of Epshteyn [28] counters this issue. In this paper she proposes a novel upwind-difference potentials method that was originally developed for composite domain problems.

Among the FEM approaches in the literature let us comment on the ones obtained by Kirk and Carey [55], Marrocco [70], Saito [90] and Strehl *et al.* [96]. Kirk and Carey [55] considered a parallel, adaptive FEM scheme, which was mainly motivated by a pattern forming model of chemotaxis. The spatial adaption was obtained via a gradient-jump error indicator and subdomain partitioning parallel solution strategy. The additional time step adaption was ensured by simple truncation error estimates. The authors were surprised that their contributions provided one of the first adaptive approaches for chemotaxis models by remarking

> "*It is interesting to note that while such features [rapid transients and highly localized spatial features, RS] make this class of problem particularly well-suited to simulation techniques employing local adaptive mesh refinement and coarsening, there has been little adaptive work for these chemotactic biological systems to date.*"

Marrocco followed a mixed FEM approach originating from the numerical treatment of semiconductor modeling, [70]. Notably, his proposed scheme is only first order accurate, since he employed the first-order Raviart-Thomas element $\mathcal{R}_0$. Furthermore, most probably driven by the semiconductor origin, Marrocco studied the parabolic-elliptic version of the minimal model, i.e., dropping the time derivative in the v equation. However, for reasons of stability the author used fully implicit artificial transients to solve the semi-discretized system in time.

> "*As the time step (physical time step) becomes smaller and smaller the convergence of the iterates becomes more and more difficult to obtain and solution of quasi-static problems is not reached. The whole algorithm failed to converge.*"

Saito also studied the simplified parabolic-elliptic version of the minimal model and derived a conservative upwind FEM scheme, [90]. The drawback of his upwinding method is its first order accuracy. The discretization in time was countered with the first order backward Euler difference quotient and the resulting nonlinear system was finally linearized by the standard first order fixed-point method. Certainly, the limit of being a first order discretization in space and time (and in the nonlinearity) is a major drawback, however, with his scheme Saito developed a first stabilized FEM scheme for chemotaxis models which preserves positivity (under reasonable restrictions on the spatial mesh and the time steps) and conserves the initial mass. The investigation of a high-order, i.e.,
$1+\alpha$-order, FEM discretization scheme for general chemotaxis models was the subject of the work of Strehl *et al.* in [96]. Therein, in contrast to Saito, the stabilized spatial discretization was derived via a discrete upwinding scheme that was corrected by algebraic fluxes, cf. Section 4.5. The advantage of this approach is the modification of the underlying system on a pure algebraic level, i.e., there is no need for an explicit approximation of fluxes on the continuous level. Moreover all modifications can be composed in a conservative manner. At the same time all the flexibility of FEM discretizations are maintained.

It is remarkable that all (except [108]) of the aforementioned numerical results only considered schemes for simulating 2D chemotaxis models. To the best of our current knowledge the development and analysis of stable, accurate, efficient and flexible numerical tools for 3D simulations of chemotaxis models are largely omitted by the numerical community. Taking into account the lack of theoretical background for three dimensions, this reveals a remarkable gap. So far, there are only very few research groups that numerically treat 3D chemotaxis models, [98]. Moreover, despite the aforementioned numerical contributions, a comparison of different solver techniques is still missing. Therefore, in Strehl et al. [97] the authors provided a first attempt to quantitatively compare different solver strategies for selected chemotaxis models.

To put it into a nutshell, up to the best knowledge of the present author, the state of the art of the numerical investigation of general chemotaxis models only provides ‘partial solutions’. In the best believe of the present author a FEM approach for tackling complex chemotaxis models admits several advantages over the other aforementioned methods. Amongst others, the most striking features are as follows:

1. **Complex geometries:** Because of its nodal basis functions, FEM can be readily adopted to irregular spatial meshes and complex geometries, which makes it particular favorable from the application point of view.

2. **r-/h-/p-adaptivity:** Pursuing the preceding point, the flexibility of FEM are the major asset when applied to highly dynamic systems, e.g., local transient solutions or deforming geometries. Relocation of mesh nodes, local refinements and augmentation of polynomial degree lead to fast convergence even in the case of strongly perturbed meshes or locally transient behavior.

Therefore the upcoming numerical sections of the current work pursues the research path of Strehl *et al.* by that it thoroughly analyzes specialized FEM discretizations of high order and suitable solver for generic chemotaxis models.

5.2. Numerical preliminaries

After we presented the current state of the art of the numerical treatment of chemotaxis related PDEs, we will introduce some preliminaries which facilitate the guidance through the upcoming numerical analysis. Besides a brief sketch over the underlying hardware of the present numerical investigations, we will also provide a conceptual survey of the software library on which our implementation is based on. Moreover, general issues concerning the algorithmic design and setup will be tackled in a following section prior to the numerical analysis.

5.2.1. On the software

The development and implementation of the numerical algorithms presented in this work has been accomplished in FORTRAN90 and is embedded into the open-source software library of FEAT[1] (**F**inite **E**lement **A**nalysis **T**ools). This software is maintained by the chair of Applied Mathematics and Numerics at the TU Dortmund and benefits from contributions of a huge variety of fields of finite element applications. The original focus of FEAT was to solve complex problems arising from fluid dynamics with an industrial background. The modelling was mainly based on variants of the incompressible Navier Stokes equations with particular interest in nonlinear viscosity, fluid-structure interaction, multiphase flow with chemical reactions, free boundary value problems with solidification, just to mention a few. However, currently FEAT experiences new algorithmic contributions coming from various applied fields of research such as hardware-oriented and parallel computing, Lattice Boltzmann methods, level-set approaches, fictitious boundary methods, collision detection models. Also from the application point of view the software library comprises increasingly more PDE problems arising from (magneto-)hydrodynamic or non-Newtonian flows, optimization of physical quantities, fluid-structure interactions, population balance equations, solid mechanics, drug delivery and, beginning with this recent work, also chemotaxis related developments.

The graphical illustrations of the resulting solutions have been rendered with GMV[2] (**G**eneral **M**esh **V**iewer) or the open-source, multi-platform data analysis and visualization application PARAVIEW[3]. The convergence plots and the representative plots of cutlines have been visualized with MATLAB[4].

5.2.2. On the general numerical setup

This section is devoted to the general notations and setup of the algorithms that will be in the spotlight in the subsequent numerical analysis. For transparency purposes Table 5.1 depicts some basic abbreviations used in the upcoming numerical data.

[1] Online presence: http://www.featflow.de/en/index.html

[2] Online presence: http://www.generalmeshviewer.com/

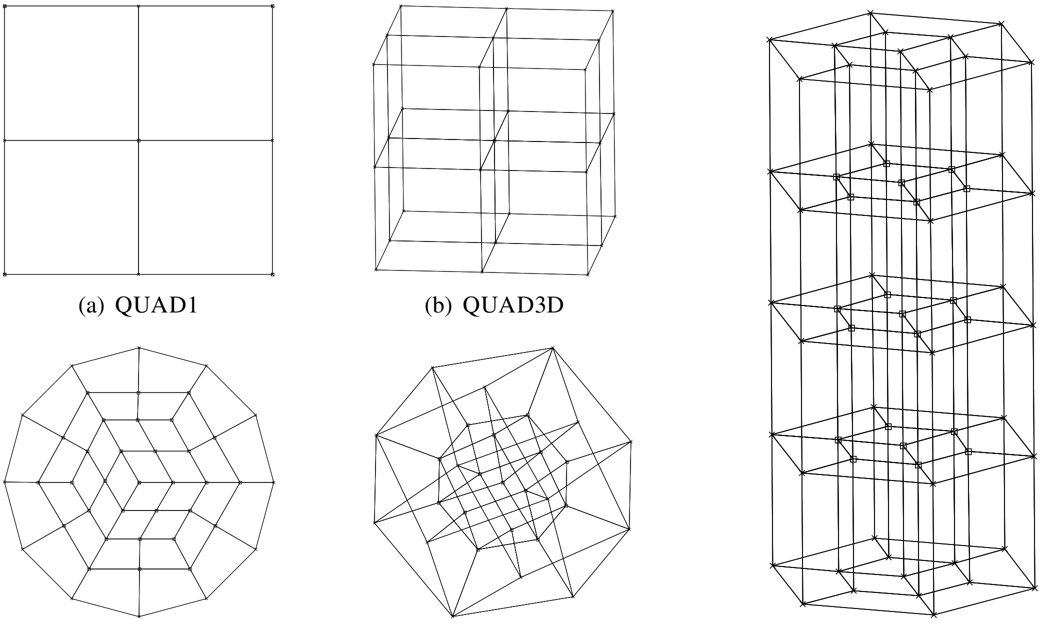
[3] Online presence: http://www.paraview.org/

[4] Online presence: http://www.mathworks.de/products/matlab/

LEV	underlying spatial level refinement (coarsest level = 0)
IT_LIN	number of linear iterations
ERR	error estimation (concerning a reference solution)
COSTS	computational costs in terms of required number of iterations
EFF	approximation of the efficiency by means of $\text{EFF} = \text{ERR}^{-1} \cdot \text{COSTS}^{-1}$
$\text{ALG}_\text{A}(\text{ALG}_\text{B})$	cascaded solver algorithm, ALG_A is preconditioned with ALG_B

Table 5.1: Overview of basic abbreviations used in the subsequent analysis.

In order to provide an overview of the resulting degrees of freedom of the underlying spatial mesh refinements, Table 5.2 lists some commonly used discretizations. Table 5.2 contains the resulting degrees of freedom for $\mathcal{Q}_1$ finite elements and the range of the mesh size δh for the main computational domains used in this present numerical investigation. Moreover in Figure 5.1 we


(a) QUAD1

(b) QUAD3D

(c) CIRC1

(d) CIRC3D

(e) CYL3D

Figure 5.1: Visualization of the different computational domains at exemplary refinement levels.

When dealing with numerical results we have to rely on the convergence of the underlying (linear/nonlinear) iteration processes. To this end, let us consider an appropriate choice of a termination criterion for the emerging iterations. As we see from Chapter 4 we have at most two iteration procedures: (i) an outer nonlinear loop obtained by the nonlinear Richardson scheme and (ii) an inner linear iteration arising from the linear subproblem of the Richardson corrections/updates – note that we do not consider exact solver for the linear subsystems in this work, since the applicability of our schemes should be retained for complex systems with a huge number of degrees of freedom which renders direct solver unpractical.

Now, in terms of a reliable nesting strategy for the outer and inner iterations, the prescribed

Ω_h	underlying domain Ω	LEV	δh	#FE	N
QUAD1	$\{\mathbf{x} \in \mathbb{R}^2 \mid$	0	1	1	4
	$0 < x_1, x_2 < 1\}$	7	$1/128$	16384	16641
		8	$1/256$	65536	66049
QUAD16	$\{\mathbf{x} \in \mathbb{R}^2 \mid$	0	16	1	4
	$0 < x_1, x_2 < 16\}$	7	$1/8$	16384	16641
		8	$1/16$	65536	66049
QUAD3D	$\{\mathbf{x} \in \mathbb{R}^3 \mid$	0	16	1	8
	$0 < x_1, x_2, x_3 < 16\}$	6	$1/4$	262144	274625
		7	$1/8$	2097152	2146689
CIRC1	$\{\mathbf{x} \in \mathbb{R}^2 \mid$	0	$\approx [0.5, 1]$	9	13
	$x_1^2 + x_2^2 < 1\}$	6	$\approx [1/128, 1/64]$	36864	37057
		7	$\approx [1/256, 1/128]$	147456	147841
		8	$\approx [1/512, 1/256]$	589824	590593
CIRC16	$\{\mathbf{x} \in \mathbb{R}^2 \mid$	0	$\approx [4, 8]$	9	13
	$x_1^2 + x_2^2 < 8^2\}$	5	$\approx [1/8, 1/4]$	9216	9313
CIRC3D	$\{\mathbf{x} \in \mathbb{R}^3 \mid$	0	$\approx [3.7, 7.4]$	16	29
	$x_1^2 + x_2^2 + x_3^2 < 8^2\}$	6	$\approx [0.06, 0.12]$	4194304	4219265
CYL3D	$\{\mathbf{x} \in \mathbb{R}^3 \mid$	0	$\approx [0.5, 1]$	36	65
	$0 < x_3 < 4, x_1^2 + x_2^2 < 1\}$	4	$\approx [1/32, 1/16]$	147456	152945

Table 5.2: Listing of the most commonly used spatial levels and resulting degrees of freedom of the different computational domains.

termination criteria must be matched. That is, the inner solver precision should be related to the current outer solver precision to counter an excessive overhead of accuracy of the inner solver. Moreover, not only the precision, but also the measurement of precision should be concerned. By means of measurement we address the two main strategies, reducing the actual error in subsequent solutions, i.e., $||x_m - x^*|| \longrightarrow$ min!, or reducing the residual error, i.e., $||b - Ax_m|| \longrightarrow$ min! for some linear system $Ax = b$. Herein x_m and x^* denote the m^{th} nonlinear iteration of the numerical solution and the exact solution, respectively. In the framework of Newton methods, the former and latter are sometimes also referred to as *error-based* and *residual-based* concepts, respectively, cf. Deuflhard in [22].

Termination criterion for a nested Richardson iteration

Recapture that we consider two schemes within the Richardson framework, a Picard and a inexact Newton(-like) linearization. Efficient cascaded iteration procedures link the termination criteria for the nonlinear (outer) and linear (inner) iteration in such a way that the outer convergence rate is retained at 'minimal' expense for the inner iteration. A constant accuracy for the inner iteration, e.g., prescribing a gain of four digits, obviously does not render this concern. Exemplary, let us confine ourselves here to the inexact Newton linearization, as this method is well popular for having a rather fuzzy convergence rate.

For the readers convenience we therefore repeat the Newton formulation. We are looking for solution updates via

$$\begin{cases} \operatorname{jac}\big(\mathcal{A}(\mathbf{w}_m)\big)\,\mathbf{y}^* &=& res(\mathbf{w}_m)\,, \\ \mathbf{w}_{m+1} &=& \mathbf{w}_m + \mathbf{y}^*\,. \end{cases} \tag{5.2.1}$$

Certainly, if we employ an inexact solver for this system, we only end up with an approximative update $\mathbf{y} \approx \mathbf{y}^*$ introducing some error ERR^{lin}. Let us now suppose that the outer Newton iteration m already reached the area of quadratic convergence, i.e., $\text{ERR}_m = \mathcal{O}\big((\text{ERR}_{m-1})^2\big)$. For the inexact Newton method this outer error apparently depends upon the inner approximation, i.e., $\text{ERR}_m = \text{ERR}_m(\text{ERR}^{lin})$. The question that naturally arises is how ERR^{lin} has to be restricted in order to still retain the quadratic convergence per outer iteration. As the linear solver is converging, it is clear that we could prescribe a precision at will. However, this strategy turns out to be very costly, because the outer Newton iteration limits the accuracy gain (in terms of quadratic convergence). In this context, Dembo *et al.* [21] suggested a choice of kind

$$\text{ERR}^{lin} = \min\Big\{c\,\text{ERR}_m, 0.5\Big\}\,,$$

where $c \leq 1$ is some constant. In other words, the inner termination criterion is proportional to the outer one. Note that the additional second argument of min accounts to the well known fact of poor initial convergence of Newton's method. More elaborated choices and corresponding proofs of convergence can be found in [26] and [22]. In our numerical studies we adapt the above choice by taking $c = 1$ and a minimal residual drop of 0.01 instead of 0.5 (as above).
As the interested reader might have remarked, we did not clarify the measurement for ERR_m and ERR^{lin} yet. This will be illuminated in the following.

On the error norms

We notice that Dembo *et al.* as well as Eisenstat *et al.* studied the convergence of inexact Newton methods in terms of consistent residual drops. Consistency accounts to the fact that the authors restricted the residual of the inner Newton system (we refer to this as *linear* or *inner residual*) in order to control the residual of the outer iteration (we refer to this as the *nonlinear* or *outer residual*).

$$\begin{aligned} \text{outer residual} &: \; res(\mathbf{w}_m) \\ \text{inner residual} &: \; res(\mathbf{w}_m) - \operatorname{jac}\big(\mathcal{A}(\mathbf{w}_m)\big)\,\mathbf{y} \end{aligned}$$

In terms of the above, the suggested constraint of Dembo *et al.* reads

$$\frac{||res(\mathbf{w}_m) - \operatorname{jac}\big(\mathcal{A}(\mathbf{w}_m)\big)\,\mathbf{y}||}{||res(\mathbf{w}_m)||} = \min\Big\{c\,||res(\mathbf{w}_m)||/||res(\mathbf{w}_0)||, 0.5\Big\}\,, \tag{5.2.2}$$

viz., the relative inner residual (the initial guess of the inner system is assumed to be zero) is controlled by the outer residual of the current (nonlinear) iteration. In fact, it is very reasonable that this consistency must be obeyed. Dembo *et al.* commented on the error convergence $||\mathbf{y} - \mathbf{y}^*|| \to 0$ as follows, [21, p.407]:

> "*The problem is that a step which makes the error very small need not result in a correspondingly small relative residual.*"

For our purposes their statement can be interpreted as: the control of the inner residual does not necessarily give a control of the actual Newton iterates $\mathbf{w}_m$, but of the corresponding outer residuals $res(\mathbf{w}_m)$.

This issue of consistency has been vastly examined by Deuflhard [22]. For our concerns two main classes of so-called *adaptive inexact Newton methods* are of importance, the residual-based and the error-based approach. The former measures the norms of the nonlinear and linear residual, whereas the latter monitors approximations of the norm of the solution error $||\mathbf{w}_m - \mathbf{w}^*||$ and $||\mathbf{y} - \mathbf{y}^*||$, respectively for the nonlinear solution and the inner Newton update. A particular choice of one of those approaches consistently entails the recommendation of a specific linear solver for the inner Newton system. In particular the standard GMRES is predestined to work within a residual-based strategy, whereas CGNE (CG *normal equations error-minimizing*) works consistently within a error-based strategy. Since error-based minimization techniques are rather uncommon, we refer the interested reader to Weiss [104] for the recent development herein. For well conditioned systemmatrices this classification might seem redundant, however in the case of ill conditioned systems (or more precisely, ill conditioned Jacobians) a proper choice is unavoidable, since the control of the residual does not necessarily imply a sufficient control over the final solution itself in these cases.

What we did not concern about yet is the well known phenomenon of 'oversolving' (of either outer and inner iteration) if the initial residual is already close to zero. In this case a relative tolerance, e.g., of kind (5.2.2), will result in excessively many iterations and hence we additionally introduce an absolute stopping criteria. Kelley [53] proposed the balancing

$$||res(\mathbf{w}_m)|| \overset{!}{\leq} \varepsilon_r \, ||res(\mathbf{w}_0)|| + \varepsilon_a,$$

where ε_r and ε_a are certain real values for the relative and absolute error tolerance, respectively. In our implementations we terminated the iteration if either the relative or the absolute threshold is reached, which is a sufficient condition for the above inequality to hold. Typically the absolute threshold is chosen close to a common machine precision, i.e., $\varepsilon_a \approx 1E\text{-}14$, and the relative tolerances can be different for the outer nonlinear (ε_r^{nl}) and inner linear (ε_r^{lin}) iteration.

In all of our numerical algorithms we implemented a residual-based termination criterion since it matches very well with the underlying linear solver. For clarity reasons, let us close our considerations with listing our implemented residual-based termination criteria for the outer and inner iteration.

$$\begin{cases} \text{outer} \; : \; \dfrac{||res(\mathbf{w}_m)||}{||res(\mathbf{w}_0)||} \leq \varepsilon_r^{nl} \quad \text{or} \quad ||res(\mathbf{w}_m)|| \leq \varepsilon_a \\[2em] \text{inner} \; : \; \dfrac{||res(\mathbf{w}_m) - \text{jac}\big(\mathcal{A}(\mathbf{w}_m)\big)\,\mathbf{y}||}{||res(\mathbf{w}_m)||} \leq \varepsilon_r^{lin} = \min\left\{ \left(\dfrac{||res(\mathbf{w}_m)||}{||res(\mathbf{w}_0)||}\right)^2, 0.01 \right\} \quad \text{or} \\[1em] \qquad\qquad ||res(\mathbf{w}_m) - \text{jac}\big(\mathcal{A}(\mathbf{w}_m)\big)\,\mathbf{y}|| \leq \varepsilon_a \end{cases} \tag{5.2.3}$$

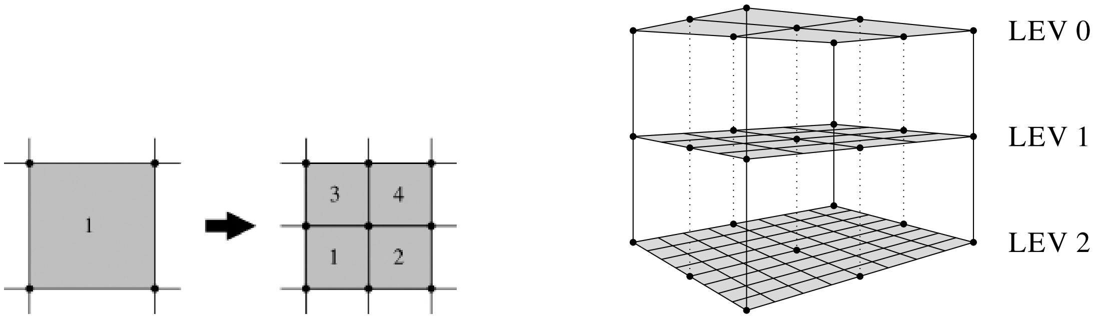
If not explicitly stated else, we choose $\varepsilon_a = 1E\text{-}14$ and $\varepsilon_r^{nl} = 1E\text{-}6$.

5.3. Analysis of the linear sub-solver

This section is devoted to the analysis of the linear (sub-) solver for the monolithic schemes LIN, PIC and NEWT, i.e., the linear solver for LIN and the inner iteration cycle within the nonlinear Richardson scheme for PIC and NEWT. Our main objective is to promote a powerful multigrid solver that delivers a mesh-independent convergence and hence a reliable feature to tackle chemotaxis related discretizations on a very accurate spatial mesh. The preliminary work of Wise *et al.* [108] already stressed the potential of applying multigrid algorithms to chemotaxis related PDEs. In the current work we focus on a basic analysis of a proper multigrid setup which includes the choice of the level-to-level smoother.

5.3.1. The multigrid algorithm

In the following we will sketch the multigrid algorithm for conforming quadrilateral finite elements with canonical level refinement leading to an approximate halfening of the mesh size, cf. Figure 5.2. In Table 5.3 we introduce some notations which will ease the understanding of the multigrid algorithm.



(a) Canonically element refinement

(b) Resulting mesh hierarchy

Figure 5.2: Visualization of the canonical 2D mesh refinement for quadrilateral finite elements. Every edge-midpoint and every face midpoint creates one new node. Hence a refinement of one quadrilateral finite element results in four quadrilateral finite elements. This refinement is applied for all Elements, particularly no hanging nodes appear. The right-most figure shows a canonically resulting three level hierarchy.

In contrast to usual single grid solver routines, multigrid operates on a given hierarchy of successive discrete subspaces, say $\{V_h\}_{h_{\min}}^{h_{\max}}$. Here, we will only focus on a pure spatial multigrid, i.e., only different spatial meshes are considered, in contrast to the full time-space cylinder. Therefore, we can also simply identify the discrete space hierarchy with the spatial mesh size, say $\{h\}_{h_{\min}}^{h_{\max}}$. The underlying discretized problem is originally only defined on the most finest grid, i.e., $\mathcal{A}_{h_{\min}}\mathbf{w}_{h_{\min}} = \mathbf{b}_{h_{\min}}$. However, the treatment of this problem is now passed through the level hierarchy until the coarse level is reached. During this transfer, auxiliary problems of the form $\mathcal{A}_h\mathbf{w}_h = \mathbf{b}_h$ are considered. But only on the coarse level a problem of the form $\mathcal{A}_{h_{\max}}\mathbf{w}_{h_{\max}} = \mathbf{b}_{h_{\max}}$ is actually solved by an underlying coarse grid solver ($\mathcal{CG}$). This is usually accomplished by a direct solver (e.g., provided by UMFPACK).

Let us introduce the multigrid algorithm in more detail for a two level hierarchy, cf. [12, Chapter V]. Given the problem on a fine level $\mathcal{A}_h\mathbf{w}_h = \mathbf{b}_h$, the multigrid algorithm consists of several smoothing steps, say s times, and a coarse grid correction. The smoothing steps are required to

$h = h_{\min}, \ldots, h_{\max}$	mesh size (as an index it also determines the corresponding mesh)
$V_{h_{\max}}, \ldots, V_{h_{\min}}$ and V	discrete subspaces related to the corresponding meshes and the continuous (spatial) space stemming from the governing PDE
$\mathbf{w}_h^i$	an i^{th} iterative solution on the level corresponding to h, particularly $\mathbf{w}_h^0$ corresponds to the initial guess
$\mathcal{A}_h$ and $\mathbf{b}_h$	system matrix and right-hand side of the problem on the level corresponding to h
$\mathcal{R}^h_{2h}$ and $\mathcal{R}^{2h}_h$	prolongation and restriction operators which will be used as transfer operators between successive mesh levels
$\mathcal{S}_h$	smoothing operator on the level corresponding to h. Remark that we simplify the notations by assuming both pre- and postsmoothers are alike (we also employ the number of pre- and postsmoothing steps being equal)

Table 5.3: Overview of most multigrid notations to which we refer to in the subsequent analysis.

damp high frequencies of the error. Usually we employ smoothing steps equally before and after the coarse grid correction. Let us denote the application of a smoothing step by the operator $\mathcal{S}_h$. The coarse grid correction involves the level transfer from the fine level to the coarse level and vice versa. This is accomplished by restriction and prolongation operators, denoted by $\mathcal{R}^{2h}_h$ and $\mathcal{R}^h_{2h}$, respectively. With j denoting an iteration index, we can formulate the coarse grid correction as follows:

$$\mathbf{w}_h^{j+1} \quad = \quad \mathbf{w}_h^j + \mathcal{R}^h_{2h} \mathcal{A}_{2h}{}^{-1} \mathcal{R}^{2h}_h (\mathbf{b}_h - \mathcal{A}_h \mathbf{w}_h^j)\,,$$

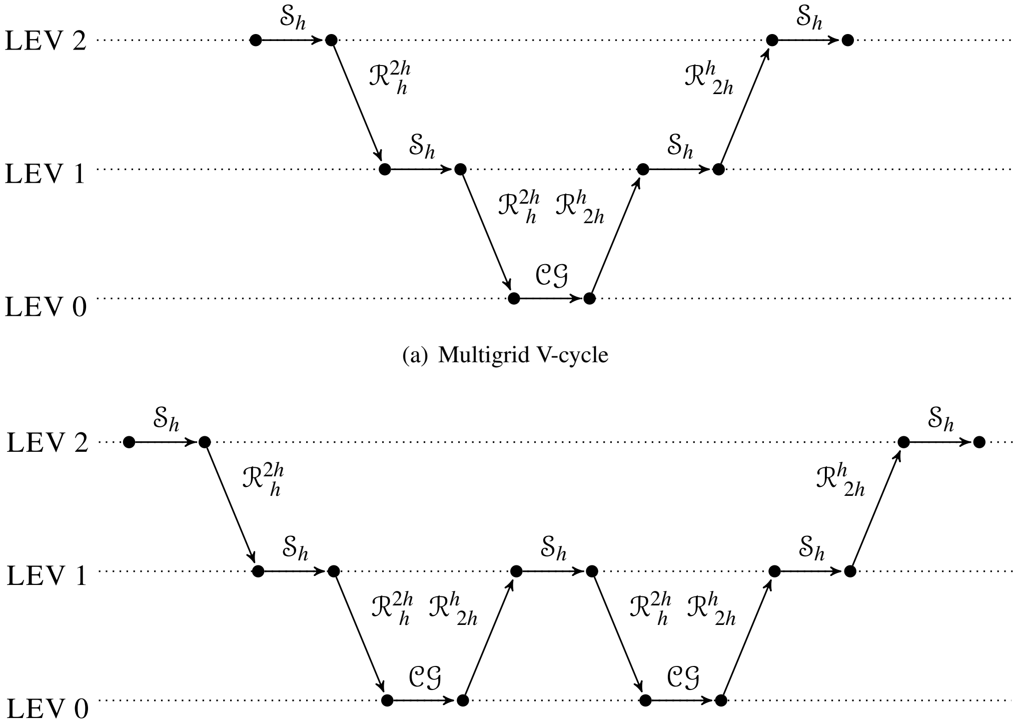
where $\mathcal{A}_{2h}$ is the conforming restriction of $\mathcal{A}_h$ on the coarse level which is assembled prior to the call of the multigrid algorithm. This formulation gives rise to an auxiliary problem on the coarse grid to be solved, and hence, we can rewrite the coarse grid correction by means of

$$\begin{aligned} \mathcal{A}_{2h} \mathbf{w}_{2h} &= \mathbf{b}_{2h}\,, \\ \mathbf{w}_h^{j+1} &= \mathbf{w}_h^j + \mathcal{R}^h_{2h} \mathbf{w}_{2h}\,, \end{aligned}$$

where $\mathbf{b}_{2h} = \mathcal{R}^{2h}_h (\mathbf{b}_h - \mathcal{A}_h \mathbf{w}_h^j)$ is the restriction of the defect. For a two level hierarchy, the auxiliary problem on the coarse grid is usually solved exactly by a direct solver, as already mentioned above. However, for a multi level hierarchy, this problem is treated by a recursive call of the multigrid algorithm.

For a level hierarchy that consists of at least three levels, multigrid algorithms can be altered by the number of (recursive) coarse grid corrections. If the coarse grid correction is only applied once, this leads to a so-called V-cycle. If the coarse grid correction is executed twice, then a so-called W-cycle is performed. The corresponding nomenclature can easily be explained by depicting the level transfers, cf. Figure 5.3.

In Algorithm 5.1 we sketch the core of multigrid. This algorithm represents one single V-cycled multigrid sweep. For a full iterative solver routine we should embed this single sweep into



(a) Multigrid V-cycle

(b) Multigrid W-cycle

Figure 5.3: Visualization of the V- and W-cycle. Herein $\mathcal{S}_h, \mathcal{R}_h^{2h}, \mathcal{R}^h_{2h}$ and $\mathcal{CG}$ denote the corresponding smoother, restriction, prolongation and (direct) coarse grid solver, respectively.

an outer loop which checks for convergence. For the alternative W-cycled sweep we must modify line 7 by employing the coarse grid correction twice in a row, note that for only two levels the V- and W-cycle coincide.

Algorithm 5.1 Single Multigrid sweep

```
 1: function MGS($\mathbf{w}_h^0, \mathcal{A}_h, \mathbf{b}_h$)
 2:     Given initial data $\mathbf{w}_h^0$, system matrix $\mathcal{A}_h$ and right-hand side $\mathbf{b}_h$
 3:     if ($h = h_{\max}$) then
 4:         Solve the coarse grid problem $\mathcal{A}_h \mathbf{w}_h = \mathbf{b}_h$ via the coarse grid solver
 5:     else
 6:         Apply presmoothing steps $\mathbf{w}_h^i = \mathcal{S}_h \mathbf{w}_h^{i-1}$ for $i = 1, \ldots, s$
 7:         Solve $\mathcal{A}_{2h} \mathbf{w}_{2h} = \mathbf{b}_{2h}$ by calling MGS($0, \mathcal{A}_{2h}, \mathbf{b}_{2h}$)
 8:         Update multigrid solution $\mathbf{w}_h^{s+1} = \mathbf{w}_h^s + \mathcal{R}^h_{2h} \mathbf{w}_{2h}$
 9:         Apply postsmoothing steps $\mathbf{w}_h^i = \mathcal{S}_h \mathbf{w}_h^{i-1}$ for $i = s+2, \ldots, 2s+1$
10:         Define solution $\mathbf{w}_h = \mathbf{w}_h^{2s+1}$
11:     end if
12:     return $\mathbf{w}_h$
13: end function
```

The smoothing steps in line 6 and 9 of Algorithm 5.1 can be accomplished by calling a suitable (preconditioned) iterative solver for a fixed number of iterates, say s times. In this work we consider JAC or a preconditioned BICGSTAB or GMRES as smoother. For further remarks about how a proper smoother can be derived we defer the interested reader to the Appendix D.

5.3.2. Numerical results for multigrid

We study the multigrid solver applied on the stationary variant of the minimal model of chemotaxis in its non-dimensional form, namely

$$\begin{cases} 0 = \partial_t u = \nabla \cdot \left(d_u \nabla u - u \chi \nabla v \right), \\ 0 = \partial_t v = \Delta v + u - v, \end{cases} \tag{5.3.1}$$

defined on two different computational domains, QUAD1 and CIRC1, cf. Table 5.2 and Figure 5.1. The objective is to capture the property of multigrid to require a discretization-independent number of iterations per nonlinear step, referred to as IT_LIN. Or, to put it in other words, we expect a similar number of iterations for different top-level discretizations, whereas for standard singlegrid solver we conjecture that this number increases significantly with finer discretizations. Moreover, we investigate the robustness of the multigrid algorithm.

For our numerical tests we use a stationary solver for system (5.3.1). Furthermore we complement the nonlinear problem with the initial guess

$$\begin{aligned} u_0 &= (1-0.01)\, u^*, \\ v_0 &= (1-0.01)\, v^*. \end{aligned}$$

The functions u^* and v^* are prescribed sinusoidal solutions with zero Dirichlet and flux boundary conditions, i.e.

$$\text{QUAD1}: \begin{cases} u^* = (1/40)\left[\cos\left(2\pi(x_2-0.5)\right)+1\right]\left[\cos\left(2\pi(x_1-0.5)\right)+1\right], \\ v^* = (1/80)\left[\cos\left(2\pi(x_2-0.5)\right)+1\right]\left[\cos\left(2\pi(x_1-0.5)\right)+1\right], \end{cases} \tag{5.3.2}$$

$$\text{CIRC1}: \begin{cases} u^* = (1/20)\left[\cos\left(\pi\sqrt{(x_1^2+x_2^2)}\right)+1\right], \\ v^* = (1/40)\left[\cos\left(\pi\sqrt{(x_1^2+x_2^2)}\right)+1\right]. \end{cases} \tag{5.3.3}$$

In order to yield a steady solution for (5.3.1) we augment the equations by corresponding right-hand sides, namely we actually solve

$$\begin{cases} 0 = \partial_t u = \nabla \cdot \left(d_u \nabla u - u \chi \nabla v \right) + g_u, \\ 0 = \partial_t v = \Delta v + u - v + g_v, \end{cases} \tag{5.3.4}$$

where g_u and g_v are the right-hand sides obtained by substituting (5.3.2) or (5.3.3) in (5.3.4) in terms of

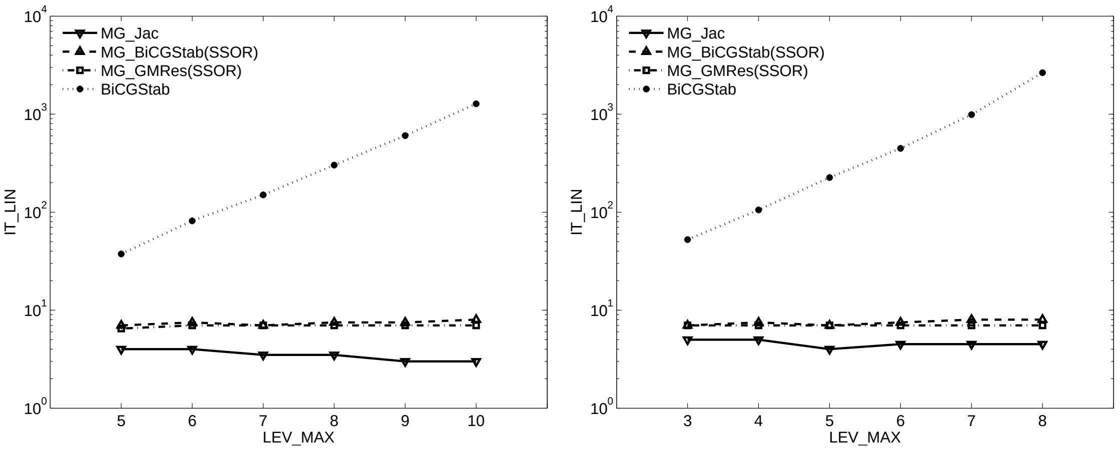
$$\begin{aligned} g_u &= -\nabla \cdot \left(d_u \nabla u^* - \chi u^* \nabla v^* \right), \\ g_v &= -\Delta v^* - u^* + v^*. \end{aligned}$$

In order to allow for a reasonable comparison of multigrid and singlegrid solver (in terms of IT_LIN), we modify the termination criteria for the underlying solver. The nonlinear iteration, which is tackled by the Newton method, is terminated solely by reaching the absolute criterion with $\varepsilon_a = 1E$-12. For the inner linear system, which is solved by the multigrid or singlegrid algorithms, we solely employ a relative termination criterion with $\varepsilon_r^{lin} = 1E$-6.

As reference singlegrid solver we use the standard preconditioned iterative Krylov-space solver BiCGStab(Jac). The multigrid solver is configured to use a W-cycle and a standard direct coarse grid solver provided by UMFPACK. As the smoothers play a key-role in a successful application of multigrid, we focus our attention on five different choices, i.e., Jac, GMRES(Jac), GMRES(SSOR), BiCGStab(Jac) and BiCGStab(SSOR). The corresponding multigrid solver are denoted by MG($\cdot$), e.g., MG(GMRES(Jac)).

Mesh-independence of IT_LIN

Let us begin with examining the number of iterations that the multigrid and singlegrid solver require to obtain a converging steady-state solution of (5.3.4). We simulate the system (5.3.4) with the model parameters $\chi = 1 = d_u$, on which we subsequently refer to as (PS1), on successively refined meshes for QUAD1 and CIRC1. The coarse level is set to 4 and 2, respectively for the former and latter mesh. Figure 5.4 depicts the corresponding averaged IT_LIN showcases in logarithmic scale. Therein we provided the results for the reference singlegrid solver BiCGStab(Jac) and for the three multigrid solver MG(Jac), MG(GMRES(SSOR)) and MG(BiCGStab(SSOR)). The plots for the two multigrid solver MG(GMRES(Jac)) and MG(BiCGStab(Jac)) are omitted since their results are very similar to the ones obtained by MG(GMRES(SSOR)) and MG(BiCGStab(SSOR)).



(a) QUAD1

(b) CIRC1

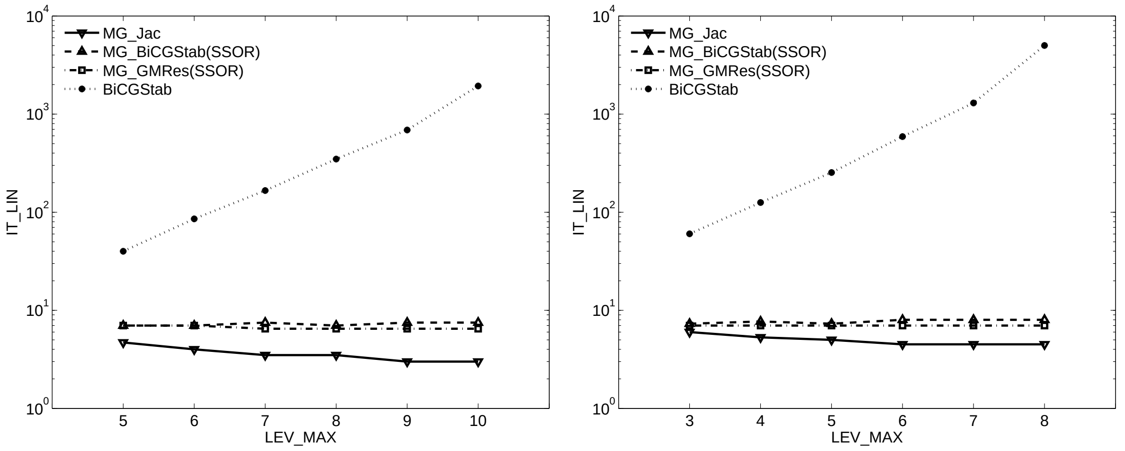
Figure 5.4: Average number of linear iterations for an increasing spatial refinement with the parameter setting (PS1). For QUAD1 the levels are ranging from LEV 5 (1024 $\mathcal{Q}_1$ elements) to LEV 10 ($1,048,576$ $\mathcal{Q}_1$ elements), where the coarse level is set to LEV 4. For CIRC1 the levels are ranging from LEV 3 (576 $\mathcal{Q}_1$ elements) to LEV 8 ($589,824$ $\mathcal{Q}_1$ elements), where the coarse level is set to LEV 2.

The results for both computational domains reveal that IT_LIN remains nearly constant for all multigrid variants. Among these different variants, MG(JAC) provides the minimal averaged number of linear iterations. Moreover, one advantage of JAC (besides its algorithmic simplicity) is the fact that no further choice of relaxing parameter is required (e.g., in contrast to SSOR), i.e., its application is somehow more convenient. Furthermore, we observe that both multigrid variants with a preconditioned smoother lead to very much comparable results.

In contrast to this mesh-independent number of iterations, we roughly observe a doubling for the reference singlegrid solver BICGSTAB(JAC) for both computational domains. This renders the singlegrid solver rather costly and unpractical at a high spatial discretization level.

Robustness of multigrid

After we have confirmed our expectations concerning the exerted number of iterations for the multigrid and singlegrid solver, in this section we turn to the analysis of the robustness of these solvers in terms of increasing values of chemosensitivity. We focus on the same setting of the numerical test as before, but consider two additional configurations of the model parameters, namely (PS2):$(d_u,\chi) = (1,40)$ and (PS3):$(d_u,\chi) = (0.1,100)$. Figure 5.5 and Figure 5.6 provide the development of IT_LIN for the multigrid and singlegrid solver for the parameter setting (PS2) and (PS3), respectively. According to the numerical tests of Figure 5.4, we confine the following analysis to the three variants of multigrid mentioned before.

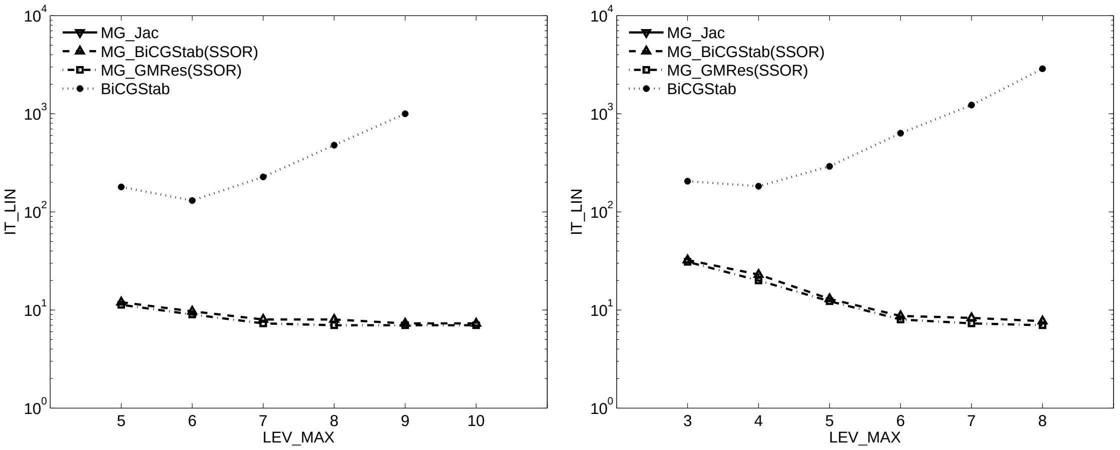


(a) QUAD1

(b) CIRC1

Figure 5.5: Average number of linear iterations for an increasing spatial refinement with the parameter setting (PS2). The range of levels for QUAD1 and CIRC1 are the same as in Figure 5.4.

We observe that there is only a marginal difference between the results for (PS1) and (PS2), cf. Figure 5.4 and Figure 5.5. However for (PS3) there are notable differences in IT_LIN among the solver algorithms. The missing plot for MG(JAC) in Figure 5.6 is caused by a diverging solution for all spatial mesh levels under consideration. This drawback clearly dominates the advantages of the simple implementation of JAC in terms of robustness. Moreover for QUAD1, the singlegrid solver diverges for the finest mesh level (therefore the last data is omitted). In contrast to these problems in terms of numerical convergence, the multigrid solver with a preconditioned smoother still provide reliable and nearly mesh-independent numbers of linear iterations. Again, let us re-



(a) QUAD1

(b) CIRC1

Figure 5.6: Average number of linear iterations for an increasing spatial refinement with the parameter setting (PS3). The range of levels for QUAD1 and CIRC1 are the same as in Figure 5.4. Note that the omitted data for MG(JAC) and for BICGSTAB(JAC) are caused by divergence of the corresponding solver.

mark that these latter results are similar to those obtained by the omitted variants of multigrid with preconditioned smoothers, namely MG(GMRES(JAC)) and MG(BICGSTAB(JAC)).

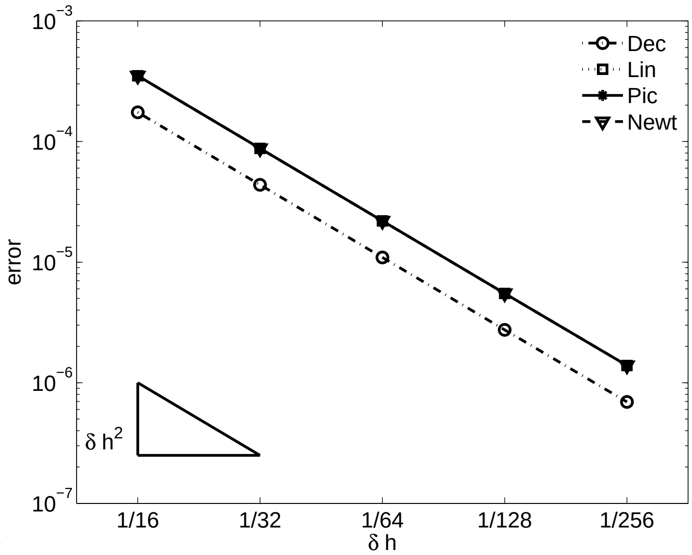
Concluding our observations, we recommend the use of a multigrid solver particularly for a high level of the spatial discretization, since in this case the exploding number of linear iterations render standard singlegrid solver unpractical. Moreover, we advise the use of a preconditioned Krylov-space smoother since these smoothers are more robust in terms of increasing chemosensitivities as demonstrated above.

5.4. Numerical comparison of the iteration schemes

After having investigated the application of multigrid solver to chemotaxis PDEs, we will focus on qualitative and quantitative comparisons of the different FEM discretization approaches introduced in Chapter 4, namely DEC, LIN, PIC and NEWT. We will present basic studies of convergence and a study of the efficiency of the four approaches. The convergence will be demonstrated on the basis of the stationary minimal model by employing artificial transients. The study of the efficiency provides a glance on the relation between required iterations and accuracy on the basis of highly transient chemotaxis models. These evaluations pursue the ideas of Strehl *et al.* in [97] and extend the results therein by considering more methodologies, namely including a multigrid solver and the nonlinear Picard iteration.

5.4.1. Convergence analysis of the stationary minimal model

In this primary study we examine the convergence of the different solver methodologies in terms of the spatial discretization. To simplify the analysis we confine ourselves to the stationary minimal model (5.3.1), however similar results have also been obtained for the stationary variants of the other exemplary models.



htb

Figure 5.7: Stationary minimal Model. Convergence towards analytical solution.

The convergence is measured towards an analytically given reference solution. The numerical approximations are obtained via introducing so-called artificial transients into (5.3.1) and simulating till steady state. Similar to the underlying model for the previous multigrid analysis, the analytic solutions u^* and v^* are given by the prescribed sinusoidal solutions (5.3.2). The corresponding computational domain is $\Omega_h = [0,1]^2$. Remark that the reference solutions (5.3.2) satisfy zero Dirichlet and Neumann boundary conditions. In order to capture the analytical reference solutions, we augment the equations by corresponding right-hand sides and again solve the modified model (5.3.4).

To simplify the calculations we consider rather simple initial conditions which are already close to the analytic solutions. Here we prescribe slightly perturbed analytical solutions, in detail

$$\begin{aligned} u_0 &= (1-0.1\,\mathrm{rand}(\mathbf{x}))\,u^*, \\ c_0 &= (1-0.1\,\mathrm{rand}(\mathbf{x}))\,v^*, \end{aligned}$$

with $\mathrm{rand}(\mathbf{x})$ denoting $[0,1]$-uniformly distributed random numbers in every coordinate.

For stability concerns steady state simulations are performed by the implicit Euler discretization in time of the artificial transients, i.e., in terms of the theta-scheme, we set $\theta = 1$. The approximation of the steady state of the numerical solution will be checked by the standard first order divided difference with a tolerance of $1E$-6, namely the simulation stops if $||w_h^n - w_h^{n-1}||/\delta t^* < 1E$-6, where $w_h^n = (u_h^n, v_h^n)$ denotes the numerical solution at the n^{th} time step. As a reference time stepping we choose $\delta t^* = 1E$-2.

Concerning the spatial convergence, we expect all numerical schemes to be of second order stemming from the underlying $\mathcal{Q}_1$ element space. In Figure 5.7 we plotted the convergence behavior for successive spatial refinements ranging from $\delta h = 1/16$ to $\delta h = 1/256$. From the figure we can numerically readily confirm the quadratic convergence of all four schemes. Moreover we recognize that all monolithic schemes, i.e., LIN, PIC and NEWT, provide effectively similar error estimates and that the errors of DEC are slightly superior to the monolithic counterparts. However, we remark that in return the monolithic schemes naturally give rise to a fully stationary solver, i.e., without artificial transients, whereas the decoupled scheme does not.

5.4.2. Efficiency of the iteration schemes

In contrast to the previous study, this section deals with the investigation of the numerical efficiency of the four discretization techniques DEC, LIN, PIC and NEWT applied on transient models for chemotaxis. Beside the transient counterpart of the aforementioned minimal model, we also consider an aggregation and a kinetic model. In order to examine the convergence behavior in time and space, reasonable numerical reference solutions are pre-calculated. In other words, we solve the underlying PDE systems with Newtons method at a properly refined spatial mesh and time stepping, say $\delta h^*, \delta t^* \ll 1$, up to a specified time instance, say T. The numerical error is then estimated in terms of $||\mathbf{w}_h^t - \mathbf{w}_{h^*}^{t^*}||$, where $\mathbf{w}_h^t$ denotes the numerical solution on the discretization level corresponding to $\delta h > \delta h^*$ and $\delta t > \delta t^*$.

Measurements of the numerical efficiency

As already proposed in [97], in this investigation the numerical efficiency is understood as the ratio of accuracy and computational costs/complexity, say

$$\text{efficiency} \quad = \quad \frac{\text{accuracy}}{\text{computational costs}}. \tag{5.4.1}$$

For the remainder we will refer to the efficiency and computational costs as EFF and COSTS, respectively. While the accuracy is measured in terms of the numerical error estimation, ERR, given above, the computational costs are to be understood as required number of iterations. Although the author is aware of the simplicity of this measurement (e.g., neglecting vector/matrix assemblies, memory concerns or overall computing time), it already serves as a valuable yardstick for our concerns.

Let us explain in detail how we calculate the ratio of efficiency of the underlying iteration schemes for our purposes. We will always compare the efficiency of two schemes, say A and B, by the relation $\text{EFF}_\text{A}/\text{EFF}_\text{B}$,

$$\frac{\text{EFF}_\text{A}}{\text{EFF}_\text{B}} \quad = \quad \frac{\text{ERR}_\text{B}}{\text{ERR}_\text{A}} \cdot \frac{\text{COSTS}_\text{B}}{\text{COSTS}_\text{A}}.$$

We apply our definition on the data obtained from successive choices of time stepping, namely for each δt we compute the error (to a numerical reference solution) and the averaged number of linear iterations per time step for both schemes A and B. Moreover, we study the development of the efficiency as the value of the chemosensitivity is increased. In other words, we are interested in how the efficiency of the numerical schemes scales with the chemosensitivity.

Particularly when considering the four underlying solver schemes, the following remarks concerning the suggested efficiency (5.4.1) are in order: (i) while DEC, PIC and NEWT exert a nonlinear iteration, LIN is purely linear. (ii) all schemes except DEC are based on a monolithic block discretization, therefore the complexity of their underlying solver is higher in comparison to DEC (iii) moreover all monolithic schemes employ a multigrid solver for the linear systems, whereas DEC uses a standard single grid solver (iv) employing a nonlinear iteration and a monolithic approach, as in the case of PIC and NEWT, possibly enhances the accuracy in contrast to purely linearized or decoupled approaches, as in the case of Lin and Dec.

Thus, there is no doubt that a comprehensive evaluation of such concerns are of utmost interest when recommending a particular solver scheme. Table 5.4 roughly presents the complexities of the four schemes. Again, the author would like to mention that this does not serve as a detailed look-up-table, since it blurs memory and matrix-vector-assembly concerns.

	Dec ($\star$)	Lin	Pic	Newt
per time step	2 RHS 1 SOL	1 BLK-RHS 1 BLK-MAT 1 BLK-SOL	1 BLK-RHS	1 BLK-RHS
per nonlin. it.	1 MAT 1 SOL	–	1 BLK-MAT 1 BLK-SOL	2 BLK-MAT 1 BLK-SOL

Table 5.4: A rough sketch of the complexities for the four underlying schemes. RHS, MAT and SOL denote the right-hand side and matrix to be built and the call of the linear solver, respectively. BLK represents the block extensions for the monolithic approaches. ($\star$) The system matrix for the v equation can be built in advance, say once at the beginning of the simulation.

In what follows, we will present the convergence results and the corresponding statements concerning the efficiency of the four different numerical schemes applied on the three transient chemotaxis PDE models.

The transient minimal model of chemotaxis

This paragraph is concerned with the convergence analysis of the transient minimal model of chemotaxis (4.1.2) on the unit square QUAD1. For reasons of readability we recapitulate the equations,

$$\begin{aligned} \partial_t u &= \nabla \cdot \left(d_u \nabla u - u \chi \nabla v \right), \\ \partial_t v &= \Delta v - v + u . \end{aligned}$$

For the upcoming simulations we choose the initial conditions

$$\begin{aligned} u_0 &= 100\, e^{-100\,[(x-0.5)^2+(y-0.5)^2]}, \\ v_0 &= 0 . \end{aligned} \tag{5.4.2}$$

The termination criteria of the underlying (non-)linear solver are set up as described in Section 5.2 with $\varepsilon_r = 1E\text{-}6$ and $\varepsilon_a = 1E\text{-}14$.

The transient minimal model of chemotaxis – basic convergence analysis

We begin by studying the spatial convergence of the numerical solution, i.e., we fix δt and observe the behavior as δh increases. The underlying numerical reference solution was obtained with

δh	$1/256$	$1/128$	$1/64$
DEC	2.2594E-03	1.2949E-02	5.5540E-02
LIN	2.7442E-03	1.3452E-02	5.6042E-02
PIC, NEWT	2.7437E-03	1.3452E-02	5.6042E-02

Table 5.5: Minimal Model. Convergence of the (spatial) error.

$\delta h^* = 2^{-9}$ (resulting in $262,144$ $\mathcal{Q}_1$ elements) and $\delta t^* = 1E\text{-}5$. The simulation was driven up to the time instance $T = 1.28E\text{-}3$. The error towards the numerical reference solution is depicted in Table 5.5. Therein, we list the error for all four iteration schemes at successive mesh refinements. We recognize that all schemes provide a reliable $\mathcal{O}(\delta h^2)$ convergence as the spatial mesh is refined. Note that PIC and NEWT effectively have the same error estimates. Our observations confirm the proper choice of the (reference) time stepping, namely all schemes effectively result in the same solution and the temporal error contribution is neglectable when δt is chosen appropriately.

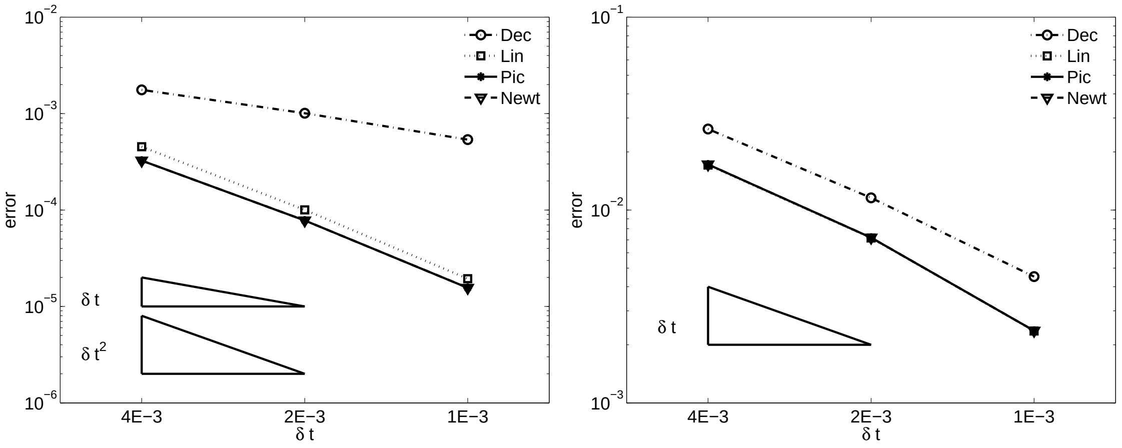
When considering the pure temporal error contribution we issue numerical simulations at one distinct mesh level and vary the time stepping δt. For the upcoming results we fixed $\delta h^* = 2^{-7}$, which already serves as a reasonable mesh level resulting in $16,384$ elements. Furthermore the time stepping for the numerical reference solution was chosen to be $\delta t^* = 5E\text{-}4$. Figure 5.8 shows the corresponding convergence results for the model parameters $\chi = 1 = d_u$ with the temporal discretization tackled by Crank-Nicolson (left figure) and Backward Euler (right figure). Let us draw our first attention to Crank-Nicolson. The figure reveals that the decoupled scheme converges linearly while all monolithic schemes provide a quadratic convergence behavior. This results in error estimates for DEC that are approximately 5 up to 34 times larger than the ones for NEWT. Note that, once again, the error estimates for PIC and NEWT are essentially the same, therefore their corresponding plots coincide. As a second observation we remark that the error estimates for the linearized scheme are up to 40% larger compared to PIC and NEWT.

Coming to the Backward Euler case we recognize that all four schemes promote a linear convergence. The error estimates for all monolithic schemes effectively coincide. However the decoupled variant provides a poorer numerical solution, the corresponding error is approximately $50-90\%$ larger than the one of the monolithic schemes.

The transient minimal model of chemotaxis – study of increasing chemosensitivity

After we have provided basic studies of the convergence behavior of all numerical schemes, we turn our focus now to the simple measurement of efficiency as introduced above. Because all four schemes reveal a similar spatial convergence behavior but a different temporal one, we draw our attention to the temporal accuracy. Indeed, the control of the time stepping allows us for better resulting statements of efficiency, since δt can be chosen arbitrary while $1/\delta h$ is restricted to take only distinct values of powers of two.

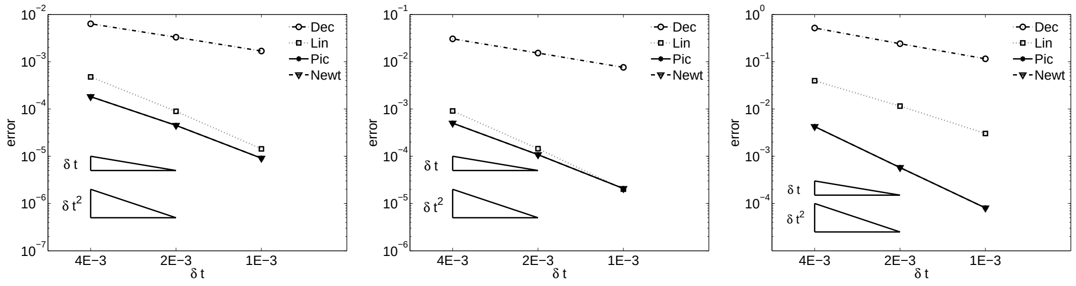
From the numerical and from the application point of view the four numerical schemes should reliably cope with reasonable variations of the model parameters. In our case of chemotaxis-driven processes we will focus on different values of the chemosensitivity χ. Figure 5.9 depicts the temporal error convergence for the Crank-Nicolson discretization. The value of χ is successively



(a) Crank-Nicolson

(b) Implicit Euler

Figure 5.8: Minimal Model. Convergence of the temporal error. **Left:** Crank-Nicolson. **Right:** Implicit Euler. The four plots represent the different solver schemes, DEC, LIN, PIC and NEWT. The time stepping was chosen $\delta t = 4E\text{-}3, 2E\text{-}3, 1E\text{-}3$.



(a) $\chi = 2$

(b) $\chi = 4$

(c) $\chi = 8$

Figure 5.9: Minimal Model. Convergence of the temporal error for $\chi = 2$, $\chi = 4$ and $\chi = 8$. The four plots represent the different solver schemes, DEC, LIN, PIC and NEWT. The time stepping was chosen $\delta t = 4E\text{-}3, 2E\text{-}3, 1E\text{-}3$.

doubled up to $\chi = 8$. The numerical reference solution is computed as above with a corresponding value of χ. From observing these plots three main points are eye-catching.

1. Firstly, three data sets are distinguishable. The two nonlinear Richardson schemes PIC and NEWT essentially provide the same error estimates, while the decoupled and linearized variants reveal characteristic differences. The decoupled discretization leads to the poorest approximation, followed by the explicitly linearized scheme. Both Richardson schemes provide the most accurate results.

2. Secondly, the convergence rates, as demonstrated above, are mainly preserved. While DEC only reveals a first order error reduction, all three monolithic schemes scale well with a quadratic error convergence.

3. The final observation concerns the development of the error for the three data sets as χ is increasing. We recognize that the error increases for all schemes when χ grows, which eventually leads to remarkable differences in accuracy for the three distinguishable data sets. For $\chi = 8$, Figure 5.9 reveals that we roughly gain one digit of accuracy when comparing

	LIN			PIC			NEWT		
$\delta t \backslash \chi$	2	4	8	2	4	8	2	4	8
$1E$-3	6.7	6.8	7.3	9.6	10.0	10.3	7.2	7.0	6.7
$2E$-3	7.3	7.5	7.9	10.0	10.5	13.8	7.0	7.0	7.0
$4E$-3	8.2	8.4	9.0	11.6	12.2	16.5	7.6	8.1	8.4

Table 5.6: Minimal Model. Averaged total number of linear iterations for all monolithic schemes.

DEC with LIN or LIN with either PIC or NEWT at the coarsest time stepping (for smaller δt we even gain up to more than 30 times the accuracy).

The transient minimal model of chemotaxis – basic efficiency analysis

In order to get a first idea of the efficiency of the four proposed schemes (in fact we will consider only three data sets in the following since the two Richardson schemes are very much similar), we assume that the plots of convergence in Figure 5.9 can be continued up to an arbitrary fine time stepping. In other words, for an arbitrary small $\delta t < 5E$-4 and a corresponding numerical reference solution, the error estimates for the four iteration schemes with $\delta t = 4E$-3, $2E$-3, $1E$-3 coincide with the data given in the figures 5.9. If their slope is also maintained we can state the following:
Consider the errors of DEC and LIN for $\chi = 2$. In order to drop the error of DEC to the level of LIN, we have to refine the time stepping more than 13 times, i.e., even with $\delta t = 5E$-4 DEC provides a larger error than LIN with $\delta t = 4E$-3. As it is not very handy to quantitatively compare the required iterations of DEC with any of the monolithic schemes (recapitulate that these schemes employ a multigrid solver for the block system rather than a single grid solver for the decoupled scalar schemes), we skip this comparison for the remainder of this section and proceed with analyzing the three monolithic schemes in more detail.

We begin with commenting on PIC and NEWT. We already saw that the corresponding error estimates effectively coincide, hence from the aspect of accuracy these schemes are identical. When considering the averaged numbers of iterations however, a slight difference can be observed. Table 5.6 provides the averaged number of linear iterations for all monolithic schemes.

Let us have a look on the case $\chi = 2$. The Picard scheme employs up to 11.6 iterations in average, whereas Newton's scheme only requires up to 7.6 iterations. Therefore, in terms of the efficiency ratio defined in equation (5.4.1), we can deduce that NEWT is approximately up to 1.5 times as efficient as PIC. For $\chi = 4$ this ratio effectively does not alter significantly, whereas for $\chi = 8$ we arrive at a efficiency ratio up to 2.0, i.e., NEWT is up to twice as efficient as PIC. Table 5.7 summarizes all ratios of efficiency for all configurations.
Let us turn to LIN and PIC. Note that now the ratio of efficiency takes into account the different error estimates. From Table 5.7 we observe that the ratio of efficiency $\text{EFF}_{\text{PIC}}/\text{EFF}_{\text{LIN}}$ for $\chi = 2$ ranges from 1.1 to 2.0. For the large value of the chemosensitivity, $\chi = 8$, the difference of the two schemes is more pronounced, $\text{EFF}_{\text{PIC}}/\text{EFF}_{\text{LIN}}$ reaches from 5.0 to 25.0. This clearly renders the nonlinear Richardson scheme to be highly favorable in case of high nonlinearities (in terms of χ).

The combination of both ratios of efficiency allows us to estimate $\text{EFF}_{\text{LIN}}/\text{EFF}_{\text{NEWT}}$ by means

	$\text{EFF}_{\text{Newt}}/\text{EFF}_{\text{Pic}}$			$\text{EFF}_{\text{Pic}}/\text{EFF}_{\text{Lin}}$			$\text{EFF}_{\text{Newt}}/\text{EFF}_{\text{Lin}}$		
$\delta t \backslash \chi$	2	4	8	2	4	8	2	4	8
$1E$-3	1.3	1.4	1.5	1.1	0.7	25.0	1.4	0.9	37.5
$2E$-3	1.4	1.5	2.0	1.4	1.0	20.0	2.0	1.5	40.0
$4E$-3	1.5	1.5	2.0	2.0	1.3	5.0	3.0	1.9	10.0

Table 5.7: Minimal Model. Ratios of efficiency $\text{EFF}_{\text{Newt}}/\text{EFF}_{\text{Pic}}$, $\text{EFF}_{\text{Pic}}/\text{EFF}_{\text{Lin}}$ and $\text{EFF}_{\text{Newt}}/\text{EFF}_{\text{Lin}}$ for all configurations.

of

$$\frac{\text{EFF}_{\text{Newt}}}{\text{EFF}_{\text{Lin}}} = \frac{\text{EFF}_{\text{Newt}}}{\text{EFF}_{\text{Pic}}} \cdot \frac{\text{EFF}_{\text{Pic}}}{\text{EFF}_{\text{Lin}}}.$$

This ratio takes values of up to 3.0 and 40.0 for $\chi = 2$ and $\chi = 8$, respectively.

Briefly concluding, we noticed that the efficiency of the two nonlinear Richardson schemes only moderately differs. Particularly when considering the additional costs of assembling the Jacobian for Newton's scheme, we do not intent to state a clear recommendation for either of these schemes. Hence, in this context more sophisticated testing is required. However, we definitely observed that the linearized scheme cannot be recommended for large chemosensitivities.

Remark 5.1 *The choice of the employed initial conditions for the minimal model of chemotaxis is delicate, because of the possible emerging of a singularity in terms of a blowing-up solution. With an initial cell concentration u_0 as in (5.4.2) we have $\chi||u_0||_{L^1} < 4\pi$. Thus we expect the solution to exist globally in time, cf. the cited results in Section 2.2. Therefore the analysis of the convergence can be undertaken as usual. However, if we prescribe an initial cell distribution which leads to a blow-up, numerical convergence is only valid before the blow-up time, which, in turn, is hard to determine – up to the authors best knowledge the precise theoretical determination of the blow-up time is still an open question. This is also the reason of the rather moderate increase of χ in the context of the study for the numerical efficiency. Indeed $\chi = 8$ is the critical value of global existing solutions of the minimal model (4.1.2).*

The transient aggregation model

Let us now turn to the aggregation model that has already been introduced in Section 4.1. For comprehensibility we recall the governing equations (4.1.3),

$$\begin{aligned} \partial_t u &= \nabla \cdot \left(d_u \nabla u - \chi \frac{u}{(1+v)^2} \nabla v \right), \\ \partial_t v &= \Delta v + \frac{u^2}{1+u^2}. \end{aligned}$$

This model introduces a nonlinear coefficient in the chemotaxis term and in the chemical production rate. Since these terms determine the 'strength' of coupling, we might expect the decoupled scheme to provide poorer results compared to the monolithic variants. Moreover, the nonlinearity itself possibly gives rise to further distinguished (subtle) convergence behavior of the different nonlinear treatments in our iteration schemes. For the upcoming numerical convergence tests we focus on the unit square QUAD1. As initial conditions we prescribe a cell distribution consisting of five initial aggregates (cf. the first subfigure in Figure 5.10) whereas the chemical concentration is zero everywhere, i.e.

$$\begin{cases} u_0(\mathbf{x}) &= 0.9 + 0.05 \left[\cos\left(4\pi(x_1 - 0.25)\right) + 1 \right] \left[\cos\left(4\pi(x_2 - 0.25)\right) + 1 \right] \\ & \quad + 0.2\, e^{-100[(x_1-0.55)^2 + (x_2-0.6)^2]}, \\ v_0(\mathbf{x}) &= 0. \end{cases} \tag{5.4.3}$$

The exponential part of the initial cell distributions renders u_0 unsymmetrical. The idea behind this choice is to observe the preferences of aggregation to occur when single aggregates are initially connected. In fact, numerically it can be shown (cf. Figure 5.10) that in case of only weak (or the absence of) connectivity, single aggregates tend to prefer to agglomerate at the boundaries. Hence for our setting the two bottom-peaks from the initial condition initially tend to move to the lower boundary. Just after all three top-peaks merge together, the two bottom-peaks start to feel more attracted to this merged aggregate (the bottom-right peak being more excited than the bottom-left). Note that the color-coding is relative to the particular snapshot, so that the cell density of different snapshots cannot be easily compared ($u_{\max}(t=0) \approx 1.3$ but $u_{\max}(t=0.85) \approx 412$). Especially, let us stress that the total cell mass is conserved during our simulation.

The transient aggregation model – basic convergence analysis

The numerical reference solution is computed at a discretization level with $\delta h^* = 2^{-9}$ and $\delta t^* = 1E$-5. The simulation run up to $T = 1.28E$-3. The termination criteria remain unchanged. Table 5.8 provides the error estimates for the spatial convergence. We observe that all schemes roughly converge in accordance to the expected $\mathcal{O}(\delta h^2)$ order. Moreover all monolithic schemes essentially produce the same error and the error of the decoupled variant only differs in the 7th/8th digit. The reason for this coincidence is the rather subtle temporal error introduced by the fine time stepping.

After the spatial convergence we now take a look on the temporal convergence behavior. For numerical convenience we use $\delta h^* = 2^{-6}$ and $\delta t^* = 3.125E$-7 as reference discretization and simulate to $T = 1.28E$-3. Figure 5.11 depicts the convergence plots of all four iteration schemes. As before the left and right figures depict the temporal error obtained from the Crank-Nicolson and Implicit Euler time discretization, respectively. We observe that certain data sets coincide and the corresponding plots cannot be distinguished in the figures. In the Crank-Nicolson case

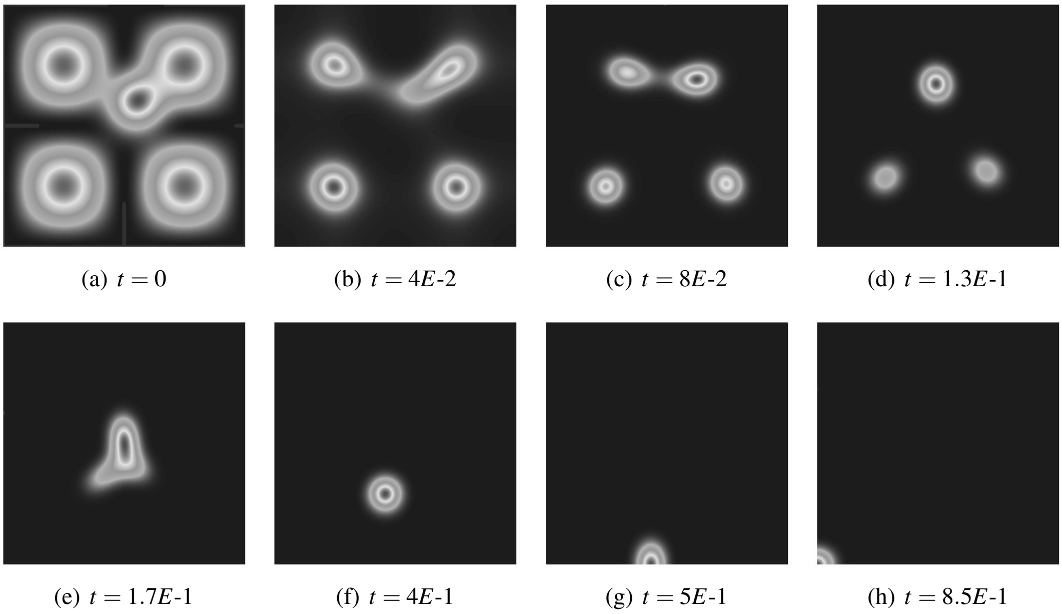


Figure 5.10: The dynamics of the aggregation model with $d_u = 0.33, \chi = 500$. Simulated with $\delta h = 2^{-7}$ and $\delta t = 1E$-3. Merging of the three top aggregates happens fast ($t = 8E$-2), while the lower aggregates require some time to 'sense' the attractive agglomeration (after $t = 1.3E$-1). Absence of interior attraction seems to lead to boundary (and finally corner) attraction (after $t = 1.7E$-1). Notably the aggregate moves first to the closest boundary ($t \sim 5E$-1) before it is directed to a corner ($t \sim 8.5E$-1).

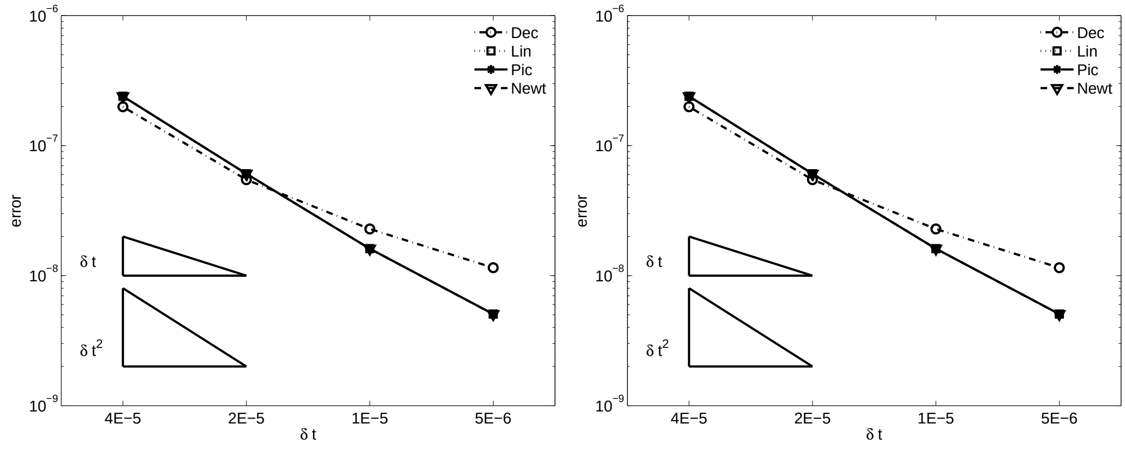
δh	1/256	1/128	1/64
DEC	1.1338E-05	5.5474E-05	2.3150E-04
LIN, PIC, NEWT	1.1366E-05	5.5502E-05	2.3152E-04

Table 5.8: Aggregation Model with $d_u = 1 = \chi$. Convergence of the (spatial) error.

all monolithic schemes essentially provide similar error estimates, i.e., their plots coincide. In the case of Implicit Euler effectively all four schemes reveal similar error estimates and we cannot distinguish their plots. Besides these first observations, we recognize in the Crank-Nicolson case that DEC converges linearly (at least for sufficient small δt), whereas the monolithic schemes converge quadratically as expected. Interestingly, DEC provides a better approximation for large time stepping before the linear convergence limits its accuracy towards the quadratically converging monolithic competitive schemes. In the case of Implicit Euler, the convergence is clearly linear.

The transient aggregation model – study of increasing chemosensitivity

As for the minimal model we now turn our focus to the study of increasing chemosensitivity. In Figure 5.12 we show the temporal convergence plots for the cases of $\chi = 100, 500, 1000$ with the Crank-Nicolson time discretization. The corresponding numerical reference solutions are obtained with $\delta h^* = 2^{-7}, \delta t^* = 5E$-6.



(a) Crank-Nicolson

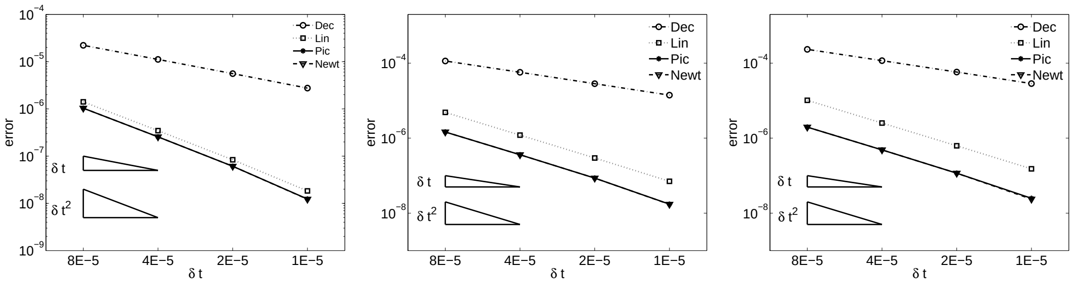
(b) Implicit Euler

Figure 5.11: Aggregation Model $d_u = 1 = \chi$. Convergence of the temporal error. The four plots represent the different solver schemes, DEC, LIN, PIC and NEWT. The time stepping was chosen $\delta t = 4E\text{-}5, 2E\text{-}5, 1E\text{-}5, 5E\text{-}6$.

Our observations are similar to the ones stated for the minimal model. The decoupled scheme performs poorest in terms of accuracy as it converges only linearly. Among the three monolithic schemes we can identify two data sets. The first set of error estimates consists of the data obtained from the two nonlinear Richardson schemes PIC and NEWT. For all three values of χ their plots effectively coincide and reveal a second order convergence of the temporal error. The linearized scheme LIN also provides a quadratic convergence, however its error estimates are significantly poorer. As χ increases the differences between the error estimates of the three data sets also increase. In other words, the strength of the chemosensitivity crucially influences the wealth of the four different iteration schemes. To emphasize our findings let us exemplarily analyze the increasing error gap for $\chi = 100$ and $\chi = 1000$. In the case of $\chi = 100$, LIN provides a 1.5 times poorer accuracy as the nonlinear Richardson schemes PIC and NEWT. Moreover, the error estimates of DEC are roughly 20–200 times poorer than the ones for PIC and NEWT. These factors significantly increase for $\chi = 1000$. In this case the approximations of LIN are even more than six times poorer than the ones for PIC and NEWT. Furthermore, the inaccuracy of the decoupled scheme is also enhanced and provide approximately 120–1200 times poorer results than PIC and NEWT. Or, to put it into the context of the increasing chemosensitivity: While the increase of chemosensitivity is ten-fold, the differences in accuracy between LIN and the two nonlinear Richardson schemes increase by a factor of nearly 4. For DEC and the two nonlinear Richardson schemes, this increase is approximately six-fold.

The transient aggregation model – basic efficiency analysis

For the analysis of the efficiency of the different schemes we proceed as described in the case of the minimal model. Firstly, we compare the two nonlinear Richardson schemes PIC and NEWT. Since their error estimates essentially coincide we can focus on their required averaged iterations, cf. Table 5.9. In the case of $\chi = 100$, PIC requires up to about 1.6 times more iterations than NEWT. Since the accuracy is almost identical, we can hence deduce that NEWT is 1.6 times more efficient in terms of our definition of efficiency, i.e., $\text{EFF}_{\text{NEWT}}/\text{EFF}_{\text{PIC}} = 1.5$. For increasing χ this ratio is augmented to 1.8 as PIC requires at most an average of 1.8 times the number of iterations as NEWT. In Table 5.10 we depict the ratios of efficiency for all schemes under consideration and



(a) $\chi = 100$ (b) $\chi = 500$ (c) $\chi = 1000$

Figure 5.12: Aggregation Model. Convergence of the temporal error for $\chi = 100$, $\chi = 500$ and $\chi = 1000$ on the spatial level with $\delta h^* = 2^{-7}$. The four plots represent the different solver schemes, DEC, LIN, PIC and NEWT. The time stepping was chosen $\delta t = 8E\text{-}5, 4E\text{-}5, 2E\text{-}5, 1E\text{-}5$.

	LIN			PIC			NEWT		
$\delta t \backslash \chi$	100	500	1000	100	500	1000	100	500	1000
1*E*-5	3.4	3.3	3.3	4.5	5.7	5.9	3.6	4.0	4.0
2*E*-5	4.0	4.0	4.0	6.0	6.8	7.2	4.0	4.0	4.0
4*E*-5	4.0	4.0	4.0	6.3	7.2	7.8	4.0	4.0	4.9
8*E*-5	4.1	4.1	4.1	6.9	8.4	8.0	4.6	5.1	5.7

Table 5.9: Aggregation Model. Averaged total number of linear iterations for all monolithic schemes.

for all configurations of δt and χ.

Secondly, let us compare the efficiencies of PIC and LIN. Because of the significant differences between the error estimation of these two schemes, we will additionally take into account the ratio of accuracy. In the case of $\chi = 100$ we do not obtain a clear picture of the resulting ratio of efficiency $\text{EFF}_{\text{PIC}}/\text{EFF}_{\text{LIN}}$. Calculations revealed that it reaches from 0.8 to 1.1, i.e., depending on the time stepping either LIN is up to 25% more efficient than PIC or PIC is 10% more efficient than LIN. However for larger χ this relation becomes clear. For $\chi = 500$ the ratio is always greater than 1.0 and lies in the range of $1.6 - 2.4$, i.e., now PIC is up to 2.4 times as efficient as LIN. For $\chi = 1000$ this becomes even more pronounced as the ratio now increases up to 3.6, i.e., PIC is 3.6 times as efficient as LIN.

Finally, combining the results for the ratios of efficiency $\text{EFF}_{\text{NEWT}}/\text{EFF}_{\text{PIC}}$ and $\text{EFF}_{\text{PIC}}/\text{EFF}_{\text{LIN}}$ allows us to approximate the ratio of efficiency $\text{EFF}_{\text{NEWT}}/\text{EFF}_{\text{LIN}}$. For $\chi = 100, 500, 1000$ the corresponding ratio reaches up to 1.7, 3.4 and 5.4, respectively.

We conclude that the ratio of efficiency scales with the chemosensitivity of the aggregation model, e.g .the value of χ. From the analysis of LIN and PIC we can also state that for moderate values of χ the linearized scheme seems to be preferable, particularly in the case of a coarse time stepping. With regard to the additional costs of assembling the Jacobian for Newton's scheme, the moderate ratio of efficiency of $\text{EFF}_{\text{NEWT}}/\text{EFF}_{\text{PIC}}$ does not allow for a clear recommendation for either of those two nonlinear Richardson schemes.

	EFF$_{\text{NEWT}}$/EFF$_{\text{PIC}}$			EFF$_{\text{PIC}}$/EFF$_{\text{LIN}}$			EFF$_{\text{NEWT}}$/EFF$_{\text{LIN}}$		
$\delta t \backslash \chi$	100	500	1000	100	500	1000	100	500	1000
1E-5	1.3	1.4	1.5	0.8	2.4	3.6	1.0	3.4	5.4
2E-5	1.5	1.7	1.8	0.9	2.0	3.0	1.4	3.4	5.4
4E-5	1.6	1.8	1.6	0.9	1.9	2.7	1.4	3.4	4.3
8E-5	1.5	1.6	1.4	1.1	1.6	2.7	1.7	2.6	3.8

Table 5.10: Aggregation Model. Ratios of efficiency EFF$_{\text{NEWT}}$/EFF$_{\text{PIC}}$, EFF$_{\text{PIC}}$/EFF$_{\text{LIN}}$ and EFF$_{\text{NEWT}}$/EFF$_{\text{LIN}}$ for all configurations.

The transient kinetic model

The last model under concern is the kinetic model (4.1.4). Here we focus on the model with $a = 0$ and remind the reader of the corresponding equations as provided in Section 4.1

$$\begin{aligned} \partial_t u &= \nabla \cdot \left(d_u \nabla u - \chi u \nabla c \right) + u^2 (1-u) \,, \\ \partial_t v &= \Delta v - \beta v + u \,. \end{aligned}$$

The main numerical challenge of this particular example of a chemotaxis model is the introduction of the kinetic term which gives rise to certain patterns. The sensitivity of those patterns require an accurate numerical scheme capturing the spatial and temporal resolution of evolving patterns in terms of sharp fronts and high transients, respectively.

Our simulations of this model were exerted on the enlarged square QUAD16 with a simulation end time of $T = 0.128$. The numerical reference solution was computed at the discretization level $\delta h^* = 2^{-5}$ and $\delta t^* = 1E\text{-}3$. Since models of this kind are known for their radially evolving traveling waves we start our simulations given the following cell/chemical distributions

$$\begin{aligned} u_0(\mathbf{x}) &= \begin{cases} 1 + 1.1 \cos^2 \left(\pi r_{8,8}(\mathbf{x}) \right) \,, & \text{for } r_{8,8}(\mathbf{x}) \leq 1.5 \\ 1 \qquad\qquad\qquad\quad , & \text{otherwise} \end{cases} \\ v_0(\mathbf{x}) &= 1/32 \,. \end{aligned}$$

Herein $r_{8,8}(\mathbf{x})$ denotes the Euclidean distance to the center point $(8,8)$ of the domain.

The transient kinetic model – basic convergence analysis

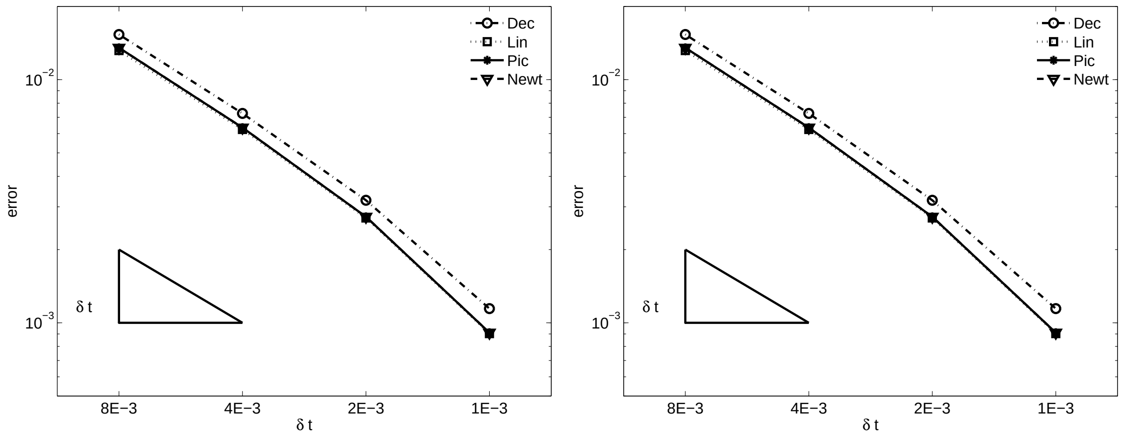
As before we provide error estimates for all of the four numerical schemes. We begin with the convergence in time. To this end let us have a look on Table 5.11. We remark that all numerical schemes provide a quadratic convergence in terms of $\mathcal{O}(\delta h^2)$, in other words the error is (at least) quartered with increasing level. Furthermore the computations reveal effectively the same errors for the both Richardson schemes PIC and NEWT.

When focusing on the temporal errors we obtain the classical behavior which has already been documented in [97]. Figure 5.13 shows the convergence plots for the Crank-Nicolson and the

δh	$1/16$	$1/8$	$1/4$
Dec	7.5570E-04	4.1882E-03	1.7062E-02
Lin	8.7983E-04	4.3157E-03	1.7184E-02
Pic, Newt	8.8227E-04	4.3182E-03	1.7186E-02

Table 5.11: Kinetic model with $d_u = 1 = \chi$. Convergence of the (spatial) error.

Implicit Euler time discretizations. Herein the numerical reference solution was computed with Newton's scheme and a discretization with $\delta h^* = 1/8$ and $\delta t^* = 5E$-4. As the spatial discretization level is fixed for the upcoming simulations we assume that no deteriorating spatial errors are introduced. First of all, in both cases we observe no essential difference in the error estimates for the Richardson schemes Pic and Newt, this is why the corresponding plots coincide. Secondly, in the case of Crank-Nicolson, the plots reveal first-order and second-order convergence for Dec on the one hand and for Lin, Pic and Newt on the other hand, respectively. In the case of an implicit Euler time discretization we obtain first-order convergence for all schemes as expected. As a last remark we observe that initially the decoupled scheme is slightly more accurate than the linearized scheme in the Crank-Nicolson case.



(a) Crank-Nicolson

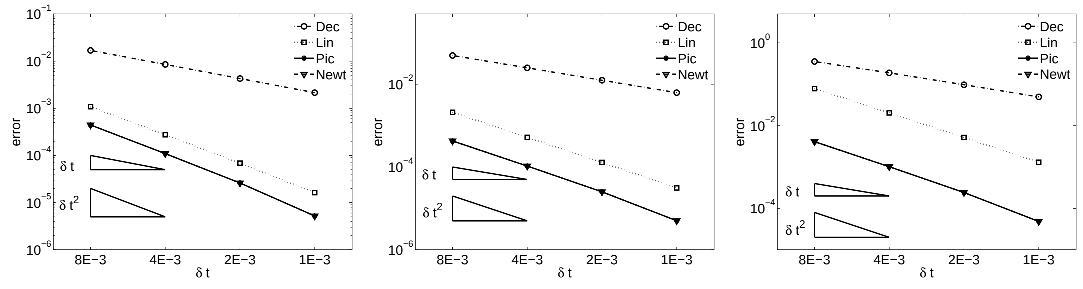
(b) Implicit Euler

Figure 5.13: Kinetic model $d_u = 1 = \chi$. Convergence of the temporal error. The four plots represent the different solver schemes, Dec, Lin, Pic and Newt. The time stepping was chosen $\delta t = 8E\text{-}3, 4E\text{-}3, 2E\text{-}3, 1E\text{-}3$.

The transient kinetic model – study of increasing chemosensitivity

In this paragraph we will provide the plots for the study of increasing chemosensitivity for all of our numerical schemes applied on the kinetic model. To study the effect of the chemosensitivity we use three distinct values $\chi = 10, 20, 50$. Our numerical reference solution is computed with $\delta t^* = 5E$-4. The spatial discretization is fixed for all simulations (including the reference solution), $\delta h^* = 1/16$. Figure 5.14 depicts the reduction of the temporal error for successively decreased δt for all iteration schemes. We observe a similar behavior of the error as we already pointed out for the aggregation model. Mainly three data sets can be distinguished, namely the poor performance of Dec, the quadratically converging error of Lin and the very similar error estimates of both nonlinear Richardson schemes, Pic and Newt. We notice that the difference between Lin and the two nonlinear Richardson schemes scale with the value of the chemosensitivity. In other words, as χ is increased, the gap between the two data sets of Lin and Pic or Newt also increases. For instance, for $\chi = 10$ the left-most plot in Figure 5.14 reveal that the two nonlinear Richardson schemes are approximately 2.5 times more accurate than Lin. For $\chi = 50$ this is significantly pronounced as the two nonlinear Richardson schemes are roughly 25 times more accurate than Lin. In contrast to this clear ten-fold increase of the gap, the situation changes remarkably when considering the difference between Dec and the two nonlinear Richardson schemes. Here, Pic and Newt are nearly $40-400$ times as accurate as Dec for $\chi = 10$, whereas this value only increases to $80-1000$ for $\chi = 50$. To put it in other words, there is roughly only a two-fold increase of the difference between Dec and Pic or Newt.

Hence, we conclude that an explicit linearization, e.g., of type Lin, of the underlying kinetic model is very sensitive to the parameter χ. Its performance, in terms of accuracy, decreases faster than the one obtained from nonlinear schemes. Note that Dec also introduces a non-trivial nonlinear iteration for the underlying kinetic model, since the growth term in the u equation gives rise to a nonlinearity (in u).



(a) $\chi = 10$ (b) $\chi = 20$ (c) $\chi = 50$

Figure 5.14: Kinetic model. Convergence of the temporal error for $\chi = 10$, $\chi = 20$ and $\chi = 50$. The four plots represent the different solver schemes, Dec, Lin, Pic and Newt. The time stepping was chosen $\delta t = 8E\text{-}3, 4E\text{-}3, 2E\text{-}3, 1E\text{-}3$.

The transient kinetic model – basic efficiency analysis

We start by comparing the efficiency of the two nonlinear Richardson schemes Newt and Pic. In order to compute $\text{EFF}_{\text{Newt}}/\text{EFF}_{\text{Pic}}$ it suffices to consider the ratio of required number of average iterations for the two schemes. Table 5.12 provides these numbers for all monolithic schemes. In the case of $\chi = 10$ this ratio reaches from 1.5 to 1.8, i.e., Pic requires up to 1.8 times as many iterations as Newt. For larger values of χ this ratio only varies subtly, $1.3-1.9$ and $1.5-2.3$ for $\chi = 20$ and $\chi = 50$, respectively. All corresponding data can be taken from Table 5.13.

When considering $\text{EFF}_{\text{Pic}}/\text{EFF}_{\text{Lin}}$, the effect of increased chemosensitivity is substantially recognizable. For a moderate choice of $\chi = 10$ the ratio of efficiency lies in between $\text{EFF}_{\text{Lin}}/\text{EFF}_{\text{Pic}} = 1.0$ and $\text{EFF}_{\text{Lin}}/\text{EFF}_{\text{Pic}} = 1.6$. For larger values of χ this ratio increases remarkably. We calculate the ranges $\text{EFF}_{\text{Lin}}/\text{EFF}_{\text{Pic}} = 2.0-2.9$ and $\text{EFF}_{\text{Lin}}/\text{EFF}_{\text{Pic}} = 5.5-12.3$ for $\chi = 20$ and $\chi = 50$, respectively. Interestingly enough, the ratio of efficiency seems to be proportional to the value of χ, namely a two- or five-fold increase of χ approximately results in a similar

	Lin			Pic			Newt		
$\delta t \backslash \chi$	10	20	50	10	20	50	10	20	50
$1E$-3	4.5	4.4	5.3	8.7	9.5	11.6	5.8	7.5	7.6
$2E$-3	4.5	4.5	5.4	8.9	11.2	13.5	6.0	5.8	8.1
$4E$-3	4.6	4.6	5.4	11.6	11.6	15.8	6.3	6.0	9.0
$8E$-3	4.8	5.0	5.6	11.6	12.0	19.5	6.8	6.8	8.4

Table 5.12: Kinetic Model. Averaged total number of linear iterations for all monolithic schemes.

	$\text{EFF}_{\text{Newt}}/\text{EFF}_{\text{Pic}}$			$\text{EFF}_{\text{Pic}}/\text{EFF}_{\text{Lin}}$			$\text{EFF}_{\text{Newt}}/\text{EFF}_{\text{Lin}}$		
$\delta t \backslash \chi$	10	20	50	10	20	50	10	20	50
$1E$-3	1.5	1.3	1.5	1.6	2.9	12.3	2.4	3.8	18.5
$2E$-3	1.5	1.9	1.7	1.3	2.1	8.5	2.0	4.0	14.5
$4E$-3	1.8	1.9	1.8	1.0	2.0	6.8	1.8	3.8	12.2
$8E$-3	1.7	1.8	2.3	1.0	2.0	5.5	1.7	3.6	12.7

Table 5.13: Kinetic Model. Ratios of efficiency $\text{EFF}_{\text{Newt}}/\text{EFF}_{\text{Pic}}$, $\text{EFF}_{\text{Pic}}/\text{EFF}_{\text{Lin}}$ and $\text{EFF}_{\text{Newt}}/\text{EFF}_{\text{Lin}}$ for all configurations.

increase of $\text{EFF}_{\text{Lin}}/\text{EFF}_{\text{Pic}}$. This was not observed for the ratio $\text{EFF}_{\text{Pic}}/\text{EFF}_{\text{Newt}}$ or other models.

The ratio of efficiency for Lin and Newt can be inferred by the transitive relation as before. For $\chi = 10, 20, 50$ we obtain values in the range of $\text{EFF}_{\text{Lin}}/\text{EFF}_{\text{Newt}} = 1.7 - 2.4$, $\text{EFF}_{\text{Lin}}/\text{EFF}_{\text{Newt}} = 3.6 - 4.0$ and $\text{EFF}_{\text{Lin}}/\text{EFF}_{\text{Newt}} = 12.2 - 18.5$, respectively.

To summarize our findings for the kinetic model, we can suggest that the nonlinear Richardson schemes are the most favorable schemes throughout all of our choices for χ. Most remarkably for large values of χ and a small time stepping either Pic or Newt are recommended. Note that we only studied the effect of scaling the chemosensitivity in the model in order to enhance the nonlinearity. The growth term in the cell equation of (4.1.4) might also play a vital role in determining the strength of the nonlinearity and hence should also be taken into account when considering the efficiency. Particularly this might lead to a more detailed distinction of the two similar appealing nonlinear Richardson schemes.

5.4.3. Conclusion of the numerical comparisons

To conclude this section, let us recapitulate the most common observations. Firstly, we recognize the consistency of all our iteration schemes and underlying models in terms of spatial and temporal convergence behavior. Our numerical findings meet our a priori expectations. For all schemes a quadratical spatial convergence is observed and a second-order temporal discretization drove all monolithic schemes to a corresponding second-order convergence in time. Because the strongly decoupled scheme does not incorporate a fully nonlinear treatment of the v equation, Dec could only be driven to first order accuracy in time.

Secondly, we observed that the ratio of efficiency among the monolithic schemes scales with the chemosensitivity. In other words, an increase of χ led to a correspondingly related increase of the ratio of efficiency. For the strongly decoupled scheme, we documented this type of scaling only for the accuracy. This was reasonable since a thorough comparison of the complexity of the underlying iterations for a singlegrid solver applied on a decoupled system on the one hand, and for a monolithic multigrid solver on the other hand, are highly intricate and would have overwhelmed the scope of this section.

A bit of caution is advised when presenting a recommendation for a specific numerical scheme at this current stage of numerical analysis. At the one hand, accuracy concerns clearly favor monolithic schemes, since DEC provides errors that are significantly larger (up to several powers of 10) than the ones for the monolithic schemes. Among the monolithic schemes, the both nonlinear Richardson schemes provide the most efficient results, particularly for large chemosensitivities.

On the other hand, the simple implementation and complexity of DEC is a feasible argument for its application in the context of rather moderate nonlinear chemotaxis models. Moreover, in our numerical study we did not take into account the (sometimes) costly task of constructing a reasonable Jacobian for NEWT. Last but certainly not least, we remark that the requirement of a stabilization method crucially influence our recommendations. Referring to Section 4.5.4, we remark that the AFC stabilizing algorithm cannot be efficiently implemented for NEWT in a straightforward manner. Therefore, in terms of efficiency, we cannot recommend the AFC stabilized Newton's method at its current stage of development. With this concern in mind, we tend to favor PIC if the initial guess for the nonlinear iteration can be chosen appropriately and χ is rather large.

5.5. Limitations of the iteration schemes

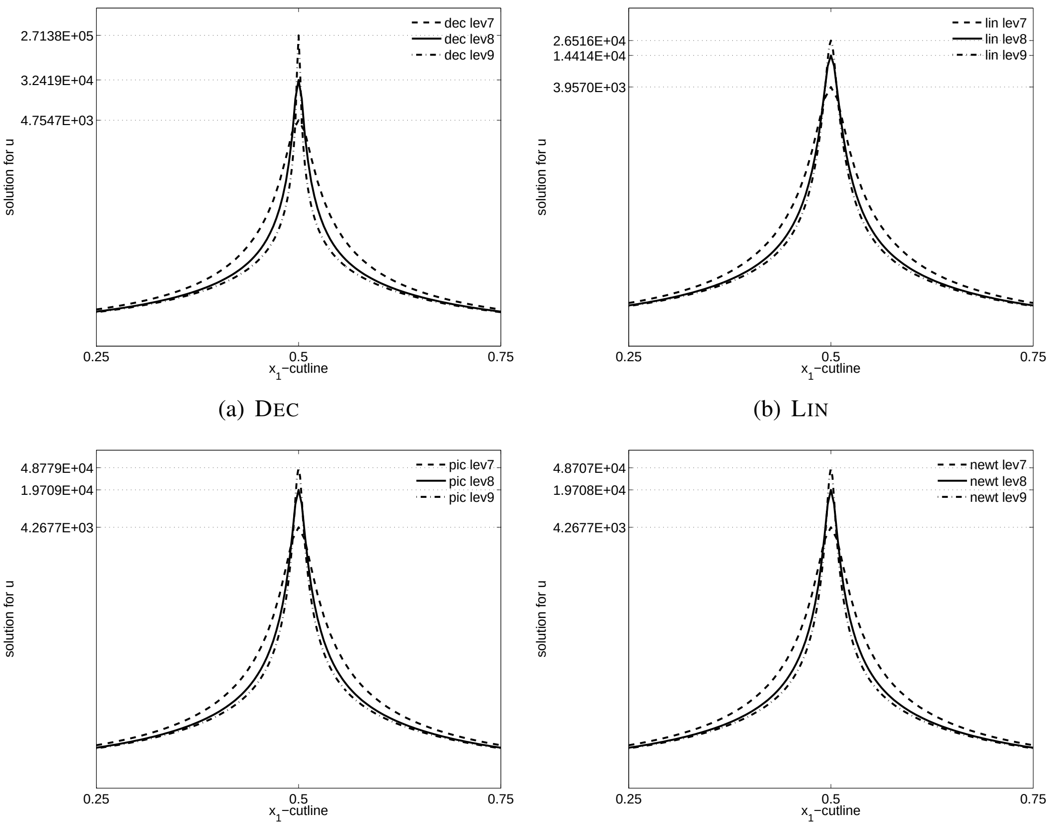
After the previous basic convergence and efficiency analysis we turn now to the limitations of the four iteration schemes, which one encounters when applying these schemes to non-academic parameter settings. Our designated testing models reveal characteristic numerical challenges, which should be captured by reliable iteration schemes.

The following observation are mostly of qualitative nature and will provide more detailed insights into the robustness and applicability of our four iteration schemes.

5.5.1. About misleading solutions

We start our investigation with the transient version of the minimal model (4.1.2). In contrast to the convergence study above, we will now focus on a parameter setup that lead to a blowing-up solution. In terms of the initial condition (5.4.2) this implies the choice $\chi = 10$ and we can expect the blow-up to occur in the center of the computational domain QUAD. In order to validate this conjecture, we apply all iteration schemes with an appropriate spatial and temporal discretization. We follow the definition of the numerical blow-up time as already practiced by Chertock and Kurganov [16]. Therein the authors identified the instant of time t^* where the maximal value of the solution (in the center) keeps increasing remarkably (does not converge) when the spatial mesh is refined, whereas the solution converges at time instants $t < t^*$. In this case, Chertock and Kurganov proposed t^* as a first reasonable estimation for the blow-up time.

In the following cutplanes in Figure 5.15 we demonstrate how the maximum of the peaky solution at the designated time instant grows as the spatial mesh is refined for all iteration schemes. This validates that our simulation setup generates a solution that blows up in finite time $t^* < 0.348$.



(a) DEC

(b) LIN

(c) PIC

(d) NEWT

Figure 5.15: Minimal Model. Capturing of the blowing-up solution for all iteration schemes when the spatial mesh is refined. The time stepping was chosen to be $\delta t = 1E\text{-}6$ for DEC and $\delta t = 1E\text{-}4$ for all monolithic schemes. All solutions are shown at simulation time $t = 0.348$.

Remarkably, the required uniform time stepping δt that is required to obtain our results differs for the underlying iteration scheme. For a coarser temporal discretization, certain schemes produce misleading results or even diverging solutions in terms of a crash-down of the solver. Let us have a closer look on this behavior.

Firstly, the decoupled scheme DEC seems to be very sensitive to the time stepping, cf. Figure 5.16. When running simulations with the larger time stepping, e.g., $\delta t = 1E\text{-}2$, the solution is misleading, as we would accept a blow-up time somewhere before $t^* < 0.04$ (left column in Figure 5.16). Moreover the underlying solver breaks down very early (after seven time steps). In other words, DEC is not as robust as the nonlinear Richardson schemes, i.e., the time stepping must be chosen carefully.

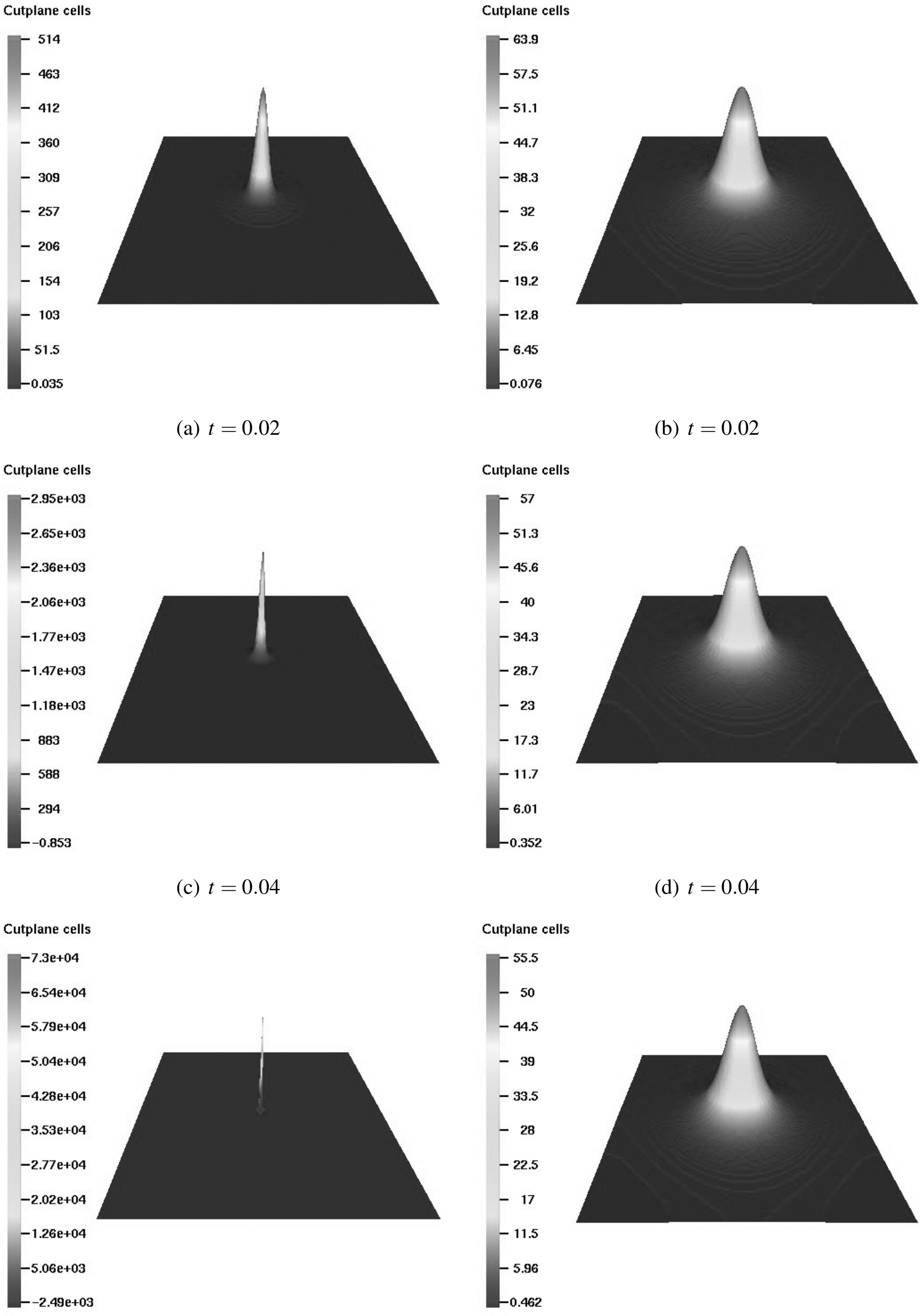
Secondly, the linearized scheme also reveals significant drawbacks. If the time stepping is chosen insufficiently small, e.g., $\delta t = 1E\text{-}2$, the solution differs dramatically from our expectations. Figure 5.17 depicts the solutions obtained by LIN and NEWT for comparison reasons. We observe that the linearized solution tends to be too diffusive and does not blow up, in contrast to the solution obtained from Newton's scheme. Indeed, ongoing simulations (not shown) revealed that LIN finally leads to the homogeneous steady state $u^* = ||u_0||_{L^1} = \pi$.

Hence, both of the schemes, DEC and LIN, are not as robust as the nonlinear Richardson schemes. From the applicant's point of view the wrong results of the linearized scheme are often more crucial, since they pretend a totally different solution, while a solver break-down of the decoupled scheme serves as a decent indicator for an erroneous simulation. This latter case can be more easily recognized by the applicant.

One possible explanation of this odd behavior is based on the fact that the nonlinear coupling between u and v cannot be accurately treated by the decoupled and linearized approach. Indeed, while LIN explicitly linearizes the nonlinear coupling, DEC breaks the two-way coupling by the approach of strong decoupling, which also leads to linearization for the minimal model of chemotaxis. Hence in both cases the treatment of the nonlinearity leads to significant inaccuracies, in contrast to the nonlinear Richardson schemes which integrate a true nonlinear iteration (for the minimal model of chemotaxis). In the light of the rather mild nonlinearity of the minimal model, caused by the chemotaxis term $\nabla \cdot (u\chi\nabla c)$, this behavior is particularly interesting and we finally conjecture that the coupling via the positive feedback in our model,

$$\text{agglomeration of } u \quad \rightarrow \quad \text{increase of } v \quad \rightarrow \quad \text{even stronger agglomeration of } u\,,$$

is the more crucial reason for the poor numerical approximations of DEC and LIN.



(a) $t = 0.02$

(b) $t = 0.02$

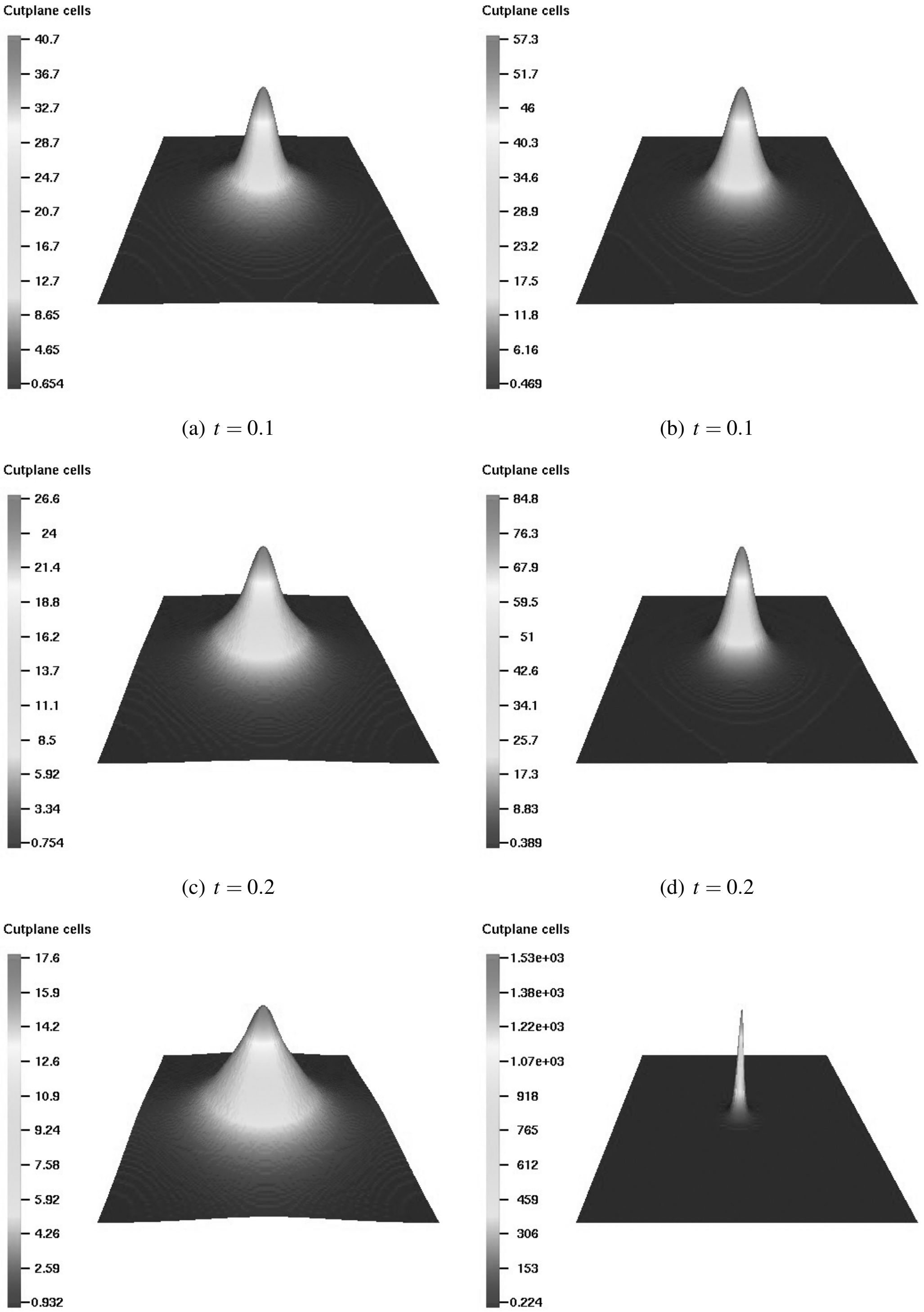
(c) $t = 0.04$

(d) $t = 0.04$

(e) $t = 0.06$

(f) $t = 0.06$

Figure 5.16: Minimal Model. The solution obtained from DEC (left column) and NEWT (right column) at certain time instants is shown. Recognize the negative values appearing for DEC. All simulations were run with $\delta h = 1/128$ and $\delta t = 1E\text{-}2$.



(a) $t = 0.1$

(b) $t = 0.1$

(c) $t = 0.2$

(d) $t = 0.2$

(e) $t = 0.3$

(f) $t = 0.3$

Figure 5.17: Minimal Model. The solution obtained from LIN and NEWT at certain time instants is shown. While LIN reveals an illusive global in time solution (left column), NEWT still captures the blow-up reliably (right column). All simulations were run with $\delta h = 1/128$ and $\delta t = 1E\text{-}2$.

5.5.2. About the nonlinearity

In this paragraph we will examine the strength of the nonlinearity of the transient aggregation and kinetic model of chemotaxis, cf. (4.1.3) and (4.1.4). In the context of nonlinearities these models are of particular interest since their order of nonlinearity give rise to more complex dynamics and numerical challenges than the minimal model of chemotaxis (4.1.2). In the course of this examination we will compare the both nonlinear Richardson schemes in terms of required nonlinear iterations. It is well known that the nonlinear Newton iterations locally converge quadratically whereas the Picard iterations only converge linearly. However, the costly assembly of the corresponding Jacobian matrices in each nonlinear iteration has to be taken into account when comparing the overall performance/efficiency of both schemes. There exist several alternatives to the nonlinear schemes presented in this work, see, e.g., [22] for competitive Newton schemes, and we will discuss some promising approaches at the end of this chapter. Nevertheless, for a first basic analysis, the upcoming results provide characteristic properties of the two most commonly used nonlinear Richardson schemes and uncover possible improvements for future work.

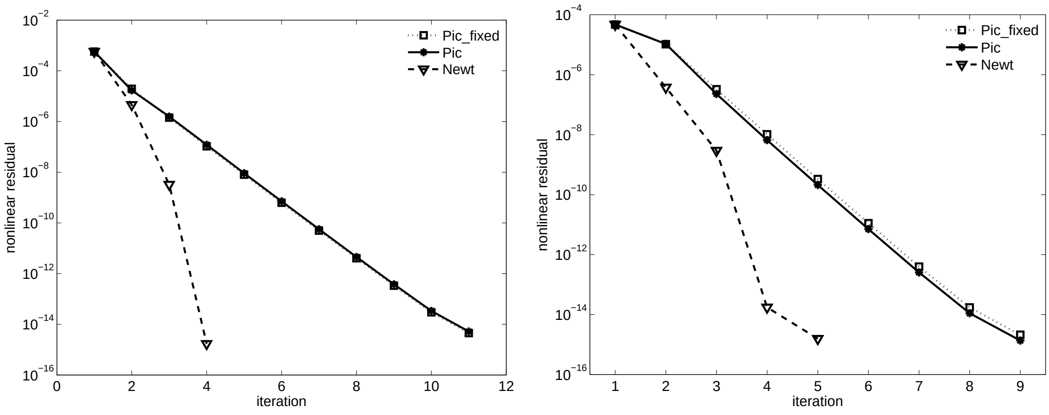
For a first comparison we consider the transient aggregation and kinetic model, (4.1.3) and (4.1.4), with the usual boundary and initial conditions, cf. (2.2.3), (5.6.1) and (5.4.4), on the default computational domain QUAD16. We discretize the domain with a uniform mesh size $\delta h = 1/8$ and use the default model parameters for both models:

Kinetic model: $d_u = 0.0625, \chi = 8.5$ and $\beta = 32$;

Aggregation model: $d_u = 1, \chi = 80$ and $d_v = 0.33$.

Figure 5.18 depicts the nonlinear residual drop of Newton's method and of the Picard iteration exemplary for the first time step with $\delta t = 0.01$. Concerning Picard's iteration we distinguish an inner termination criterion of kind (5.2.3) (as it is also used for Newton's method) and a criterion with a fixed relative threshold of $1E$-10. We observe that this differentiation is rather subtle and hence we employ the criterion (5.2.3) for NEWT and PIC for all numerical simulations. Furthermore we observe the linear convergence of Picard's iteration and the quadratical residual drop of Newton's method. Note that the last nonlinear iteration of Newton's method applied on the kinetic model does not drop the residual quadratically since the absolute stopping criterion for the nonlinear iterations, $\varepsilon_a = 1E$-14, can already be reached by terminating the corresponding inner multigrid solver after very few iterations. Indeed, while the third nonlinear residual drop (from $2.9553E$-9 to $1.7426E$-14) requires five linear multigrid iterations, the last (from $1.7426E$-14 to $1.5686E$-15) only requires one single multigrid sweep.

To examine the difference in the residual drop in more detail, we continue our study with different temporal discretizations. In Figure 5.19 we plot the average number of exerted nonlinear iterations for PIC and NEWT for increasing values of δt. We clearly observe that the average number of nonlinear iterations remain almost constant for Newton's method, whereas the number remarkably increases for Picard's linearization, for that it requires up to 15 and 5 times the number of iterations of Newton's method for the aggregation and kinetic model, respectively. This additional computational expense cannot be balanced by the extra assembly costs of Newton's Jacobian. In fact, the simulations revealed that PIC requires up to 11 and 4 times more CPU time than NEWT for the aggregation and kinetic model, respectively (not shown). Hence, for larger time steps Newton's method clearly pays off. Furthermore, these numbers support the idea that the aggregation model seems to be more sensitive to the accuracy of the nonlinear treatment than the kinetic model. This meets our introductory conjecture in Section 4.1 about the increased nonlinear index of that model, i.e., in the aggregation model both equations (u and v) contribute to the

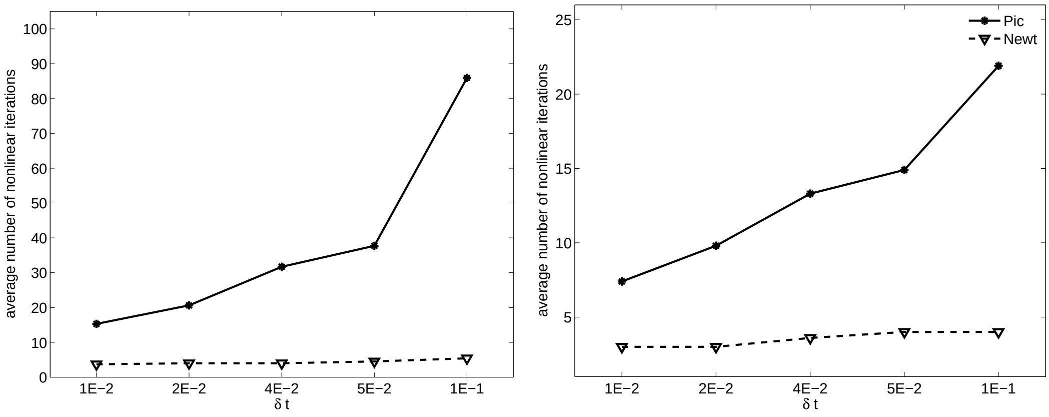


(a) Aggregation model

(b) Kinetic model

Figure 5.18: Nonlinear residual drop obtained from Pɪᴄ and Nᴇᴡᴛ. Additionally, the results for a fixed relative stopping criterion for Pɪᴄ are displayed, Pɪᴄ fixed. The data is collected from the first time step with $\delta t = 0.01$.

nonlinearity. However, these findings should be interpreted with caution. The δt-threshold beyond which Newton's method pays off, can be unpractical in particular cases, because the temporal error introduced by excessive large time stepping can significantly deteriorate the overall accuracy of the scheme.



(a) Aggregation model

(b) Kinetic model

Figure 5.19: Nonlinear residual drop obtained from Pɪᴄ and Nᴇᴡᴛ for increasing time step widths, $\delta t \in [0.01, 0.1]$.

5.6. Stabilization via AFC

The stable numerical treatment of chemotaxis-dominated PDEs, namely a dominant contributions of the chemotaxis term in the overall system dynamics in terms of advection, is a challenging task. This is particularly crucial if local agglomeration of cells leads to steep gradients of the chemical substance, e.g., in the case of blowing-up solutions or traveling fronts. As already mentioned in the numerical state-of-the-art in Section 5.1, plenty of approaches of stabilizing nature have been introduced by many researchers. The arising trade-off between preserving physical properties and a certain accuracy of the solution is the crucial point for stabilizing algorithms. In the context of FEM, yet, only first-order accurate stabilization schemes have been introduced (cf. Section 5.1).

This section is concerning about the applicability of stabilization via AFC, which has been introduced in Section 4.5. One of its key-features is its convergence of practically mixed-order, hence, providing a competitive novel approach in stabilizing chemotaxis-dominated PDEs.

In the following paragraphs we will present qualitative analysis of AFC stabilized numerical schemes applied on the three exemplary transient chemotaxis models which have already been used before. The author would like to remark that partial results can already been found in previously published work by Strehl *et al.*, cf. [96, 98]. Since the models have been considered multiple times in the course of this chapter, we kindly refer the reader to the corresponding equations for details.

5.6.1. AFC for the transient minimal model of chemotaxis

The reader is referred to (4.1.2) for model details. Since this model gives rise to blowing-up solutions, i.e., immensely steep gradients of solutions of nearly vanishing support, it deals as a valuable yardstick for stabilization schemes. The upcoming numerical simulations are performed on the unit square QUAD1. The minimal model will be complemented by initial conditions that theoretically lead to a blowing-up solution. Correspondingly to, e.g., Chertock and Kurganov [16] we focus on

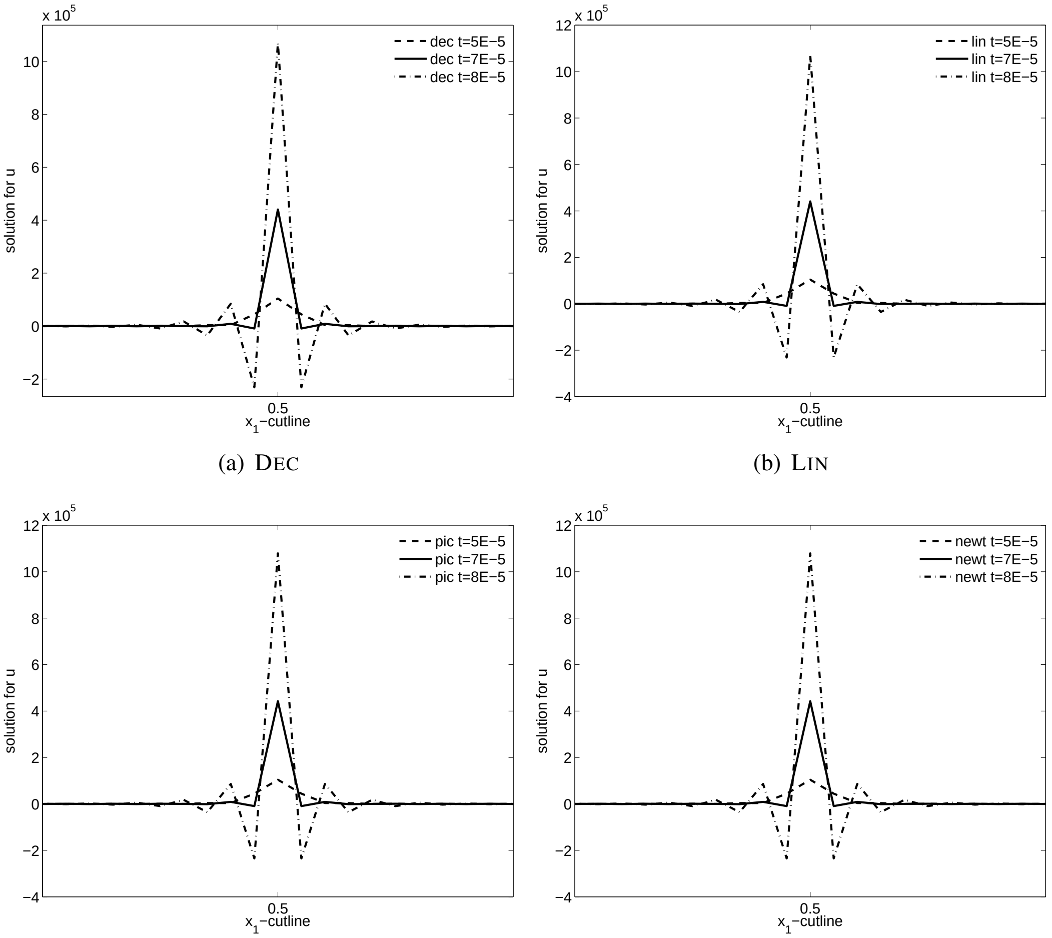
$$\begin{aligned} u_0(\mathbf{x}) &= 1000\, e^{-100\,[(x_1-0.5)^2+(x_2-0.5)^2]}\,, \\ v_0(\mathbf{x}) &= 500\, e^{-50\,[(x_1-0.5)^2+(x_2-0.5)^2]}\,. \end{aligned}$$

Since $||u_0||_{L^1} > 8\pi$ we expect a blow-up in finite time even for $\chi = 1$. This blow-up will be located in the center of the domain, likewise to the initial condition. In contrast to the initial conditions accompanying the minimal model before, cf. (5.4.2), the choice of initially non-vanishing chemoattractant concentration accelerates the blow-up and is only due to computational reasons.

In the upcoming numerical studies we qualitatively present (i) unstable transient solutions provided by the high-order numerical scheme, i.e., without the application of AFC (ii) the low-order counterpart via discrete upwinding, introduced as an intermediate step of the AFC paradigm and finally (iii) the AFC stabilized solution. Once again, we like to mention that quantitative results of the AFC methodology are not the scope of this current study but definitely give rise to future research since they have not been considered in the literature so far.

AFC for the transient minimal model of chemotaxis – high-order solutions

The first task is to reveal that all iteration schemes often provide only meaningless high-order solutions, because of severe numerical instabilities. To this end we choose $\delta t = 1E\text{-}6$ and observe the development of the solution. Figure 5.20 documents the appearance of negative values that finally lead to the formation of spurious oscillations for all four iteration schemes, from top to bottom, DEC, LIN, PIC and NEWT.



(a) DEC

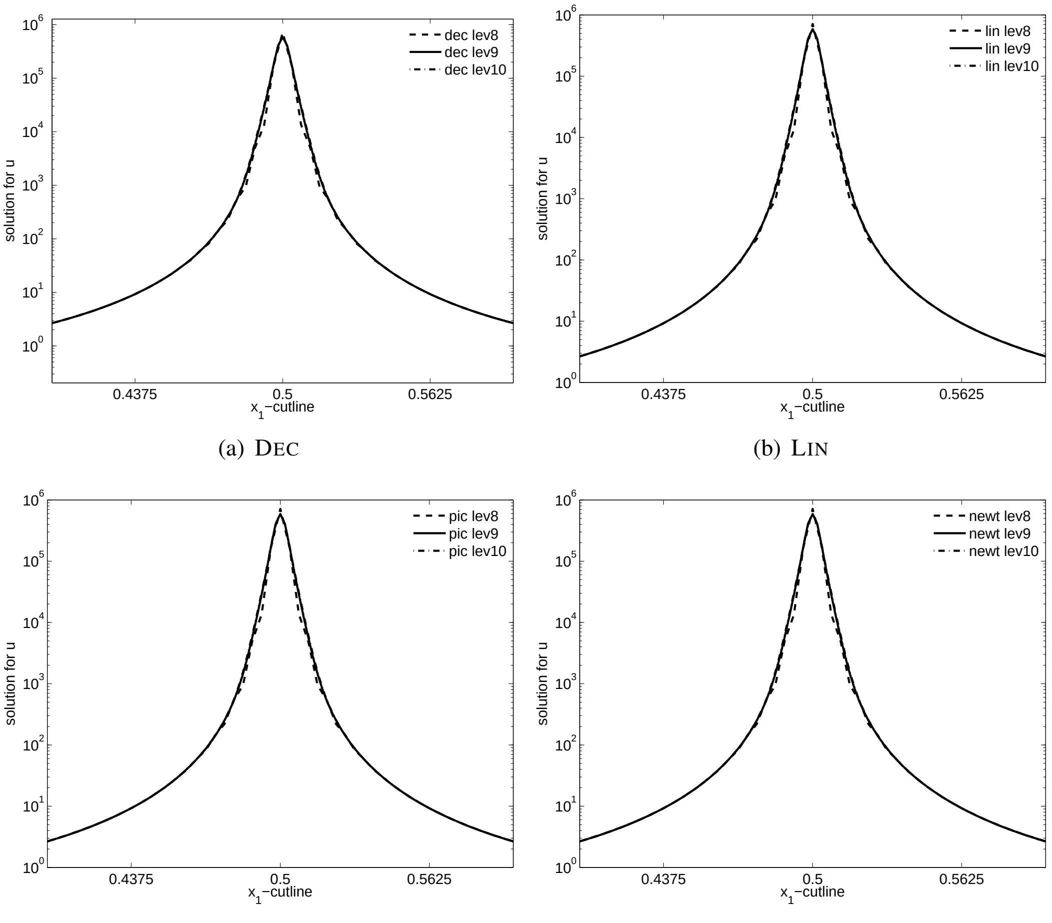
(b) LIN

(c) PIC

(d) NEWT

Figure 5.20: Minimal model. Documentation of numerical instabilities for all four high-order iteration schemes at different instances of time, namely $t = 5E\text{-}5$, $t = 7E\text{-}5$ and $t = 8E\text{-}5$. Discretization parameters were chosen as $\delta h = 1/128, \delta t = 1E\text{-}6$.

These artifacts are of pure numerical character as our findings in Figure 5.21 at successive spatial refinements ($\delta h < 1/128$) suggest that the solution exists (and particularly does not blow up) at the given time instant. Therefore our results demonstrate that the four high-order schemes are very sensitive in terms of the discretization parameters and cannot be considered robust. Furthermore, we recognize that all four schemes provide very similar results, i.e., qualitatively we cannot distinguish their solutions for the current parameter setting.



(a) DEC

(b) LIN

(c) PIC

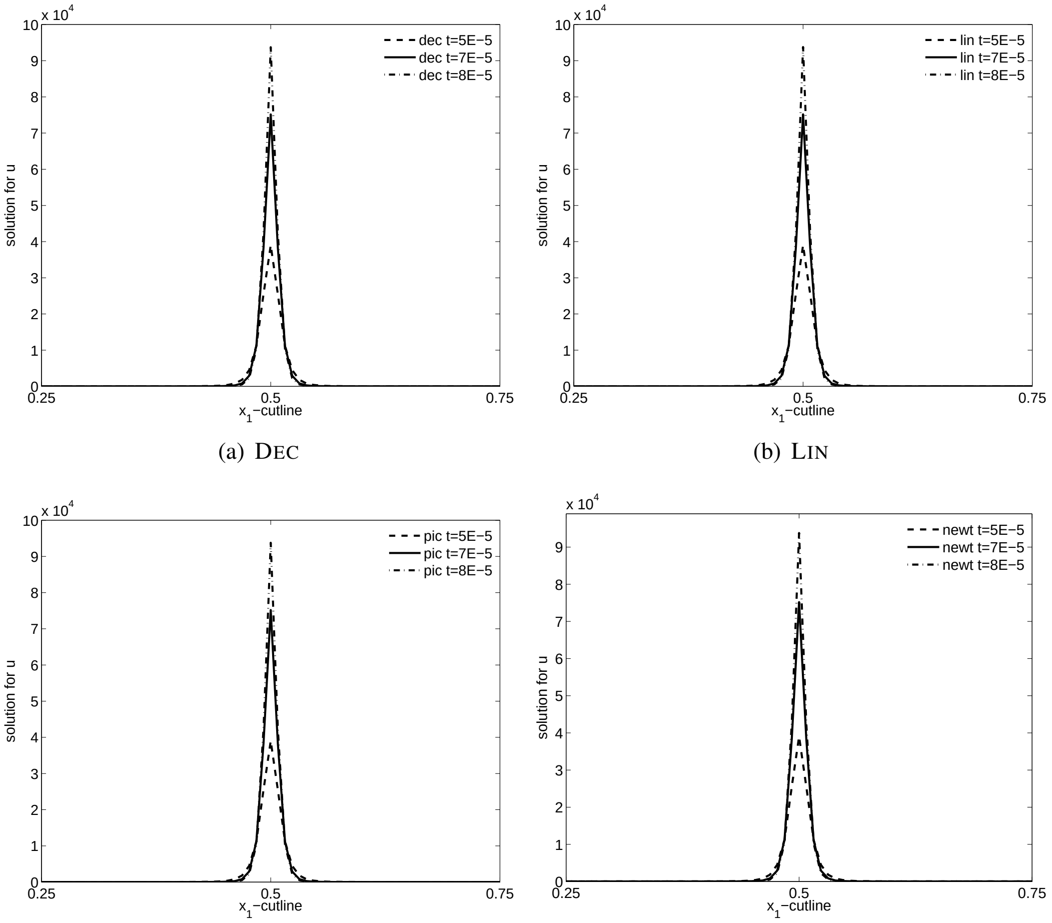
(d) NEWT

Figure 5.21: Minimal model. Documentation of the existence of the solution at $t = 8E\text{-}5$ for all four high-order iteration schemes at successively decreased spatial mesh sizes, $\delta h = 1/256$, $\delta h = 1/512$ and $\delta h = 1/1024$. Time stepping was chosen as $\delta t = 1E\text{-}6$.

AFC for the transient minimal model of chemotaxis – low-order solutions

In what follows we present the solutions for the low-order counterparts of our underlying four iteration schemes. As we already discussed in Section 4.5.2 we expect the schemes to remedy the numerical artifacts, such as negative solution values. For reasons of comprehensibility, Figure 5.22 displays the corresponding solutions at the same time instances as Figure 5.20. We observe that, accordingly to our expectations, all solutions yield positivity and no oscillations can be identified.

The stabilizing character of the discrete upwinding method allows for a more detailed analysis of the blowing-up solution. Indeed, it is computationally a very hard task to estimate the blowing-up time for the high-order scheme, simply because the numerical pollution would require impractical fine discretization levels to study the spatial convergence/divergence of the solution as practiced before, [16]. The low-order solutions, however, are not polluted by instabilities and hence, the analysis for the blow-up time can be conducted on reasonable discretization levels as before. Following our results documented in Figure 5.23 and the definition of the numerical blow-up in [16], we conjecture that the blow-up happens at a time $t^* \in (8E\text{-}5, 1E\text{-}4]$, since we know that the numerical solution converges at $t = 8E\text{-}5$ (cf. Figure 5.21) but does not so at $t = 1E\text{-}4$ (cf. Figure 5.23).



(a) DEC

(b) LIN

(c) PIC

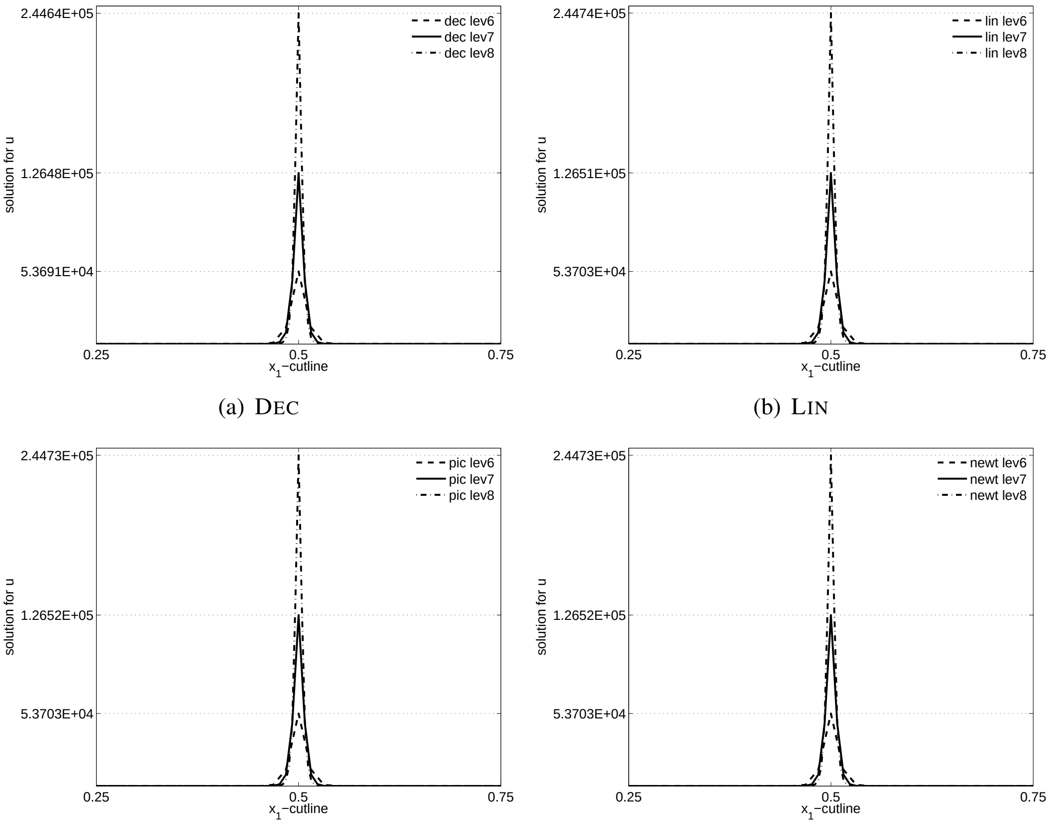
(d) NEWT

Figure 5.22: Minimal model. Plots of the low-order counterparts of the four iteration schemes at different instances of time, namely $t = 5E\text{-}5$, $t = 7E\text{-}5$ and $t = 8E\text{-}5$. Discretization parameters were chosen as before, $\delta h = 1/128, \delta t = 1E\text{-}6$.

AFC for the transient minimal model of chemotaxis – AFC solutions

In this paragraph, the final step of eliminating overdiffusive fluxes from the solutions is presented. Because of those artificially introduced diffusive contributions, the low-order solutions, as depicted above, cannot shape the singular behavior well enough. Indeed, when comparing the maxima of the low-order solutions for $\delta h = 1/512$ at $t = 1E\text{-}4$ (Figure 5.23) with the corresponding maxima of the high-order solutions for the same discretization level but at $t = 8E\text{-}5$ (Figure 5.21), we recognize that the former are less pronounced. That is a contradiction to the fact that at the blow-up time all mass is expected to be agglomerated at one single point (delta-singularity), which implies that we expect our solution to attain its maximum (obeying mass conservation) at the time of the blow-up. One reasonable explanation of this discrepancy is the additional diffusion which tries to smoothen the formation of such a singularity, resulting in a smaller maximum value.

The results given in Figure 5.24 and 5.25 confirm our conjectures. Firstly, we observe that AFC is capable of preserving positivity of the solution. Moreover, no severe oscillations appear,



(a) DEC

(b) LIN

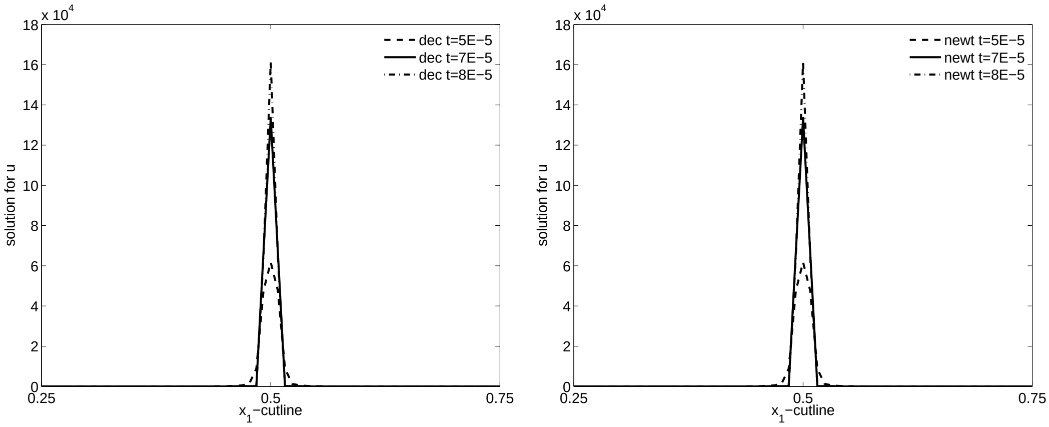
(c) PIC

(d) NEWT

Figure 5.23: Minimal model. Increasing low-order solution at $t = 1E\text{-}4$ for all four iteration schemes at successively decreased spatial mesh sizes, $\delta h = 1/64$, $\delta h = 1/128$ and $\delta h = 1/256$. This affirms the conjecture a blowing-up time of $t^* \in (8E\text{-}5, 1E\text{-}4]$. Time stepping was chosen as $\delta t = 1E\text{-}6$.

Figure 5.24. Since all four iteration schemes provide similar results, in Figure 5.24 we only plot the solution for two of the four schemes. As in the case of the discrete upwinded schemes, AFC allows for capturing the blowing-up solutions properly. Figure 5.25 shows the solutions obtained from the AFC application for all four iteration schemes. As the spatial mesh is successively refined we observe a significant increase of the solutions' maxima which affirms the conjecture of the blowing-up time as $t^* \in (8E\text{-}5, 1E\text{-}4]$.

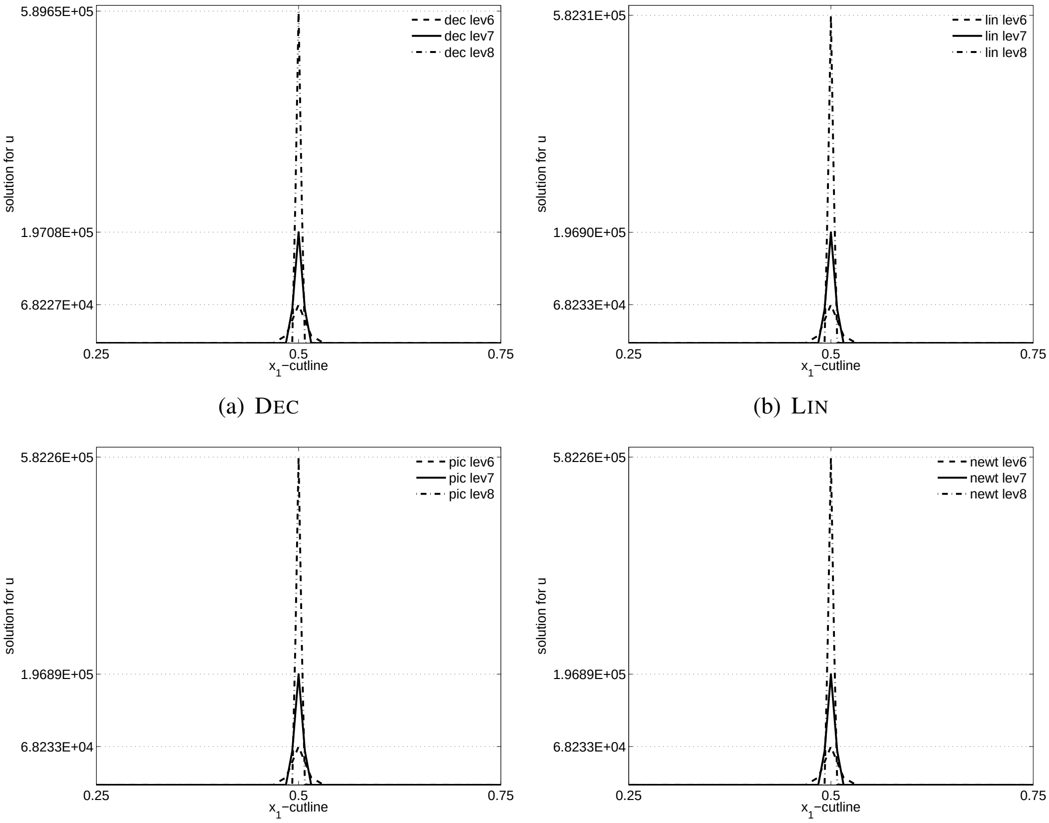
Remarkably, the maxima of the AFC solutions differ from the low-order solutions, compare the corresponding assays in Figure 5.23 and Figure 5.25. AFC provides solutions with larger maxima than their low-order counterparts, which agrees well with the overdiffusive nature of the upwinded schemes and the concept of AFC.



(a) DEC

(b) NEWT

Figure 5.24: Minimal model. Exemplary plots of the AFC counterparts of DEC and NEWT at different instances of time, namely $t = 5E$-5, $t = 7E$-5 and $t = 8E$-5. Discretization parameters were chosen as before, $\delta h = 1/128, \delta t = 1E$-6.



(a) DEC

(b) LIN

(c) PIC

(d) NEWT

Figure 5.25: Minimal model. With the application of AFC, increasing solution at $t = 1E$-4 for all four iteration schemes at successively decreased spatial mesh sizes, $\delta h = 1/64$, $\delta h = 1/128$ and $\delta h = 1/256$. This affirms the conjecture a blowing-up time of $t^* \in (8E\text{-}5, 1E\text{-}4]$. Time stepping was chosen as $\delta t = 1E$-6.

AFC for the transient minimal model of chemotaxis – an exemplary 3D case

After presenting the AFC stabilized results for the minimal model in a classical setting, i.e., blow-up in a 2D square, we will present now selected results from a 3D application that are taken from [98]. The purpose of this demonstration is to promote the flexibility of AFC in terms of dimension and underlying computational domain.

The upcoming simulations are subject to a discretization of $\delta t = 1E\text{-}4$ and a total of 147,456 conforming trilinear finite elements. We used the decoupled iteration scheme DEC and a slightly different limiting strategy for the AFC application, see [98] for details.

We simulated the minimal model of chemotaxis (4.1.2) on the computational domain CYL3D with standard Neumann boundary conditions (2.2.3). Initially we prescribe well separated cell and chemoattractant distributions, see Figure 5.26,

$$\begin{aligned} u_0(\mathbf{x}) &= 1000\,e^{-100\,[x_1^2+x_2^2+(x_3-2)^2]}\,, \\ v_0(\mathbf{x}) &= 500\,e^{-50\,[x_1^2+x_2^2+(x_3-3)^2]}\,. \end{aligned}$$

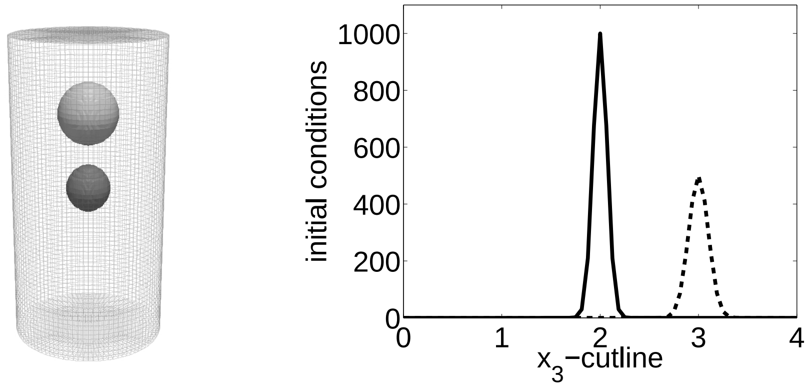
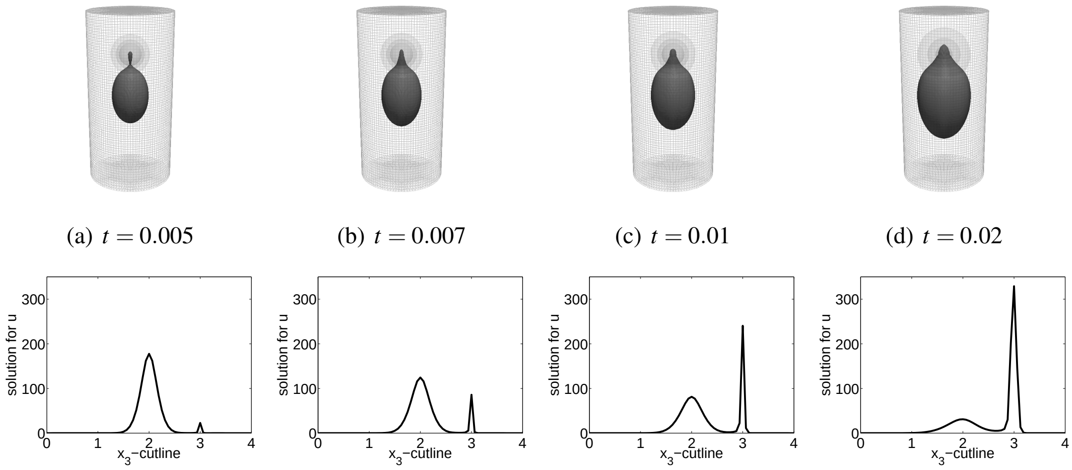


Figure 5.26: Minimal model. Initial condition for the 3D simulation. **Left:** Initial distribution for the cells (centered dark spot) and the chemoattractant (upper bright spot). **Right:** Cutlines of the cell distribution (solid line) and chemoattractant concentration (dashed line) along the x_3-axis.

Our simulations in Figure 5.27 reveal that the cells tend to chemotax to the origin of the chemoattractant and blow up. Here we only show the results for $\chi = 1$ and $d_u = 1$. The top row of the figure displays the 3D distribution of the cells (dark) and the chemoattractant (bright) as isovolumes. The bottom row depicts cutlines along the x_3-axis of the cell distribution. We clearly recognize the smooth solution profile without any appearance of severe oscillations or negative values. In contrast to that, the pure high-order Galerkin discretization provides poor numerical solutions, namely the solution profile is polluted by frequent oscillations and negative values, see [98] for details.



(a) $t = 0.005$ (b) $t = 0.007$ (c) $t = 0.01$ (d) $t = 0.02$

(e) $t = 0.005$ (f) $t = 0.007$ (g) $t = 0.01$ (h) $t = 0.02$

Figure 5.27: Minimal model. Development of cell and chemoattractant concentrations for $\chi = 1, d_u = 1$ at $t = 0.005, 0.007, 0.01, 0.02$; $\delta t = 1E$-4. **Top:** Distribution of cells u (dark volumes) and chemoattractant v (bright volumes). **Bottom:** Cutline along the x_3-axis for cells u.

5.6.2. AFC for the transient aggregation model

The motivation behind the aggregation model (4.1.3) was to introduce certain saturation coefficients that render the solution to exist globally in time, particularly, no blow-up can be observed. However, numerically speaking, this model remains challenging because of it nonlinearity and the usual chemotaxis-favoring parameters, e.g., $\chi \gg 1$ cf. [16].

In what follows we firstly consider the high-order solutions of our four iteration schemes and reveal their numerical artifacts that accumulate and eventually lead to divergence of the underlying solver. Then we will see how AFC helps to stabilize the solutions and allows us to give qualitative conclusions. For the upcoming numerical simulations the aggregation model (4.1.3) is accompanied by a random distributed initial concentration of u and initial absence of the chemoattractant v,

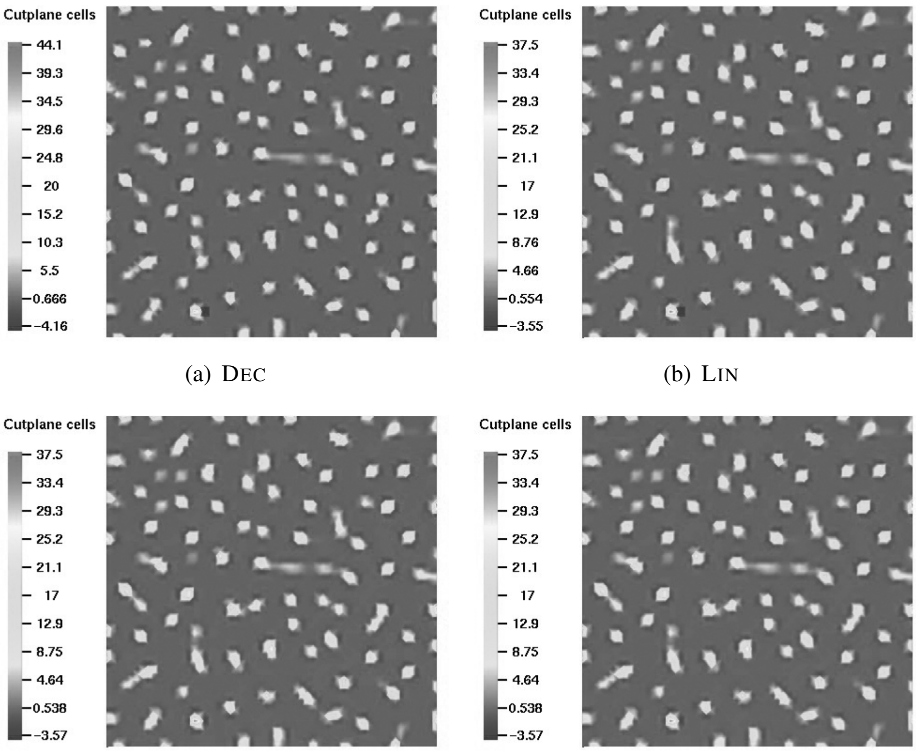
$$\begin{cases} u_0 &= 0.9 + 0.2\,\mathrm{rand}(\mathbf{x})\,, \\ v_0 &= 0\,. \end{cases} \tag{5.6.1}$$

We used the same random numbers for all simulations running on the same spatial discretization level, i.e., the initial conditions for the high-order solutions coincide with the ones of the AFC application when the spatial discretization is not altered. This allows us to compare the solutions obtained for one particular δh.

As standard model parameters we follow [16] and set $d_u = 1, \chi = 80, d_v = 0.33$. The computational domain is set to QUAD16 and the standard discretization yields $\delta h = 1/4$ and $\delta t = 1E$-2.

AFC for the transient aggregation model – high-order solutions

Figure 5.28 depicts the high-order solutions for our four iteration schemes at the dimensionless time instant $t = 1$, namely, after 100 time steps. We observe that numerical undershoots lead to negative solution values for all four iteration schemes. In order to compare the numerical results more carefully, in Figure 5.29 we plotted the solution values along the x_1/x_2–cutline $(0,0)-(16,16)$ which crosses the computational domain diagonally. The right plot zooms into the data marked by the highlighted box. We recognize that the undershoots mostly appear at the sharp interfaces of the separate accumulation sites, which is a well known characteristic for numerically instable transport equations. The cutlines for the monolithic and decoupled schemes are clearly distinguishable, whereas the difference between the linearized and the Richardson schemes is more subtle, which is most notable in the zoomed view. Finally, the cutlines of the Picard linearization and the Newton scheme practically coincide. These observations confirm our expectation of the qualitative comparison of our four underlying iteration schemes based on their mathematical development as outlined in Chapter 4.



(a) Dec

(b) Lin

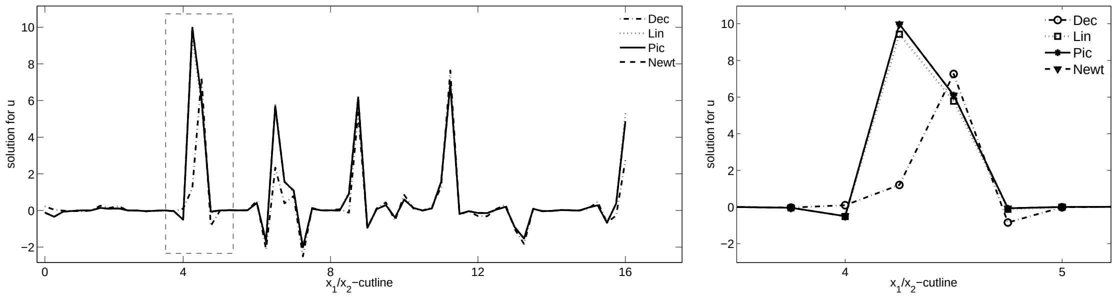
(c) Pic

(d) Newt

Figure 5.28: Aggregation model. High-order solutions at $t = 1$ for all four iteration schemes. Particularly we recognize the occurrence of numerical undershoots, negative values appear. Standard model and discretization parameters are used.

AFC for the transient aggregation model – AFC solutions

The AFC application renders the solution of our four iteration schemes positivity-preserving as Figure 5.30 evidently demonstrates. Therein we depict the AFC stabilized solutions for all four iteration schemes subject to the same model and discretization parameters as in Figure 5.28 before. Besides countering the negative solution values, the application of AFC also provides a more robust solution in terms of a consistent solution profile throughout all four iteration schemes, which can be best recognized from Figure 5.31. This figure sketches the solution profiles obtained from the solution values along the x_1/x_2–cutline $(0,0)-(16,16)$, as it was already done in Figure



(a) Full cutline

(b) Zoom on box

Figure 5.29: Aggregation model. Same high-order solutions as in Figure 5.28. This time the values along the x_1/x_2–cutline $(0,0)-(16,16)$ are shown. The solution profiles clearly reveal the numerical undershoots. Standard model and discretization parameters are used. The right figure zooms into the data of the marked box.

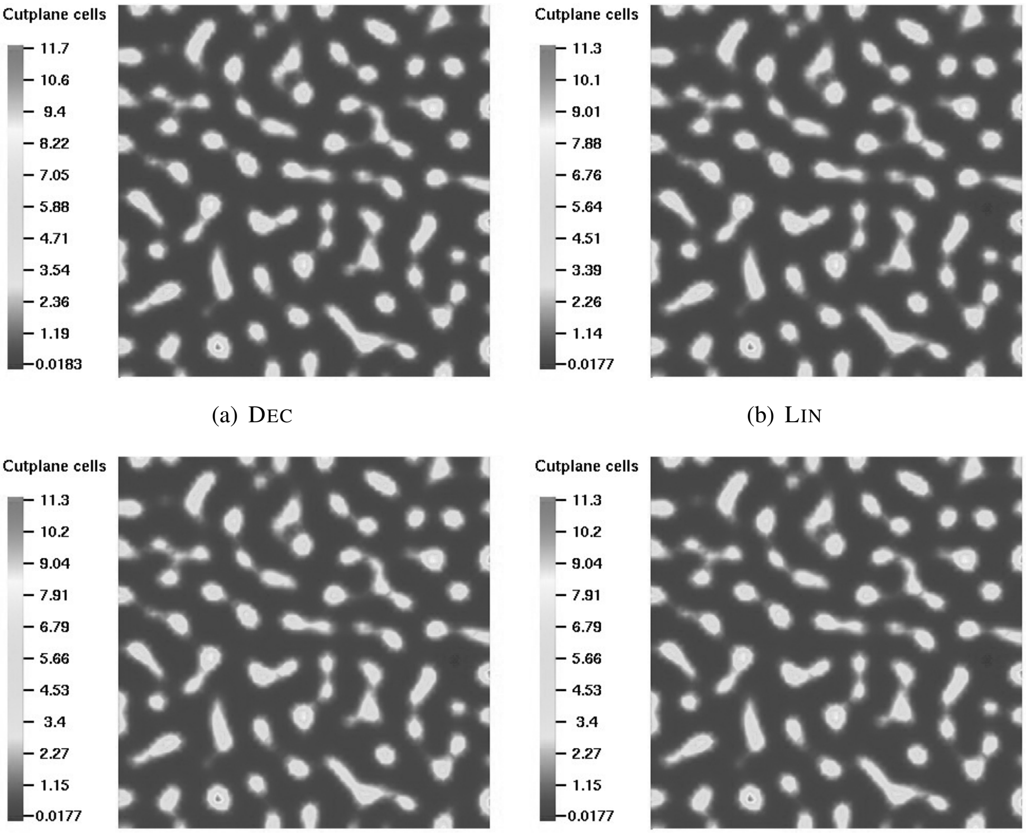
5.29. Moreover, the zoomed view on the right provides a clearer picture of the first solution peak marked by the box. Similar comparisons reveal that the four cutlines do not alter as much as they did for the high-order solutions. Even in the zoomed view, no significant differences between the four iteration schemes can be recognized. However, this stabilization comes along with a slightly diffused solution profile, which can be readily observed in the provided figures.

The robustness and stability of the AFC application allows for a qualitative analysis of the underlying aggregation model. Indeed, we can now draw our attention more closely to the relations of the involved model parameters without concerning too much about temporal or spatial discretization issues.

Experimental assays that study chemotaxis-driven aggregation can take several hours or even days of observation. Exemplary, in-vitro case studies of roughly 200 starving amoebae of the species *Dictyostelium discoideum* are tracked within an eight hour time frame[5]. With our stabilized schemes at hand we can address questions concerning the aggregation with less time consuming simulations. Exemplary in-silico experiments shown in Figure 5.32 run for less than five minutes (`Intel Core i7 X 980 @ 3.33 Mhz`). The provided results were obtained from the AFC stabilized variant of PIC. As we demonstrated above, cf. Figure 5.31, similar results could have been obtained with the other iteration schemes. Our numerical simulations were performed on CIRC16, discretized with a total of $9216\,\mathcal{Q}_1$ elements. The temporal discretization remains at $\delta t = 1E\text{-}2$.

When comparing the three different numerical simulation assays presented in Figure 5.32, we recognize that an increase of χ leads to a finer grained accumulation while the total number of aggregates roughly remains constant throughout the different strength of chemosensitivity. Because the underlying aggregation model yields mass conservation in the u equation, the stronger accumulation at larger χ necessarily implies a higher concentration at these aggregation sites. This can be easily verified by the corresponding color bars of the given screenshots which indicate the maximal and minimal solution values. These observations agree well with the nature of (attractive) chemotaxis flux.

[5] Laboratory for the Physics of Life, Princeton University. Blog of November 21, 2008: http://tglab.princeton.edu/blog/classic-papers/



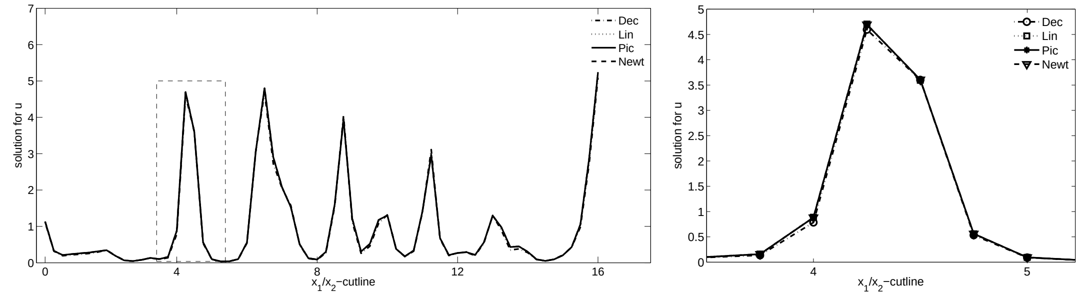
(a) DEC

(b) LIN

(c) PIC

(d) NEWT

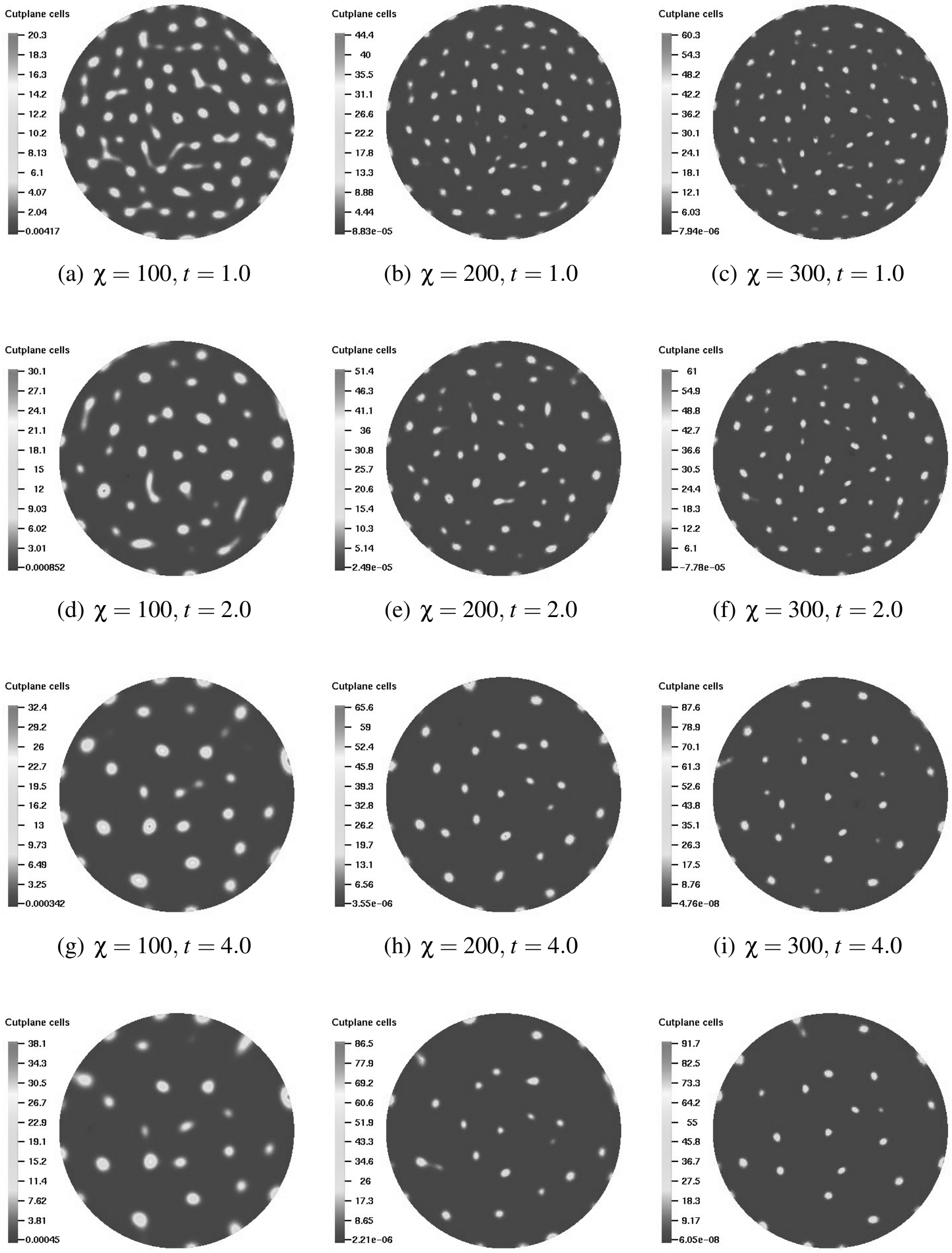
Figure 5.30: Aggregation model. Solutions obtained via AFC at $t = 1$ for all four iteration schemes. No undershoots or negative values appear. Standard model and discretization parameters are used.



(a) Full cutline

(b) Zoom on box

Figure 5.31: Aggregation model. Same AFC stabilized solutions as in Figure 5.30. This time the values along the x_1/x_2–cutline $(0,0) - (16,16)$ are shown. The positivity of the solution is clearly maintained. The right figure zooms into the data of the marked box.



(a) $\chi = 100, t = 1.0$

(b) $\chi = 200, t = 1.0$

(c) $\chi = 300, t = 1.0$

(d) $\chi = 100, t = 2.0$

(e) $\chi = 200, t = 2.0$

(f) $\chi = 300, t = 2.0$

(g) $\chi = 100, t = 4.0$

(h) $\chi = 200, t = 4.0$

(i) $\chi = 300, t = 4.0$

(j) $\chi = 100, t = 5.0$

(k) $\chi = 200, t = 5.0$

(l) $\chi = 300, t = 5.0$

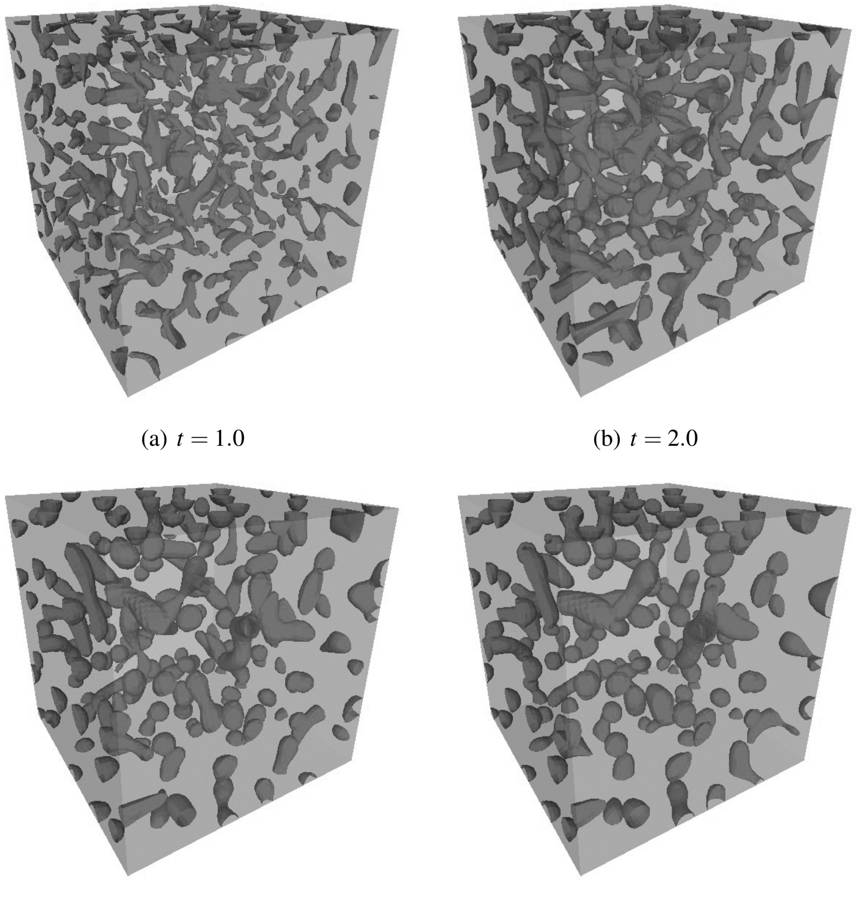
Figure 5.32: In-silico experiments of the aggregation of *Dictyostelium discoideum* with the underlying aggregation model for different chemosensitivities. Exemplarily we used the AFC stabilized variant of PIC for simulation. Every column shows snapshots of one particular simulation run, from left to right $\chi = 100, 200, 300$, $d_v = 0.33$ is fixed for all simulations. Discretization parameters: $\delta h \approx 1/8$ ($9216\,\mathcal{Q}_1$ elements) and $\delta t = 1E\text{-}2$.

AFC for the transient aggregation model – an exemplary 3D case

In correspondence to the 3D simulations for the minimal model of chemotaxis, we will now present particular results for a 3D application for the aggregation model provided by a previous paper of the current author [98]. Again, the AFC stabilized decoupled iteration scheme DEC was employed, for details the interested reader is referred to the provided reference [98].

Together with the usual Neumann boundary conditions (2.2.3) and the priorly defined initial conditions (5.6.1) the aggregation model (4.1.3) is simulated on the 3D cubic domain QUAD3D. The uniform temporal and spatial discretization was set to $\delta t = 0.01$ and $\delta h = 0.25$ (resulting in a total of 262,144 conforming trilinear finite elements), respectively. The simulations shown in Figure 5.33 were run with the same parameters as in the 2D case, i.e., $d_u = 1, d_v = 0.33, \chi = 80$.

From the Figure 5.33 we can easily track the aggregation which takes place after the random initial cell distribution undergoes chemotactic communication. The depicted isosurfaces refer to solution values of 5.0 in all subfigures. We can therefore readily observe (e.g., by counting the simple-connected cell batches) that, as time evolves, single cell agglomerations merge together to form bigger (more favorable) aggregates. Neither oscillatory solution profiles nor negative solution values are emerging.



(a) $t = 1.0$

(b) $t = 2.0$

(c) $t = 4.0$

(d) $t = 5.0$

Figure 5.33: Aggregation model. Simulation in 3D with the screenshots taken at the distinct times $t = 1.0, 2.0, 4.0, 5.0$.

5.6.3. AFC for the transient kinetic model

Kinetic models of kind (4.1.4) as stated in Section 4.1 provide a spatially proliferation of an initial inoculum. When prescribing a perturbed concentration positioned in the center of the domain, the solution exhibits multiple radially traveling waves. The traveling fronts leave trailing (spiky) spots as they propagate through the entire domain, cf. the numerical results of, e.g., [16, 55, 80, 87, 88, 96, 100]. The sensitivity of generated patterns and their richness require accurate and particularly stable numerical solver.

Our upcoming numerical study considers the model (3.3.1) which introduces a logistic growth term. We choose $\kappa = K = 1$ and simple linear reaction terms in the chemoattractant equation, so as to use the kinetic model (4.1.4) with a Fisher growth term

$$\left\{ \begin{array}{lcl} \partial_t u & = & \nabla \cdot \big(d_u \nabla u - \chi u \nabla c \big) + u(1-u)\,, \\ \partial_t v & = & \Delta v - \beta v + u\,. \end{array} \right. \tag{5.6.2}$$

Note that despite the carrying capacity of $K = 1$, the solution u is not limited to unity, as in the case of the original Fisher's equation. The reason is the non-saturating chemotaxis feedback which leads to local accumulation.

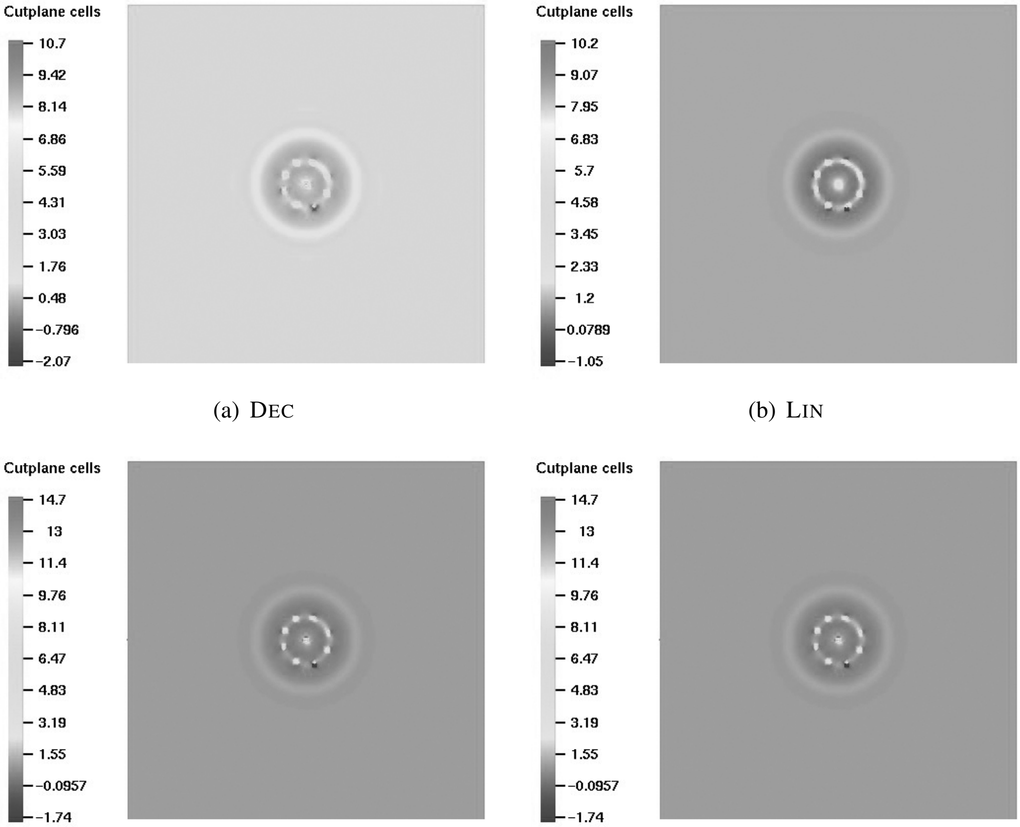
As underlying computational domain we use the square QUAD16. The numerical simulations of (5.6.2) are complemented by randomly perturbed initial data

$$\left\{ \begin{array}{lcl} u_0(\mathbf{x}) & = & \left\{ \begin{array}{ll} 1 + \mathrm{rand}(\mathbf{x})\,, & \text{for } ||\mathbf{x} - (8,8)^T|| \le 1.5 \\ 1 \qquad\qquad , & \text{otherwise} \end{array} \right. \\ v_0(\mathbf{x}) & = & 1/32\,. \end{array} \right. \tag{5.6.3}$$

The standard model parameters (after Aida *et al.* [2]) read $d_u = 0.0625, \chi = 8.5$ and $\beta = 32$. As standard discretization parameters we choose $\delta h = 1/8$ and $\delta t = 0.1$. Similarly to the aggregation model, we will examine the numerical instability of the high-order solutions and compare them with the results obtained from the AFC application.

AFC for the transient kinetic model – high-order solutions

Figure 5.34 displays screenshots of the high-order solutions for all four iteration schemes at the dimensionless time instant $t = 1.5$. From these figures we observe that negative values appear and pollute the solution profile. In fact, the resulting over- and undershoots lead to divergence of the underlying solver shortly after the depicted time instant ($t \le 2.2$). Excepting the very similar results of the two Richardson schemes, the solutions at $t = 1.5$ of the iteration schemes provide significantly different profiles as demonstrated in Figure 5.35, note the logarithmic scaling. The highly varying profiles can be best recognized in the zoomed view on the right of the figure. Thus, besides the numerical instabilities, the high-order solutions for the different iteration schemes are not as robust as in the case of rather moderate model and discretization parameters (compare the basic convergence analysis given in Section 5.4.1).



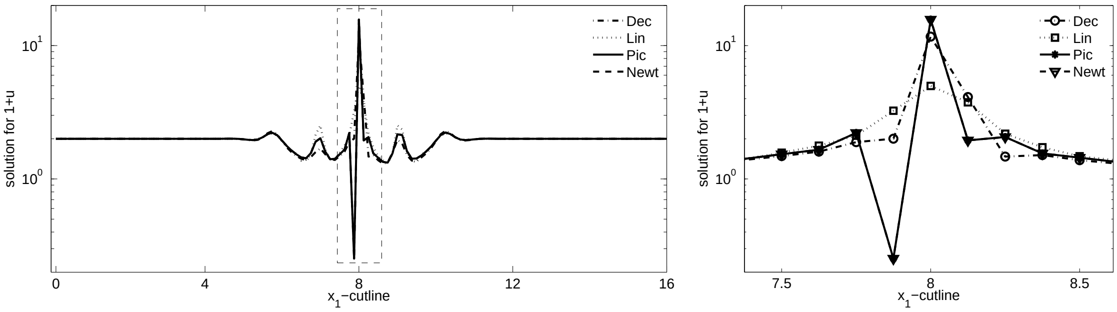
(a) DEC

(b) LIN

(c) PIC

(d) NEWT

Figure 5.34: Kinetic model. High-order solutions at $t = 1.5$ for all four iteration schemes. Note the appearance of negative values. Standard model and discretization parameters are used as given above.



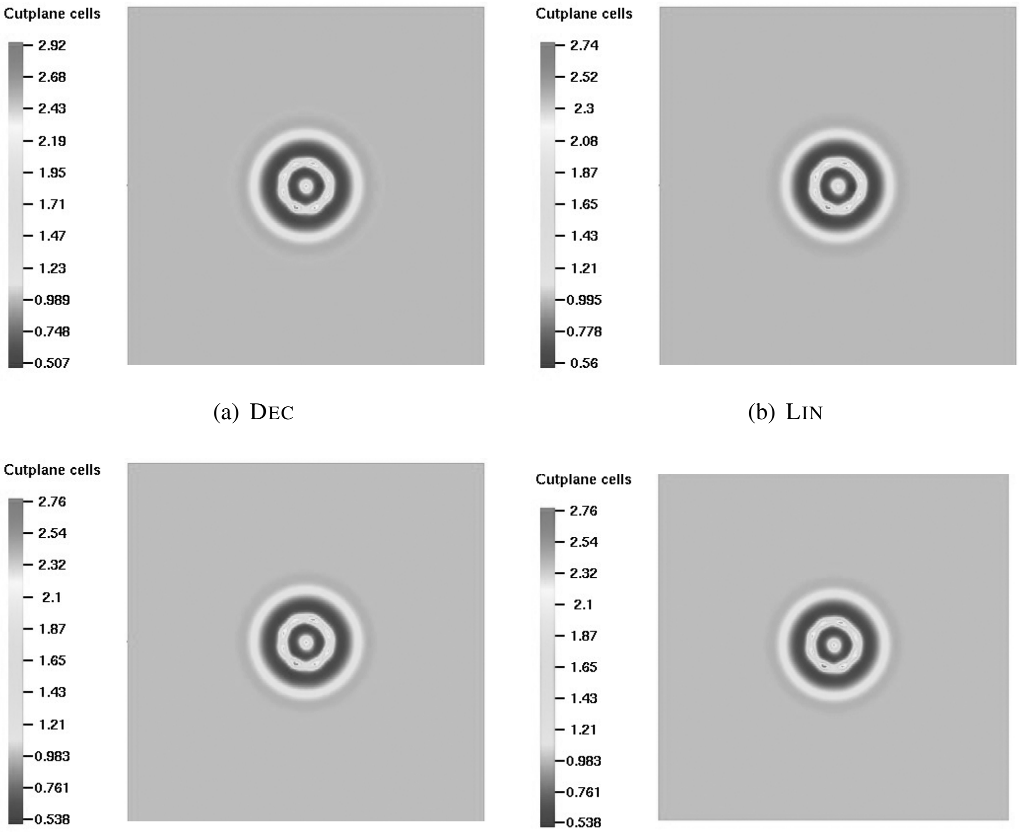
(a) Full cutline

(b) Zoom on box

Figure 5.35: Kinetic model. Same high-order solutions as in Figure 5.34. This time the values along the x_1–cutline $(0/16)-(8,8)$ are shown on a logarithmic scale, note that $1+u$ is displayed. Remark the highly varying solution profiles for DEC, LIN and both Richardson schemes (PIC and NEWT). Standard model and discretization parameters are used. The right figure zooms into the data of the marked box.

AFC for the transient kinetic model – AFC solutions

Now, let us consider the application of AFC on all our four iteration schemes. In Figure 5.36 we can readily acknowledge the positive solution values, the propagating fronts and the emergence of trailing spots. Notably, all four stabilized iteration schemes provide consistent solution profiles, which is even more recognizable in Figure 5.37. As in the high-order case, the solution values are plotted in logarithmic scale, which allows for a comparison with Figure 5.35.



(a) DEC

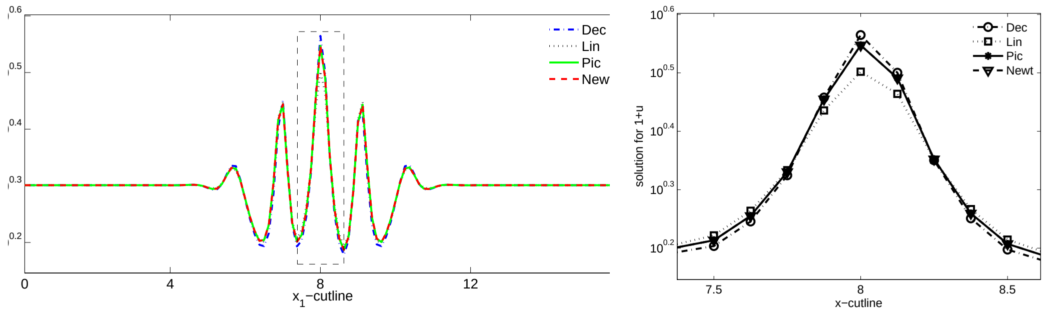
(b) LIN

(c) PIC

(d) NEWT

Figure 5.36: Kinetic model. AFC stabilized solutions at $t = 1.5$ for all four iteration schemes. All solutions remain positive in contrast to Figure 5.34. Standard model and discretization parameters are used as given above.

We see that the AFC stabilized schemes provide robust and meaningful results in terms of consistency and positivity preserving, non-oscillatory solution profiles and hence, allow for a more detailed analysis of the model parameters even on a rather coarse discretization level. In the following we illuminate the chemosensitivity dependent propagation of the emerging waves. We provide the AFC stabilized solutions for the PIC. In Figure 5.38 we examine the solutions character for an increasing chemosensitivity, $\chi = 8.5, 10, 15, 20$. Our results seem to be two-fold. On the one hand they reveal that the value of χ influences the speed of propagation, i.e., larger values of χ trigger a faster wave-like propagation until the boundary of the computational domain is reached. On the other hand, the screenshots show that our choice of χ only slightly influences the number of trailing spots. Note that the inner most circle consists of six trailing spots for $\chi = 8.5$, while for $\chi = 20$ one additional spot can be detected.



(a) Full cutline

(b) Zoom on box

Figure 5.37: Kinetic model. Same AFC stabilized solutions as in Figure 5.36. Again the values along the x_1–cutline $(0/16)-(8,8)$ are shown on a logarithmic scale, note that $1+u$ is displayed. In contrast to the high-order solutions, cf. Figure 5.35, the AFC stabilized profile for all iteration schemes do not differ substantially. Standard model and discretization parameters are used. The right figure zooms into the data of the marked box.

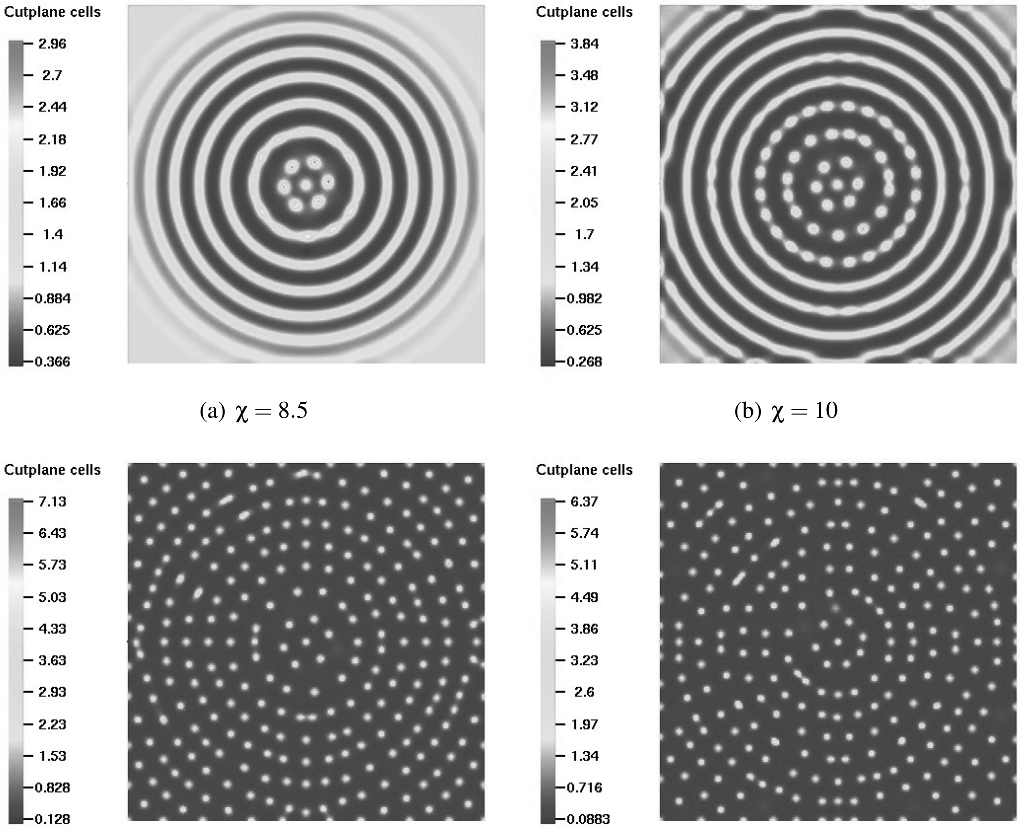
AFC for the transient kinetic model – an exemplary 3D case

Let us now have a look on the application of the AFC stabilization for the kinetic model (5.6.2) on a 3D computational domain, namely the sphere CIRC3D. The upcoming results can already be found in a preceding paper of the author [98] and were performed with the decoupled scheme DEC. We use the same boundary conditions as before and similar initial conditions

$$\begin{cases} u_0(\mathbf{x}) &= \begin{cases} 1+\text{rand}(\mathbf{x})\,, & \text{for } ||\mathbf{x}|| \le \sqrt{2} \\ 1 \qquad\qquad , & \text{otherwise} \end{cases} \\ v_0(\mathbf{x}) &= 1/32\,. \end{cases} \tag{5.6.4}$$

The uniform temporal discretization was set to $\delta t = 0.1$ while the spatial mesh was discretized to provide a total of 4,194,304 conforming trilinear finite elements. For the following simulations we choose the same model parameters as before, namely $d_u = 0.0625, \chi = 8.5$ and $\beta = 32$. In Figure 5.39 we observe how the initial random distribution of cells propagates into the whole domain in a moving-wave pattern where trailing spots are left as the advancing wave-front moves on. Furthermore we observe that the solution values remain positive and no oscillations occur.

After the wave-fronts hit the boundary of the domain the trailing spots tend to aggregate in a chaotic manner (not shown). Indeed, the longterm behavior of chemotaxis models incorporating kinetic terms is subject of current research as it is not entirely investigated if steady states exists and how they are influenced by the chemosensitivity.



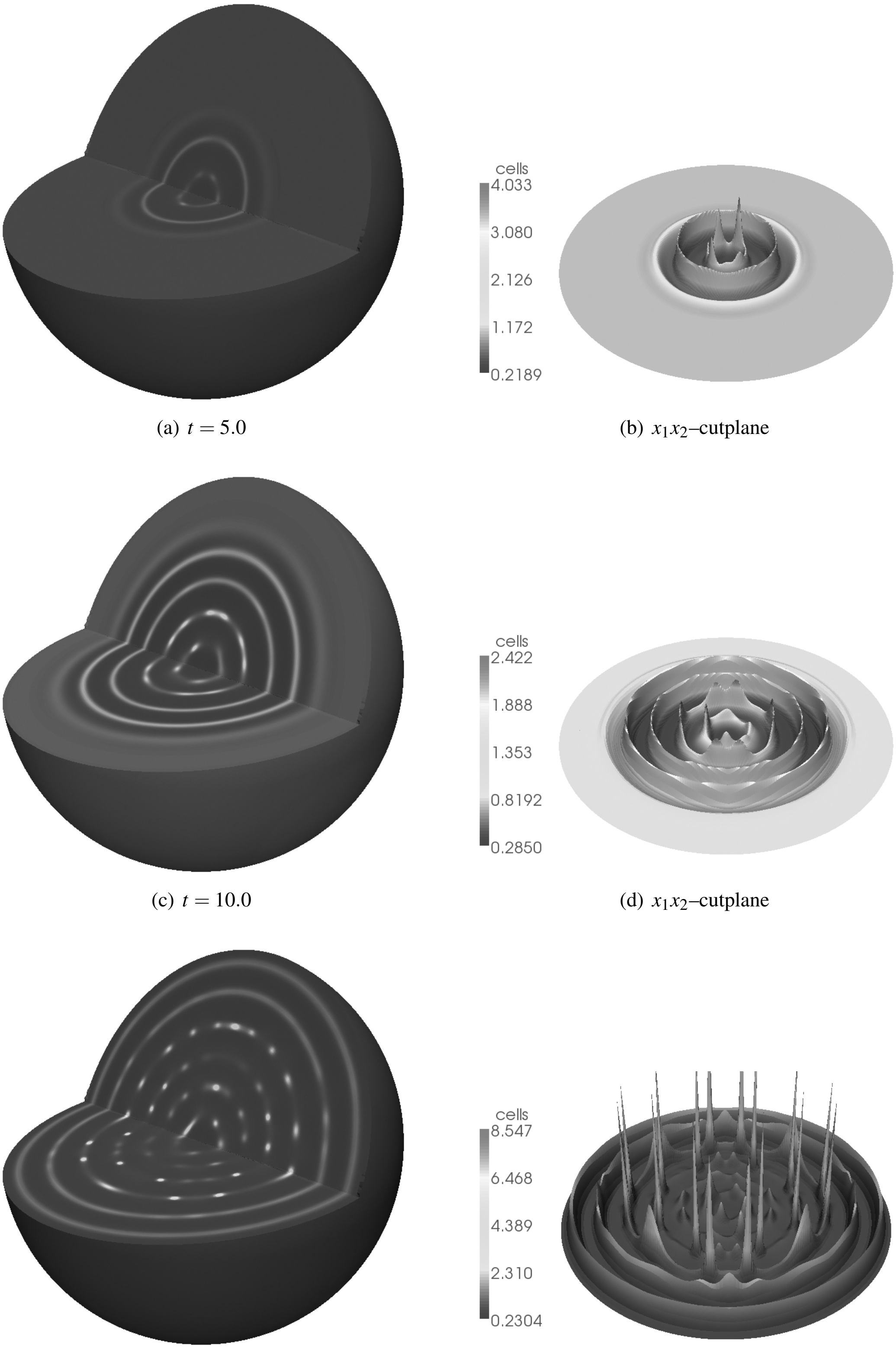
(a) $\chi = 8.5$

(b) $\chi = 10$

(c) $\chi = 15$

(d) $\chi = 20$

Figure 5.38: Kinetic model. AFC stabilized solutions obtained with PIC at $t = 10$ for different values of χ. Standard model and discretization parameters are used as given above.



(a) $t = 5.0$

(b) x_1x_2–cutplane

(c) $t = 10.0$

(d) x_1x_2–cutplane

(e) $t = 20.0$

(f) x_1x_2–cutplane

Figure 5.39: Kinetic model. AFC stabilized cell distribution (after [98]) obtained with DEC. The screenshots were taken at times $t = 5.0, 10.0, 20.0$. Left column: sliced spherical domain, gray-scale-coded cell concentration from low to high. Right column: centered x_1x_2–cutplanes of the cell concentration, gray-scale- and heights-coded cell concentration from low to high.

5.7. Numerical summary

Let us summarize the numerical results provided by the preceding sections to have a brief compendium of our numerical investigation.

In order to identify a robust and efficient solver algorithm for the resulting discrete linear systems, we analyzed the applicability of a geometric multigrid solver. In the present, multigrid frameworks become increasingly popular because of their wide-spread practicability for a variety of (also non-elliptic) problems under consideration. In our numerical analysis we found out that multigrid algorithms can indeed improve the robustness and practicability of the solving process for chemotaxis-dominated PDEs, particularly in the case of a large chemosensitivity (Section 5.3.1).

The exploration of the role that chemosensitivity plays in the successful application of a numerical scheme is indeed of paramount interest. In the course of the numerical assays which focused on this, we observed that (i) the numerical error remarkably scales with χ for all underlying models and (ii) the difference between the solutions of the four iteration schemes also significantly scales with χ. These results emphasize that chemosensitivity plays a crucial role for the accuracy and overall efficiency of numerical solution algorithms. Hence, a proper numerical treatment of the corresponding discrete terms are of utmost relevance (Section 5.4).

Moreover, the study of the different iteration schemes focused on their consistency with the analytical properties of the underlying chemotaxis models. In this context we can state that the decoupled and the linearized variants of our iteration schemes provide misleading solutions if the simulation parameters are chosen inappropriately. To the best of the authors belief, this issue has been revealed for the first time. The outcome is that the numerical analyst must either be aware of suitable discretization parameters for convenient numerical schemes or he/she is advised to use more elaborated solver algorithms, such as the nonlinear Richardson schemes. For the latter, we examined their efficiency concerning the required nonlinear iterations. Our findings clearly promote Newton's method for a coarse time stepping, emphasizing the strength of nonlinearity in the governing models (Section 5.5).

Since all of the high-order solution schemes recapitulated so far suffer from numerical artifacts which pollute the solutions in generic cases, stabilization techniques are highly recommended. With the advent of AFC, a promising and flexible stabilization technique for FEM has been identified and successfully applied to chemotaxis models. Our results show that all our numerical schemes can be stabilized, allowing for a more detailed theoretical analysis of the governing PDE model, e.g., approximating the blow-up time or the speed of traveling waves/pattern formation. Moreover, with the AFC stabilization, we obtained numerical schemes that provide a consistent solution. Throughout all four iteration schemes, the AFC stabilization remarkably weakened the high volatility of the difference between the solutions: after the application of AFC all four schemes provide similar solutions, which was not the case for the high-order solutions (Section 5.6).

Finally, let us point out that our numerical results also provided certain points for optimization/improvement of the algorithms in terms of accuracy and, in particular, computational efficiency. One of the main points is the calculation of the Jacobian in the Newton-like method for the low-order and AFC scheme. The currently implemented expensive calculation limits its application to a rather moderately refined spatial discretization. Possible algorithmic improvements of the current implementation will be one of the topics in the last concluding chapter of this work, Chapter 6, where we will conclude our work and discuss some ideas for ongoing investigations.

6 Conclusion

In this last chapter of the present work, we will provide the reader with some concluding paragraphs. In Section 6.1, we start with a short reflection of what we have studied in this thesis and emphasize the main achievements of our numerical investigation. Subsequently, we point out in Section 6.2 the conclusions of our work and discuss them in a broader context. The final Section 6.3 of this chapter is devoted to possible further extensions of our work and ongoing researches in the field of chemotaxis-driven PDEs. This outlook is divided into two parts. While the first part, Section 6.3.1, considers algorithmic improvements of our presented numerical schemes, the second part, Section 6.3.2, deals with future applications of our numerical methodologies in the field of Mathematical Biology.

6.1. Summary

Owing to our belief in providing a self-contained work, we started our thesis by demonstrating how chemotaxis models can be derived via simple argumentation about a biased random walk. The motivation of both strands of derivations, microscopic and macroscopic, can be easily explained. However, the application of the limiting process in the microscopic derivation is vitally discussed in the community, particularly because the employment of a PDE approach for the development of cells cannot be recommended in general. Nevertheless, for our purpose, the formulation of a PDE system for the cell-chemical interaction is well-suited. Besides the modeling, we also pointed out that basic chemotaxis PDEs are still not completely solved by theoretical analysts, despite their wide-ranged consideration in real life applications. Many open questions, particularly concerning the development of the cell concentration, cannot be tackled by today's analytical methods and hence demand numerical approaches. Since chemotaxis processes take mostly place in a highly vital environment, e.g., a living body, a growing tissue or a moving cell, a numerical methodology that can easily cope with unstructured evolving meshes is highly recommended.

In our work, we employed FEM for the spatial discretization and presented different corresponding temporal discretization strategies, namely a decoupled approach (derived from a Schur complement ansatz), an explicit linearization technique, a Picard linearization and Newton's method. The temporal discretization was exerted by the method of lines, where the employed one-step theta-scheme gives rise to an overall second-order FE approximation. Besides these basic discretization schemes, one of the main assets of this work was the formulation of a proper AFC stabilization scheme which renders the FE solution robust and positivity preserving, two properties that basic high-order FE discretization schemes can hardly meet in general. It has already been approved in preliminary publications by the author and his collaborators that the application of AFC is primarily required for the discretized chemotaxis operator, whereas all remaining terms are mainly well-conditioned, at least under some mild assumptions on the spatio-temporal discretization. Moreover, for reasons of efficiency we employed a linearization technique for coping with the anti-diffusive fluxes, which are, surprisingly enough, of implicit character, even in the case of an explicit time integration. A smart implementation of the overall resulting AFC stabilized FE discretization schemes is of paramount interest when competing with existing FE stabilization techniques. The edge-wise formulation of fluxes by Kuzmin, e.g., [61, section 2.1.8], provides favorable properties which readily enhances the assembly of the stabilizing and correcting fluxes. Furthermore, regarding Newton's method, careful considerations regarding the differentiation of the discrete Upwinding should be taken into account. In our work, we applied first order divided differences that can be understood as mimicking algorithmic differentiation [74].

After we presented the proper formulation of our discretization schemes, we focused on their application to some selected chemotaxis-driven models. We validated our numerical schemes in terms of consistency and convergence and mainly obtained results that meet our experienced expectations. In order to extend our basic numerical analysis we furnished them with studying the effect of increasing chemosensitivities and also proposed certain statements about the overall computational efficiency of the application of the underlying discretization schemes. These studies allowed us to compare the different high-order discretization schemes applied on chemotaxis models in detail for the first time and revealed some remarkable drawbacks of the decoupled and the linearized approaches. Furthermore, we demonstrated that our standard high-order schemes crucially limit the numerical analysis of the underlying chemotaxis models because of the emerging numerical pollution of the approximate solution, i.e., negative values and severe numerical oscillations occur that eventually lead to divergence of the numerical schemes. In order to address this problem, we applied the AFC stabilization technique in these situations. Our findings document the tremendous benefit of the AFC stabilization, not only since it renders the solution positivity-preserving and oscillation-free, but particularly because the stabilized solution allows us to analyze the underlying PDE models in much more detail, e.g., approximating blow-up time, exploring emerging patterns and the speed of propagation. Another selling-point of the FE stabilization technique via AFC is its flexibility concerning the structure of the underlying mesh, particularly for 3D applications. For this reason, we provided the reader with selected results of AFC stabilized solutions in 3D domains.

We saw that this thesis not exclusively deals with the formulation and application of an accurate, stabilized and efficient numerical solver for chemotaxis-driven PDEs, but that it also covers some introductory principles of the modeling and a brief survey of selected analytical results concerning basic chemotaxis processes occurring in Mathematical Biology. Besides pure mathematical ambitions, in the belief of the author, these non-numerical insights were mandatory in order to provide the reader with basic and mandatory background information on the subject of chemotaxis modeling that significantly eases the understanding, finally rendering this thesis self-contained. Moreover, particularly since Mathematical and Computational Biology are rather young research disciplines, these modeling and analytical insights can also provide a source of fruitful discussions or even promising new challenges and tasks for future work.

6.2. Discussion

The obligatory task of formulating an appropriate ‘take home message’ for this thesis, in terms of a conclusion, is quite delicate. On the one hand, keeping in mind the proclaimed self-contained character of this work, we may tend to emphasize the potent numerical results in terms of

i) the comparative nature of the computational accuracy and efficiency of the numerical schemes in Section 5.4.2;

ii) the identification of remarkable drawbacks of certain discretization schemes in Section 5.5;

iii) the flexible and stabilizing AFC application in Section 5.6.

On the other hand, as we partially already mentioned before, our employed numerical scheme gives rise to certain algorithmic enhancements, with some of their ideas already present in the literature provided in Section 5.1 on the numerical state of the art. For example this comprises parallelism, temporal and, particularly, spatial adaptivity.

Therefore, we like to include the modeling principles and the analytical results into the following consideration for this paragraph. Despite its strongly debated justifications, many chemotaxis-driven models are based on PDE approaches for cells and chemicals modeled at a similar scaling level. Presently, special focus has been on modeling and simulating tumor development in some early stages. In these stages, however, the tumor may effectively consist of a number of abnormal cells that is too low to neglect individual cell development, rendering a continuous approach rather inconvenient. Thus, before considering suitable discretization schemes for PDEs, we have to take into account the modeling framework of our underlying problem. This may seem to be a trivial and obvious concern. However, in the context of living matter, e.g., developing cell compounds, this is a crucial task, since the modeling framework may have to be adjusted to the current state of the development, e.g., growing and proliferating, living matter under consideration.

Another important concern when dealing with chemotaxis-dominated PDEs are the analytical results which we partially summarized in Section 2.2. Hereby, the blow-up results may be paid the most attention to. It is not only the lack of pure analytical results for this phenomenon in higher dimensions, but it is also the numerical challenge that arises from capturing this effect adequately. Spiky solution profiles are commonly regarded as being of pure numerical character. Hence, when simulating models that tend to blow up, analysts may simply be misled, because they assume their solution being polluted by some numerical artifacts. Despite the fact that models from which blow-ups may emerge seem to be rather senseless from the biological perspective, their (mathematical) study faces some fascinating questions, e.g., about determining the blow-up time and critical masses in higher dimensions, cf. the open problems at the end of Section 2.2. We have to also acknowledge that the minimal model of chemotaxis, from which blow-ups may emerge, serves as a ‘blueprint’ for extended and more complex models for chemotaxis-driven processes. That is, the (mathematical) study of this basic model should be claimed to be of high priority before considering to proceed with much more complex chemotaxis-driven models. With the recent increase of numerical interest and computational power, we arrived at a point where the analytical instruments can be readily complemented by the huge wealth of numerical tools. In the authors opinion, this symbiosis has tremendous potential to lead to fruitful new insights into analytical problems, such as the blow-up challenge, studying developing patterns and sets of possible steady states. Indeed, linear analysis as introduced in the appendix, cannot reveal the complete development of feasible patterns and a nonlinear analysis, as exerted by e.g., Tyson *et al.* [100] and Murray [79], is very tedious.

When we have taken this into account and strive to simulate a PDE system of general kind (4.1.1) the numerical studies conducted in this work provide valuable assets for numerical analysts, see (i)–(iii), in order to identify a most appropriate numerical solver scheme for their particular chemotaxis-driven PDE model. As our conclusion is two-fold, it is advisable to mention the subtle restrictions concerning the efficiency of our presented discretization schemes. While the (relative) comparisons across our four standard high-order schemes remain justified, their absolute performance could have readily been enhanced by implementing parallel and adaptive algorithms from the literature, see Section 5.1. Since we do not claim to provide a computationally optimized code and since we understand the development of efficient and accurate numerical schemes as a long term evolving process which keeps improving in terms of accuracy, memory concerns and rate of floating point operations per second, this present thesis can be used as reference for a fundamental numerical analysis of chemotaxis-driven PDEs rather than as a fully developed and 'ready-for-market' product of a solver for such PDEs. Nonetheless, we particularly like to stress that the application of an AFC stabilized scheme on chemotaxis-driven models reveals a potent, flexible, robust, accurate and efficient numerical tool for investigating chemotaxis processes at a promising high level of details for the first time.

6.3. Outlook

As we have already mentioned above, our numerical framework could be extended by algorithmic improvements which potentially enhance the overall performance. Besides considering these technical challenges in some more detail, this section is devoted to providing the reader with an outlook on possible further investigations in the field of numerical treatment of chemotaxis-driven PDEs. We are certainly aware of the fact that we cannot cover all the associations which the interested reader may come up with, particularly because Computational Biology is such a youthful discipline and hence offers an overwhelming pool of possible further studies. We will therefore confine ourselves to selected ideas for ongoing research which the author personally got into touch with in the pursuit of this thesis. Similar considerations hold for the algorithmic improvements.

6.3.1. Algorithmic improvements

Stable higher-order time integrators such as multi-step, strong stability preserving Runge-Kutta methods [54] or backward differences formulas and higher-order finite elements, e.g., $\mathcal{Q}_2$ (biquadratic polynomials), are the most convenient approaches to improve existing PDE solvers. Moreover, it is often advisable to employ higher-order discretization techniques in time and space at a similar scale, simply because the time stepping and the spatial mesh size can often not be independently chosen, cf. CFL-like restrictions, and they contribute equally to the overall FE approximation error. As we will remark in some of the following selected ideas for possible future investigations, the proper application of higher-order discretizations can nevertheless be quite a challenging task. Besides this consideration, the subsequent objectives aim at providing ideas how to improve the nonlinear numerical schemes and the AFC application.

Adaptive time stepping In the present work, we introduced numerical schemes with a uniform time stepping. While this is very handy for a simple numerical analysis, particularly to compare different solutions and/or schemes, it is definitely preferable to employ adaptive time stepping methods when solutions undergo a mixture of transient and stationary stages.

This is indeed often the case in chemotaxis-driven PDEs. For instance, consider the kinetic model: in a first stage, the kinetic terms drive the solution to form propagating rings that spread out into the entire computational domain and, in a second stage, the chemotaxis term drives a rather chaotic aggregation phase which happens at a considerably lower speed as the wave propagation.

In the context of the aforementioned higher-order multi-step time integrators, we like to note that the proper implementation of adaptive time steps may be a tedious task. Remark that a R-step method requires memory for $p \cdot R$ solution vectors if we allow a p-fold step width increase. Furthermore, we would have to interpolate the solution values of $u(\cdot, \mathbf{x})$ at up to R intermediate time steps when decreasing the step width, e.g., we require at time $t = n\,\delta t$ approximations of the solutions $u([n - r/2]\delta t, \mathbf{x})$, $r = 1, \ldots, R$, if the step width is halved.

Adaptive spatial discretization Besides an adaptive temporal discretization the highly localized chemotaxis aggregation gives naturally rise to the application of *r-/h-/p-adaption schemes*, cf. Kirk and Carey's statement in [55] (quoted already in Section 5.1). The *r-adaptivity* 'deforms' the underlying spatial mesh to better fit to the locations that require a finer discretization. Particularly, it does not introduce new degrees of freedom. However, the deformation should still yield certain moderate regularity constraints in order not to degenerate discrete operators such as diffusion. The well-known *h-adaptivity* simply refines the spatial mesh locally, e.g., certain (quadrilateral) elements. While this approach is often practiced for triangular elements, its application for quadrilateral elements might lead to the challenge of handling hanging nodes. The so-called polynomial adaptivity, *p-adaptivity*, augments the order of local finite elements, e.g., while one element is of type $\mathcal{Q}_1$, the next one may be of kind $\mathcal{Q}_2$. We will see in the last point, why a complete 'tessellation' with $\mathcal{Q}_2$ elements cannot necessarily be recommended.

Modifications of Newton's method From the analysis of the AFC stabilization, in particular from the construction of the Jacobian, we recognize the poor efficiency of the current algorithm for NEWT. The explicit derivation of the Jacobian is an expensive task and therefore it limits the overall efficiency, although it provides physically meaningful solutions. There are indeed some conceivable approaches that might remedy this problem of inefficiency. Certainly, these approaches need to be validated by numerical analysis and hence give rise to further research challenges.

For the ordinary Newton method, the Jacobian needs to be calculated in every nonlinear iteration. One way to circumvent this task is to use one stationary Jacobian for all nonlinear iterations, i.e., we shift the calculation of the Jacobian outside of the nonlinear loop. When evaluating the Jacobian only at the initial guess we end up with the so-called *Simplified Newton*. Resulting from the geometric perspective this method is also called the *Chord method*. The iteration reads

$$\begin{aligned} jac\big(\mathcal{A}(\mathbf{w}_0)\big)\,\mathbf{y} &= res^n(\mathbf{w}_m), \\ \mathbf{w}_{m+1} &= \mathbf{w}_m + \mathbf{y}. \end{aligned}$$

The remedy to the explicit calculation of the Jacobian in every step is accompanied by its approximate nature. For the overall applicability of the simplified Newton methods we have to take into account the possibly increasing number of nonlinear iterations and the smaller region of convergence.

Even if we maintain the derivation of the Jacobian in every iteration it might be more efficient to use an approximation of it, which leads to so-called *Newton-like methods*. Besides approximating the exact Jacobian by divided difference schemes or Taylor expansions, as it is required if a matrix-vector representation of the Jacobian-vector product cannot be achieved [97], another promising approximation of the Jacobian aims at dropping weak couplings. This may result in more favorable algebraic properties of the approximated Jacobian, such as sparsity patterns, bandwidth, diagonal dominance or factorization properties.

Splitting of Upwind The next three points consider improvements of the application of AFC. Therefore we may suggest the interested reader to briefly recapitulate the general concept of AFC introduced in Section 4.5.1.

We saw in Section 4.5.4 that the nonlinearity of the Upwind operator $\mathbf{D}(\cdot)$ prohibits a splitting that would promote some pre-calculations in order to save some computation time, see (4.5.26). However, we also recognized that the contributions to the approximate Jacobian stemming from the perturbation of the chemotaxis operator in the $\mathbf{e}_j$-direction, $\mathbf{K}(\mathbf{e}_j)$, only modify a total of 81 matrix entries, see (4.5.27). In other words, the inequality (4.5.26) only holds for these 81 matrix entries. This gives rise to another accelerating technique. The transition from $\widetilde{\mathbf{K}}_1(\mathbf{w})$ to $\widetilde{\mathbf{K}}_1(\mathbf{w}_j^{\pm})$ only alters the 81 matrix entries determined by the indices of the set $\mathcal{N}_j$. Therefore, the calculation of $\widetilde{\mathbf{K}}_1(\mathbf{w}_j^{\pm})$ in line 8 of Algorithm 4.5 can be readily simplified.

Let $\mathbf{K}_1^{\complement\mathcal{N}_j}(\mathbf{w})$ denote the usual discrete chemotaxis operator $\mathbf{K}_1(\mathbf{w})$ where all matrix entries corresponding to $\mathcal{N}_j$ have been set to zero, i.e.,

$$\left[\mathbf{K}_1^{\complement\mathcal{N}_j}(\mathbf{w})\right]_{kl} = \begin{cases} \left[\mathbf{K}_1(\mathbf{w})\right]_{kl} & \text{, for } k,l \notin \mathcal{N}_j\,, \\ 0 & \text{, for } k,l \in \mathcal{N}_j\,. \end{cases}$$

Furthermore, let $\mathbf{K}_1^{\mathcal{N}_j}(\mathbf{w})$ denote the matrix obtained from the matrix entries of $\mathbf{K}_1(\mathbf{w})$ corresponding to $\mathcal{N}_j$, i.e.,

$$\left[\mathbf{K}_1^{\mathcal{N}_j}(\mathbf{w})\right]_{kl} = \begin{cases} \left[\mathbf{K}_1(\mathbf{w})\right]_{kl}, & \text{for } k,l \in \mathcal{N}_j\,, \\ 0, & \text{for } k,l \notin \mathcal{N}_j\,. \end{cases}$$

Thus, for each fixed j we can write

$$\widetilde{\mathbf{K}}_1(\mathbf{w}_j^{\pm}) = \widetilde{\mathbf{K}_1^{\complement\mathcal{N}_j}}(\mathbf{w}) + \widetilde{\mathbf{K}_1^{\mathcal{N}_j}}(\mathbf{w}_j^{\pm})\,,$$

where $\mathbf{K}_1^{\mathcal{N}_j}(\mathbf{w}_j^{\pm}) = \mathbf{K}_1^{\mathcal{N}_j}(\mathbf{w}) \pm \sigma\mathbf{K}_1^{\mathcal{N}_j}(\mathbf{e}_j)$. Furthermore, with this formulation in mind, the calculation of the (1,2) block of the Jacobian (4.5.22) significantly simplifies. We note that

$$\begin{aligned} \widetilde{\mathbf{K}}_1(\mathbf{w}_j^{+}) - \widetilde{\mathbf{K}}_1(\mathbf{w}_j^{-}) &= \left[\widetilde{\mathbf{K}_1^{\complement\mathcal{N}_j}}(\mathbf{w}) + \widetilde{\mathbf{K}_1^{\mathcal{N}_j}}(\mathbf{w}_j^{+})\right] - \left[\widetilde{\mathbf{K}_1^{\complement\mathcal{N}_j}}(\mathbf{w}) + \widetilde{\mathbf{K}_1^{\mathcal{N}_j}}(\mathbf{w}_j^{-})\right] \\ & \widetilde{\mathbf{K}_1^{\mathcal{N}_j}}(\mathbf{w}_j^{+}) - \widetilde{\mathbf{K}_1^{\mathcal{N}_j}}(\mathbf{w}_j^{-})\,. \end{aligned}$$

Thus for the (1,2) block of the Jacobian, Upwinding can be readily accelerated, because it suffices to apply it only on the (possibly) 81 non-zero entries of $\mathbf{K}_1^{\mathcal{N}_j}(\mathbf{w}_j^{\pm})$. Algorithm 6.1 depicts these improvements. Therein the differences to Algorithm 4.5 are indicated by comments.

Algorithm 6.1 Improved computing of the approximated low-order Jacobian $\widetilde{\mathbf{J}}(\mathbf{w})$ (given $\mathbf{w}$)

```
Require: Let us assume that all linear matrices, i.e., $\mathbf{M}, \mathbf{M}_L$ and $\mathbf{L}$, and all parameters are passed to this routine
Assemble $\mathbf{K}_1(\mathbf{w})$
Calculate $\widetilde{\mathbf{K}_1}(\mathbf{w}) = \mathbf{K}_1(\mathbf{w}) + \mathbf{D}(\mathbf{w})$
Build jacblock(1,1): $\left[\widetilde{\mathbf{J}}(\mathbf{w})\right]_{i \leq \mathrm{N}}^{j \leq \mathrm{N}} = \left[\mathbf{M}_L + \theta\,\delta t\left(\mathbf{L} - \widetilde{\mathbf{K}_1}(\mathbf{w})\right)\right]_{ij}$
Build jacblock(2,1): $\left[\widetilde{\mathbf{J}}(\mathbf{w})\right]_{i > \mathrm{N}}^{j \leq \mathrm{N}} = -\theta\,\delta t\,\mathbf{M}_{ij}$
Build jacblock(2,2): $\left[\widetilde{\mathbf{J}}(\mathbf{w})\right]_{i > \mathrm{N}}^{j > \mathrm{N}} = \left[\mathbf{M} + \theta\,\delta t\,(d_v\,\mathbf{L} + \mathbf{M})\right]_{ij}$
for $j = \mathrm{N} + 1, 2\mathrm{N}$ do
    Extract ${\mathbf{K}_1}^{\mathcal{N}_j}(\mathbf{w})$ and $\widetilde{\mathbf{K}_1^{\complement\mathcal{N}_j}}(\mathbf{w})$    ▷ no explicit assembly is necessary
    Assemble $\mathbf{K}_1^{\mathcal{N}_j}(\mathbf{e}_j)$
    Calculate $\mathbf{K}_1^{\mathcal{N}_j}(\mathbf{w}_j^{\pm}) = {\mathbf{K}_1}^{\mathcal{N}_j}(\mathbf{w}) \pm \sigma\,\mathbf{K}_1(\mathbf{e}_j)$    ▷ only 81 entries are considered
    Calculate $\widetilde{\mathbf{K}_1^{\mathcal{N}_j}}(\mathbf{w}_j^{\pm})$    ▷ Upwinding for a 81 element matrix
    Build jacblock(1,2): $\left[\widetilde{\mathbf{J}}(\mathbf{w})\right]_{i \in \mathcal{N}_j}^{j} = \frac{\theta\,\delta t}{2\sigma}\left[\left(\widetilde{\mathbf{K}_1^{\mathcal{N}_j}}(\mathbf{w}_j^{+}) - \widetilde{\mathbf{K}_1^{\mathcal{N}_j}}(\mathbf{w}_j^{-})\right)\mathbf{u}\right]_{ij}$
end for
```

Non-conservative AFC schemes AFC schemes of the kind we proposed do currently not explicitly take into account reaction terms which give rise to the birth of local extrema. As long as these terms do not harm the positivity of the numerical scheme, they obviously do not require any further numerical attention. However, the AFC stabilization of reaction terms that cause severe hazards to the positivity, is an objective of current research. From the brief survey in [61, section 1.6.3.3] we acknowledge preliminary studies in that direction. In his seminal book, Patankar [89] introduced a *negative-slope linearization* technique which yields positivity preservation for (nonlinear) reaction terms, say $r(u)$, which can be reformulated as

$$r(u) \quad = \quad r_+ - r_-\,u\,,$$

where the parameters r_+ and r_- may also depend on the unknown solution values and yield non-negativity, $r_+ \geq 0$, $r_- \geq 0$. More recently, MacKinnon and Carey [69] studied the positivity preservation of the above reformulation of reaction terms, where $r_+ = \alpha u$, α being a constant. Particularly, they remarked that such a favorable reformulation is possible in the case of a combination of first- and second-order kinetics which we already encountered in (3.3.1). Hence, the detailed exploration of non-conservative AFC schemes undoubtedly provide promising improvements of the stabilization of chemotaxis-driven PDEs.

Adaptive AFC schemes Another improvement of the currently implemented AFC stabilization aims at combining the stabilization with adaptive FEM schemes. Besides the widely employed adaptive time stepping schemes and r-/h-adaption for the underlying spatial mesh, current research has a special focus on coupling AFC with p-adaptivity. Since the development of properly defined correcting fluxes is an utmost difficult task for higher-order elements such as $\mathcal{Q}_2$, very recent considerations of Bittl and Kuzmin [10] promote the use of higher-oder elements in regions of smooth solutions and (bi-)linear elements in regions of steep fronts.

6.3.2. Future applications

Our entire numerical framework was developed to cover its application to a variety of chemotaxis PDE models, cf. the general formulation of the model (4.1.1). Although, in this work, we have focused our numerical analysis on three particular models, there is no doubt that considerations of similar models can be easily undertaken and would deliver similar and qualitatively comparable results. As this thesis provides a reference for fundamental numerical analysis of chemotaxis-driven PDEs, our findings can be used by future modelers and numerical analysts to get familiar with the numerical properties of selected solver methodologies and recognize the huge potential of the presented AFC application for chemotaxis models, allowing a detailed view on the modeled biological process by liaising between (experimental) biologists, (mathematical) modelers and numerical analysts.

From the practical point of view the probably most interesting question is, whether we can use the stabilized FE scheme for our kind of model (if it does not fit into the formulation 4.1.1)? Let us therefore propose some possible further applications of our AFC stabilized numerical framework that should convince and might even inspire the interested reader.

Open analytical problems As we have partially motivated our development and detailed analysis of numerical schemes for solving chemotaxis-driven PDEs by reflecting some open problems that (theoretical) analysts and modelers still face, we believe that our numerical methodologies can lead to detailed insights and hence further the understanding of blow-up, steady states and patterns for chemotaxis models. Particularly with the proposed algorithmic improvements in terms of adaptive discretization techniques (in time and space), approximations of the solution's behavior near the blow-up time, the faster (and still accurate) simulation of steady states and the high resolution of spatially local agglomerations in the context of evolving patterns are three particular future objectives. Of course a tight collaboration with (theoretical) analysts is mandatory in order to keep track of the theoretical relevance of open questions.

Multiple chemical agents and species At the most detailed scale, the (intracellular) chemical signaling pathways that underlie the modeled chemotaxis process actually comprises an enormous network of chemical triggers, receptors and several mediators, rf. e.g., [3, Chapter 15] or the short review [99], that can hardly be modeled with a set of only two equations. There are certain models that incorporate a third or fourth PDE in order to model another entity that may not be directly part of the chemotaxis signaling cascade, but is necessary nevertheless, e.g., a nutrient (for bacterial propagation in a semi-solid medium [101]) or an extracellular matrix (for tumor development via angiogenesis [4]). In all of those cases our numerical framework has to be extended by certain additional PDEs. Then the (numerically) most interesting challenge is to identify possible 'trouble-makers' in these new equations and adapt the AFC stabilization correspondingly. In fact, models for *multi-species chemotaxis* (to which class the aforementioned examples generally belong) have recently gained increasing popularity in the (theoretical) analysis community [30, 47, 58, 66, 67, 95, 109] and hence their numerical treatment will soon be one of the mostly challenging tasks in order to understand the entire network of interactions. Let us formulate a basic multi-species model, that can also be derived by the more general model of Horstmann [47],

$$\begin{aligned}
\partial_t u_i &= d_{u_i}\Delta u_i + \nabla\cdot\left[\left(\sum_{l=1,l\neq i}^{n} u_i k_{i,l}\nabla u_l\right) - \left(\sum_{j=1}^{m} u_i\chi_{i,j}\nabla v_j\right)\right], \\
\partial_t v_j &= d_{v_j}\Delta v_j - \sum_{k=1}^{m}\alpha_{k,j}v_k + \sum_{k=1}^{n}\beta_{k,j}u_k,
\end{aligned}$$

where $i = 1,\dots,n$ and $j = 1,\dots,m$ denote the particular species and chemical agent under consideration, respectively, which share one common habitat. The coefficients $k_{i,l}$ describe the direct attracting or repelling effect of species l on species k, whereas all other coefficients, $d_{u_i},\chi_{i,j},d_{v_j},\alpha_{k,j}$ and $\beta_{k,j}$, are self-explaining when considering the usual model of chemotaxis correspondingly. Some interesting model paradigms have been mentioned and investigated in the already provided literature. For instance Wolansky already formulated a first reasonable mathematical definition for the *absence of conflicts* and *presence of conflicts* between two species, say k and l. For simplicity let us assume $k_{\cdot,\cdot} = 0$. We define $\lambda_{k,l} = \sum_{j=1}^{m}\chi_{k,j}\beta_{l,j}$ and interpret $\lambda_{k,l} > 0 (< 0)$ as the indirect effect that species k is attracted (repelled) by species l. Particularly, remark that in general $\lambda_{k,l} \neq \lambda_{l,k}$. With this in mind, Wolansky proposed the following definition (modified from [109])

> "*The situation where $\lambda_{k,l}$ and $\lambda_{l,k}$ are opposite in sign is of a particular interest.*
> *We denote this case as a 'conflict of interests' between the k and l species.*"

Where Wolansky studied the conflict-free case, Horstmann [47] extended this definition and even provided some preliminary remarks on the situation of conflicts.

It is this latter situation that certainly gives rise to many interesting questions. Even for a small number of species/chemicals, the complexity of their possible interactions makes it very much unhandy for a theoretical analyst to develop results concerning interesting phenomena such as (non-)uniform co-existence or extinction of species. This problem can even be more emphasized when considering multi dimensions. In fact, up to the current date, most of the analytical results are limited to the 1D (or uniform spatial distributions of species/chemicals) or, exceptionally, 2D case.

Based on our numerical framework that we presented in this thesis, Sokolov[1] already initiated some preliminary numerical studies for multi-species models which were communicated with Horstmann. The yet unpublished results reveal the basic applicability of AFC for these systems. However, more elaborated numerical techniques are crucially required in order to cope with the extended complexity of these systems. This addresses both computational concerns and the challenge of stabilization. Whereas in the former context, the storage-management for the additional species/chemicals is the most striking point, the latter concern must take into account new positive feedback interactions that give rise to advancing the AFC technique by means of stabilizing additional discretized terms, e.g., consider the coefficients $k_{i,l}$ for the direct attraction of two species.

Chemotaxis on surfaces When considering situations where motile cells live on a certain densely packed tissue that cannot be modeled via a flat surface, we naturally end up with a chemotaxis system defined on a non-trivial surface, e.g., on an ellipsoid. As being among the first

[1] Dr. Andriy Sokolov is a senior colleague at the chair of Applied Mathematics, LS3, TU Dortmund.

who explored chemotaxis-driven PDEs on surfaces, the reader is kindly referred to the work of Landsberg *et al.* [64] and Elliott *et al.* [27]. Particularly because of the algebraic character of the AFC concept, the stabilization of chemotaxis PDEs on a surface seems to be a feasible task, albeit the derivation of a proper FE discretization for an arbitrary surface is all but a trivial challenge. Nonetheless, Sokolov *et al.* [93] presented a first approach in this direction by applying an AFC stabilization technique on the minimal model of chemotaxis and the kinetic variant both defined on either a sphere or an ellipsoid. These surfaces were modeled implicitly via the *level set method*, i.e., the surfaces were given by the set of roots of a level set function. Alternative FE approaches employ certain parametrization methods or a so-called *surface finite element method* [25]. A future goal will be to embed chemotaxis PDEs on surfaces in a model for a regular domain. In this context a level set approach as employed by Sokolov *et al.* is favorable since no explicit (spatial) discretization of the surface is required. Moreover, very recent modeling frameworks consider evolving surfaces which certainly augment the numerical complexity but also gives rise to a more detailed model and enriches possible applications. Consider for example bacterial development on a growing cell-tissue, where the growth can either be uniform or can even interact with the bacterial development on its surface, e.g., growth is inhibited/promoted at a high/low concentration of bacteria.

A

Linear stability analysis

In this appendix we will show how a basic linear stability analysis can be conducted. A linear stability analysis is one of the major analytical tools for examining the dynamics of an underlying PDE model. Exemplary we will employ the linear stability analysis on the minimal model in dimensionless form (3.2.4) in 1D,

$$\begin{cases} \partial_t u(t,x) &=& \nabla \cdot \big(d \nabla u(t,x) - u(t,x)\chi \nabla v(t,x)\big)\,, & \text{for } (t,x) \in I \times [0,l]\,, \\ \partial_t v(t,x) &=& \Delta v(t,x) + u(t,x) - v(t,x)\,, & \text{for } (t,x) \in I \times [0,l]\,. \end{cases} \tag{A.1}$$

The goal is to identify the possibilities of a solution to form heterogeneous steady states when starting with an initially slightly perturbed homogeneous profile. We remark that the linear analysis only captures first-order behavior of the solution. Nevertheless the linear analysis provides conditions for the emerging of possible heterogeneous steady states.

Linearization of the 1D minimal model

The first step will be to linearize the system (A.1) around a homogeneous steady state, say (u^*, v^*). To this end we derive the corresponding jacobian of the right-hand side of (A.1). We can transform our equation into

$$\partial_t \begin{bmatrix} u \\ v \end{bmatrix} = \mathbf{F}(u,v), \tag{A.2}$$

where $\mathbf{F}(u,v) = \big(F_1(u,v), F_2(u,v)\big)^T$ denotes the right-hand side of (A.1). The aforementioned jacobian (of F) now reads

$$\begin{cases} \partial_u F_1(u,v) &= d\nabla^2 \bullet - \chi \nabla \cdot (\bullet \nabla v)\,, \\ \partial_v F_1(u,v) &= -\chi \nabla \cdot (u \nabla \bullet)\,, \\ \partial_u F_2(u,v) &= I\,, \\ \partial_v F_2(u,v) &= \nabla^2 \bullet - I\,. \end{cases} \tag{A.3}$$

After evaluating this jacobian at (u^*, v^*) – note that according to (A.1) the steady state yields $v^* = u^*$ – we end up with the following linearized system

$$\begin{cases} \partial_t \hat{u} &= d\nabla^2 \hat{u} - \chi u^* \nabla^2 \hat{v}\,, \\ \partial_t \hat{v} &= \nabla^2 \hat{v} + \hat{u} - \hat{v}\,. \end{cases} \tag{A.4}$$

Herein the notation $(\hat{\cdot})$ stems from the linearization about the steady state (u^*, v^*), i.e.,

$$\begin{aligned} u &= u^* + \varepsilon \hat{u} + \mathcal{O}(\varepsilon^2)\,, \\ v &= v^* + \varepsilon \hat{v} + \mathcal{O}(\varepsilon^2)\,. \end{aligned}$$

Let us remark that the linearized system (A.4) introduces a further parameter u^* which was not present in the original non-dimensionalized model (A.1). In the next paragraph we will recognize its role in the stability analysis.

Stability analysis of the 1D minimal model

After we have transformed the governing nonlinear model (A.1) into an adequate linear model (A.4) we are now able to apply standard linear stability analysis. This will provide us with sufficient conditions for unstable homogeneous steady states. The basic concept can be found in standard PDE analysis books. In the background of biological motivated PDEs we refer the interested reader to, e.g., chapter 14 in [78]. In the following we will recall the main results for the minimal model which have already been postulated by Keller and Segel [52] themselves and continued by Nanjundiah [83].

The main concept can be formulated by looking for solutions of the governing PDE via the method of separation of (temporal and spatial) variables, i.e., $\mathbf{w}(t,\mathbf{x}) = \boldsymbol{\xi}(t)\,\boldsymbol{\zeta}(\mathbf{x})$, where $\mathbf{w}$ denotes the solution vector $\mathbf{w} = (u,v)$. When following the argumentations in the literature about the scalar 1D spatial-dependent solution $\zeta(x)$, we will encounter the following auxiliary PDE (also referred to as *spatial eigenvalue problem*)

$$\nabla^2 \zeta + k^2 \zeta \;=\; 0\,, \tag{A.5}$$

where k represents a certain eigenvalue, which is also referred to as *wavenumber*. Together with Neumann boundary conditions this PDE is responsible for the spatial heterogeneous solution behavior. From corresponding literature we learn that general solutions for (A.5) in one dimension take the form

$$\zeta(x) \;=\; A_1\, cos(kx) + A_2\, sin(kx)\,. \tag{A.6}$$

Herein A_1 and A_2 are constant coefficients. For Neumann boundary conditions and a domain $\Omega = (0,l)$ we can deduce $A_2 = 0$, $k = n\pi/l$ and hence (for a given k) the space of eigenfunctions is spanned by $\{cos(kx)\}_{k\geq 1}$.
When focusing on the temporal-dependent solution, $\xi(t)$, we use the standard growth/decay ansatz $\xi(t) = A_3\, exp(\lambda t)$. Putting everything together (and introduce new constants) we end up with the following representation of a general solution of shape

$$w(t,x) \;=\; \sum_n B_n e^{\lambda t} cos(n\pi x/l)\,, \tag{A.7}$$

where the sum stems from the different admissible n, determining the *wave frequencies*. Keeping the above derivations in mind, we will now consider our original governing PDE (A.4). To begin with, we rewrite (A.4) in terms of

$$\partial_t \mathbf{w} \;=\; (\mathfrak{D}\nabla^2 + \mathfrak{A})\,\mathbf{w}\,, \tag{A.8}$$

where ∇^2 is meant component-wise, $\mathbf{w} = (u,v)$ and $\mathfrak{D}$, $\mathfrak{A}$ are defined as

$$\begin{aligned} \mathfrak{D} &= \begin{bmatrix} d & -\chi u^* \\ 0 & 1 \end{bmatrix}, \\ \mathfrak{A} &= \begin{bmatrix} 0 & 0 \\ 1 & -1 \end{bmatrix}. \end{aligned}$$

Using the solution (A.7) we find ($\mathbf{k}$ being a vector of admissible wave modes for the solution (u,v))

$$\begin{aligned} \lambda\mathbf{w} &= -\mathfrak{D}|\mathbf{k}|^2\mathbf{w} + \mathfrak{A}\mathbf{w} \\ \Rightarrow \quad \left[\lambda I + |\mathbf{k}|^2\mathfrak{D} - \mathfrak{A}\right] \begin{bmatrix} B_n^1 \\ B_n^2 \end{bmatrix} &\overset{\text{coeff.cmp.}}{=} 0. \end{aligned}$$

In other words, the stability can be studied from the determinant of the left-hand side matrix,

$$\left|\lambda I + |\mathbf{k}|^2\mathfrak{D} - \mathfrak{A}\right| \;=\; \lambda^2 + \left(|\mathbf{k}|^2 + 1 + d\,|\mathbf{k}|^2\right)\lambda + d\,|\mathbf{k}|^2 + d\,|\mathbf{k}|^4 - |\mathbf{k}|^2\chi u^*\,.$$

Hence its two roots read

$$\lambda_\pm \;=\; \frac{1}{2}\left(-|\mathbf{k}|^2 - 1 - d\,|\mathbf{k}|^2 \pm \sqrt{|\mathbf{k}|^2 + 1 + d\,|\mathbf{k}|^2 - 4d\,|\mathbf{k}|^2 + 4\,|\mathbf{k}|^2\chi u^* - 4d\,|\mathbf{k}|^4}\right).$$

Since the term in front of the square root is strictly negative, the only possibility of λ_+ to get a positive real part is that the term under the square root becomes sufficiently large, i.e., $4d\,|\mathbf{k}|^2 - 4\,|\mathbf{k}|^2\chi u^* + 4dk^4 < 0$. Basic calculus reveals that a necessary and sufficient condition for this to hold reads

$$(1 + |\mathbf{k}|^2)\,d \;<\; \chi u^*\,. \tag{A.9}$$

This condition can also be illustrated by a dispersion relation diagram as a function $Re\big(\lambda(k)\big)$ for certain parameter settings for d,χ and u^*, see Figure A.1. In the figure we recognize that more wave modes become unstable as the value of χ is increased. The admissible modes are undoubtedly of utmost interest when concerning the shape of possible heterogeneous steady states. From the above calculations we can deduce that admissible wave modes must satisfy

$$0 < |\mathbf{k}| < \chi u^*/d - 1\,.$$

For the parameters in Figure A.1 this condition simplifies to $0 < |\mathbf{k}| < \chi - 1$.

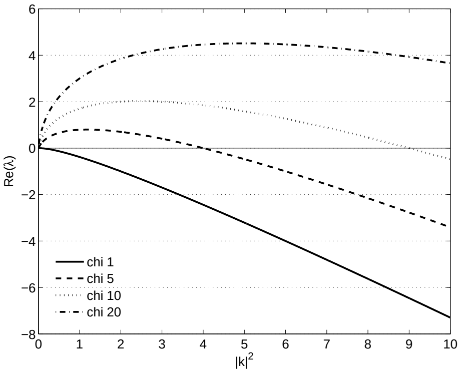
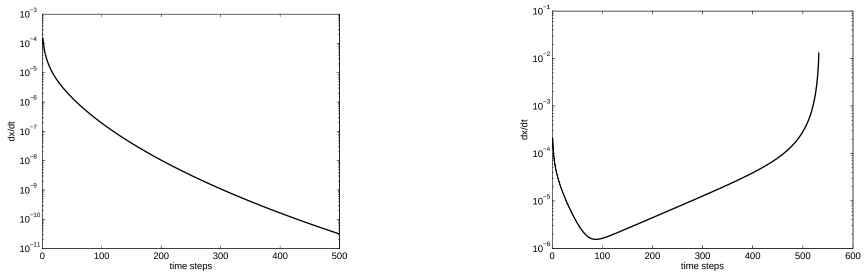


Figure A.1: Dispersion relation for the condition (A.9) for different values of $\chi = 1, 5, 10, 20$ and fixed parameters $d = 1, u_0 = 1$. The plot illustrates $Re(\lambda)$ as a function of the wave mode $|\mathbf{k}|^2$.

Discussions

In the presence of multiple admissible modes let us remark that our linear analysis is not able to identify the most dominant. Furthermore, the condition (A.9) only indicates that homogeneous equilibria can be unstable. It does, however, not provide certain shapes of inhomogeneous equilibria. Indeed, note that condition (A.9) gives rise to an exponential (unbounded) growth (in time) of the solution of type $\mathbf{w} \approx B \exp(\lambda(k)t) \cos(kx)$. Hence for (inhomogeneous) equilibria to appear, this linear approximation fails. We might expect a growing mode to be bounded by the nonlinearity of the original PDE and eventually obtain a steady solution similar to a corresponding cosine mode. However, following the calculations of Keller and Segel [52] and Nanjundiah [83] initial fluctuations of homogeneous equilibria lead either to homogeneous solutions with $u = ||u_0||_{L^1} = v$ or δ-singularities. A comprehensive discussion about possible non-trivial (inhomogeneous) steady states exceeds the scope of this appendix. Therefore we refer the interested reader to the corresponding paragraphs in the literature, e.g., [46], [79] or [100]. Where Horstmann nicely recapitulates the historical accomplishments beginning with the minimal model in [46], the latter two references consider a nonlinear analysis for a particular chemotaxis model, cf. Section 3.3.

In Figure A.2 we sketch the evolution of the steady state ($\partial\mathbf{w}/\partial t \doteq (\mathbf{w}^n - \mathbf{w}^{n-1})/\delta t$) against the time steps, once for a stable setting, $\chi = 1 = d, u_0 < 1$, and once for an instable setting, $\chi = 1 = d, u_0 > 1$. The solution remains bounded and eventually approaches the homogeneous equilibrium $u^* = ||u_0||_{L^1} = v^*$ in the former case while it excites a blowing-up solution in the latter.



(a) $u_0 < 1$

(b) $u_0 > 1$

Figure A.2: Evolution of an initially perturbed homogeneous equilibrium. **Left:** Initial value below critical level. **Right:** Initial value above critical level.

B
Lyapunov functions

When studying analytic results for complex PDE-systems like the classical Keller-Segel(KS) model (e.g., existence or uniqueness of solutions as well as long-time behavior), one encounters the theory of Lyapunov functions (beside a variety of certain inequalities). Lyapunov functions are closely connected to the field of dynamical systems. Therefore, let us try to briefly recall this theory following Grüne [37].

Definition B.1 *Let $\mathbb{T}$ be a time line (in the following we assume $\mathbb{T} = \mathbb{R}$) and X a metric space. A dynamical system is a continuous mapping $\Phi : \mathbb{T} \times X \to X$ satisfying the following two properties*

$$\begin{aligned} \Phi(0, \mathbf{x}) &= \mathbf{x}, \quad \textit{for all } \mathbf{x} \in X\,, \\ \Phi(t+s, \mathbf{x}) &= \Phi\Big(t, \Phi(s, \mathbf{x})\Big), \ \textit{for all } t, s \in \mathbb{T}\,. \end{aligned}$$

For a fixed $\mathbf{x}_0 \in X$ (the so-called initial-condition) we call $\Phi(t, \mathbf{x}_0)$ a solution or trajectory.

In the context of ODEs a dynamical system can be simply considered as the solution of the governing ODE. Let us consider an autonomous ODE of the form

$$\dot{\mathbf{x}}(t) = f\Big(\mathbf{x}(t)\Big), \tag{B.1}$$

subject to the initial condition

$$\mathbf{x}(0) = \mathbf{x}_0\,, \tag{B.2}$$

wherein $\mathbf{x} : \mathbb{R} \to \mathbb{R}^{\dim}$, $\mathbf{x}_0 \in \mathbb{R}^{\dim}$ and $f : \mathbb{R}^{\dim} \to \mathbb{R}^{\dim}$. If these functions are 'well-suited', i.e., a unique solution $\mathbf{x}(t; \mathbf{x}_0)$ exists, then we can call $\Phi(t, \mathbf{x}_0) := \mathbf{x}(t; \mathbf{x}_0)$ a *dynamical system.*

When studying dynamical systems with a trajectory $\Phi(t, \mathbf{x})$ we might be interested in stable equilibriums $\Phi(t, \mathbf{x}^*) = \mathbf{x}^* \quad$ for all $t \in \mathbb{R}$, i.e., we are interested in *regions of stability $U_{\mathbf{x}^*}$* or even *regions of asymptotic stability $U'_{\mathbf{x}^*}$*. These sets can be described as

$$\begin{aligned} \mathbf{x} \in U_{\mathbf{x}^*} \quad &\Rightarrow \quad \Phi(t, \mathbf{x}) \in U_{\mathbf{x}^*}\,, \\ \mathbf{x} \in U'_{x^*} \quad &\Rightarrow \quad \Phi(t, \mathbf{x}) \overset{t\to\infty}{\to} \mathbf{x}^*\,. \end{aligned}$$

In a more informal way these two properties can be viewed as

1. when perturbing the equilibrium the trajectories remain in a certain neighborhood (of the equilibrium) for all instances of time;
2. when perturbing the equilibrium the trajectories converge (back) to the equilibrium.

As an extension to regions of asymptotic stability, we can particularly consider so-called *attractors*. For its definition, we briefly introduce *invariant sets* in the context of trajectories

Definition B.2 (Invariant set) *Let $\Phi(t,\mathbf{x})$ be a trajectory of a dynamical system. A subset $D \subseteq X$ is invariant, if*

$$\Phi(t,D) \quad = \quad D\,, \qquad \textit{for all } t \in \mathbb{R}\,.$$

Informal this means that if we initiate trajectories in D, they will remain in this domain and they cover all of D.
Now we can provide the definition of attractors.

Definition B.3 (Attractor) *Given a dynamical system, a region of asymptotic stability $A \subseteq X$ is an attractor, if this set is furthermore invariant. Attractors can also be characterized as the minimal region of asymptotic stability.*

In the following we will always assume, that the considered dynamical system arises from a PDE, cf. the corresponding definitions for ODEs (B.1) and (B.2).

Although the above properties are easy to understand theoretically, they are unhandy to proof for a given dynamical system or, in our particular case, for a PDE. In this context Lyapunov functions provide a good tool. Lyapunov functions can be viewed as energy functionals, which correspond to the underlying system. The construction of a suitable Lyapunov function is non-trivial in most cases, indeed there is no fixed ‘manual’ which works for all physical, chemical or biological systems. As a guideline there are two properties which have to be satisfied by Lyapunov functions.

Definition B.4 (Lyapunov function, PDE version) *Given a n-dimensional autonomous vector valued PDE*

$$\dot{\mathbf{w}}(t,\mathbf{x}) \quad = \quad f(\mathbf{w}(t,\mathbf{x}))\,, \qquad \textit{for } (t,\mathbf{x}) \in I \times \Omega\,,$$

complemented by suitable boundary conditions and initial conditions $\mathbf{w}(0,\mathbf{x}) = \mathbf{w}_0(\mathbf{x})$ for $\mathbf{x} \in \Omega$ with a fixed point $\mathbf{w}^*$, e.g., $f(\mathbf{w}^*) = 0$. Let this PDE be well-defined, in a way that for all $\mathbf{w}_0$ there exists a unique solution $\mathbf{w}(t,\mathbf{x};\mathbf{w}_0) \in C^1(I;W)$, where W is some sufficiently smooth space. A Lyapunov function for this PDE is a continuous differentiable functional $E : W \to \mathbb{R}$ which satisfies

1. *E is locally positive definite, i.e., $E(\mathbf{w}) > 0$ for all $\mathbf{w} \in \mathcal{B}^{\varepsilon}_{\mathbf{w}^*} \setminus \{\mathbf{w}^*\}$ and $E(\mathbf{w}^*) = 0$,*
2. *$\dot{E}$ is locally negative definite, e.g., $\dot{E}(\mathbf{w}) < 0$ for all $\mathbf{w} \in \mathcal{B}^{\varepsilon}_{\mathbf{w}^*} \setminus \{\mathbf{w}^*\}$.*

Herein $\dot{E}$ denotes the derivative of E along the trajectories of the given PDE and $\mathcal{B}^{\varepsilon}_{\mathbf{w}^*} = \{\mathbf{w} : ||\mathbf{w} - \mathbf{w}^*|| < \varepsilon\}$ is some neighborhood around the equilibrium.

The focus of Lyapunov functions is to prove existence of attractors (for a given PDE). That is, whenever we can find a Lyapunov function, we have a practical tool to try to characterize the stable behavior of solutions. For instance, starting from an equilibrium, we may be interested in the critical mass-perturbation (of the solution) in a way, that the perturbed solution is not anymore in the region of stability. Thus we can observe a different temporal development of the solution.

In the presence of our chemotaxis-driven PDEs we would like to gain insights into the bounded evolution of solutions. In particular one main challenge is to classify initial conditions which lead to either a ‘stable’ solution (steady-state or norm-bounded solutions) or a blow-up in finite or infinite time. In terms of dynamical systems we like to ‘control’ the trajectories obtained from the governing chemotaxis PDE. For example in the 2D case of our classical minimal model (2.2.1), we already obtained certain critical mass-perturbation. Given a suitable domain and boundary conditions, if the L_1-norm of the initial cells exceeds a limit, then the solution exits the region of stability and the L_∞-norm of the cells becomes unbounded (as time evolves).

The basic idea to classify the solutions’ behavior by means of Lyapunov functions can be captured as followed:

Given a suitable Lyapunov function E for the governing PDE, we are able to estimate certain norms of the solution to obtain upper (and lower) bounds. However, finding a proper functional is a difficult task in general. To this end certain physical properties like conservation-laws (energy, mass, momentum) can lead to first attempts. To obtain L^2 estimates of the solution we certainly have to invest more sophisticated analysis. Remark that in many cases the L^1 norm of the cells should remain constant along the time line, because of the mass conservation $||u(t,\cdot)||_{L^1} = ||u_0||_{L^1}$ for all $t > 0$.

As briefly sketched in Theorem 2.2 in Section 2.2 there are already certain results for Lyapunov functions in the context of our governing minimal model of chemotaxis (2.2.1):

Yagi [110] already proved that a blow-up of the $H^{1+\varepsilon}$ norm occurs whenever $T_{\max} < \infty$. Herein $[0, T_{\max})$ denotes the maximal interval of existence of the solution (u, v). He even stated that $T_{\max} < \infty$ implies the blowing-up of the L^p norm for $1 < p \leq \infty$. On the other hand Gajewski and Zacharias [33] showed that the unboundedness of the Lyapunov function leads to a blow-up of the L^2 norm of the cell density. On the other hand, in many cases that boundedness of the Lyapunov function (as time evolves) implies the boundedness of certain norms of the solution. The imminent task of finding conditions which lead to either $T_{\max} < \infty$ or the unboundedness of the Lyapunov function, is still vitally discussed in the community.

C

The law of mass action and the Michaelis-Menten kinetics

This appendix provides an overview of the basic knowledge of understanding the modeling of common reaction terms and more sophisticated kinetics employed particularly in the presence of chemotaxis PDEs, see Section 3. When modeling a (biological) system one of the first steps tackles the mapping of observable relations among underlying entities (cells, enzymes, chemicals or even single atoms) in terms of a stoichiometric equation such as $A+B \rightleftharpoons C$. This diagram can now be 'translated' to ODEs or under certain quasi steady-state assumptions even to expressions of the concentration of underlying entities. The resulting equations can subsequently be used to eventually derive a targeting model describing the real system precisely enough (up to certain modeling preliminaries). For the remainder of this appendix the encouraged reader might want to check basic literature, e.g., [57, 92, 94], which we will follow in the proceeding paragraphs. Besides detailed derivations, therein the authors also discuss the theoretical assumptions (e.g., constant temperature, pH) and other preliminaries which we will skip for reasons of clarity.

To allow a smooth introduction, the first section of this appendix is devoted to a brief classification of chemical reactions which are considered in our context. The second section will demonstrate what the law of mass action postulates and how it can be applied to obtain a corresponding ODE. In other words, we will see how reaction diagrams can be 'translated'. The third section will offer a brief survey about a particular theory which was originally established for enzyme kinetics in 1913, namely the Michaelis-Menten theory. We will sketch how this theory can be applied to chemotaxis models. A concluding fourth section will point out some remarks about the Michaelis-Menten theory which give rise to certain revisions.

A simple classification of chemical reactions

Here we will concern about (biochemical) reactions and its inferred mathematical model in terms of an ODE. Having in mind our focus of this thesis, we restrict ourselves to two main classes of reactions which are formulated as stoichiometric equations.

A first class describes a simple (ir-)reversible chemical reaction of certain *order* and *molecularity*, i.e.,

$$aA+bB \quad \leftrightharpoons \quad cC+dD\,. \tag{C.1}$$

Therein A,B,C and D denote the underlying reactants/products and a,b,c,d are stoichiometric coefficients. The bidirectional arrow describes the reversibility, whereas a simple arrow denotes irreversibility. A simple example would be the irreversible formation of water out of hydrogen and oxygen given by

$$2H_2+O_2 \quad \to \quad 2H_2O\,. \tag{C.2}$$

A reaction of kind (C.1) is called *elementary* if the indicated products are formed directly from the reactants, e.g., via a direct collision at the molecular level (which is not the case in (C.2)). Furthermore let us require that by virtue of mass conservation, the stoichiometric coefficients of an elementary reaction of kind (C.1) yield

$$a+b \quad = \quad c+d\,,$$

namely (loosely speaking) the kinetics are consistent with stoichiometry, cf. [92]. Note however that for abstract elementary reactions this is not always fulfilled in some books, simply because the abstract reactants/products represent a molecules of different complexity and weight. Exemplary in [94] we find the elementary reaction

$$2I+H_2 \quad \to \quad H_2I+I\,,$$

which can be casted in abstract form as

$$2A+B \quad \to \quad C+A\,,$$

with $A=I, B=H_2$ and $C=H_2I$. Obviously the stoichiometric coefficients yield $2+1\neq 1+1$, however reconsidering the definitions of A,B,C mass conversation is still preserved. For the remainder of this section we only consider elementary reactions, if not explicitly stated different.

A second class of reactions is catalysis. In certain reactions of kind (C.1) another entity influences the reaction which explicitly does not appear in the stoichiometric equation, these complexes are called catalysts. In the context of biochemical reactions these catalysts are commonly particular enzymes. Formally we rewrite such reactions in terms of (C.1) as a two-step reaction

$$A+B \quad \leftrightharpoons \quad AB \quad \to \quad A+C\,, \tag{C.3}$$

where A is now referred to as a catalyst, e.g., an enzyme, B,C are correspondingly a catalysis reactant and product, respectively, and AB is the catalyst-reactant complex. These reactions are commonly used for biochemical signal pathways and we will return to them in the proceeding section.

The law of mass action

Now that we have the two (in our belief) most interesting classes at hand, we want to discuss the further (mathematical) modeling processing. We still owe the reader a proper definition of the order and molecularity of a certain reaction. To this end let us state the law of mass action.

Originally established by Guldberg and Waage [38] in 1864, the law of mass action states that the rate of a elementary chemical reaction is proportional to the product of active masses. Here, we understand active masses as the number of active reactants, i.e., the stoichiometric coefficients. The molecularity of a reaction is simply the number of different reactants that are involved in the reaction step. Furthermore, the order with respect to one reactant is determined by the index to which the concentration of this reactant is raised in the rate equation. The overall order is simply the summation of all single reactant orders.

Let us consider a simple example of an irreversible chemical reaction of one single reactant:

$$A \quad \rightarrow \quad B$$

This is a first order unimolecular reaction since its reaction rate r reads

$$r \quad = \quad k[A]\,,$$

i.e., the concentration of A, denoted by $[A]$, is only raised to the first power. Here k is a constant corresponding to the law of mass action. Sometimes this rate-constant is also directly depicted in the stoichiometric equation. A second example is a third order bimolecular reaction

$$2A+B \quad \rightarrow \quad 3C\,.$$

Here the reaction rate reads

$$r \quad = \quad k[A]^2[B]\,.$$

For a reversible reaction

$$A \quad \leftrightharpoons \quad B$$

we define the reaction rate as

$$r \quad = \quad k_+[A]-k_-[B]\,,$$

that is, we read the reaction from left to right and sum up the related ‘single reaction rates’: a plus and minus sign contributes to the reaction from left to right and right to left, respectively.

Now that we introduced the law of mass action we turn to the next step towards a derivation for the chemical concentrations (in quasi equilibrium). For simplifications consider the reaction occurring in a closed homogeneous environment, a so-called batch reactor, e.g., a petri dish. This simplifies the general equation of mass balance

$$\text{accumulation} \quad = \quad \text{input} \; - \; \text{output} \; + \; \text{generation by reaction}$$

to

$$\text{accumulation} \quad = \quad \text{generation by reaction}\,.$$

The input and output correspond to (continuous) feed-in and effluence of involved chemicals. A commonly used environment for reaction-studies in the field of biological processes is a chemostat, see Figure C.1. The stirring practically provides homogeneous chemical distributions. Chemostats are commonly used for steady state studies of involved chemicals. The obtained data can be used

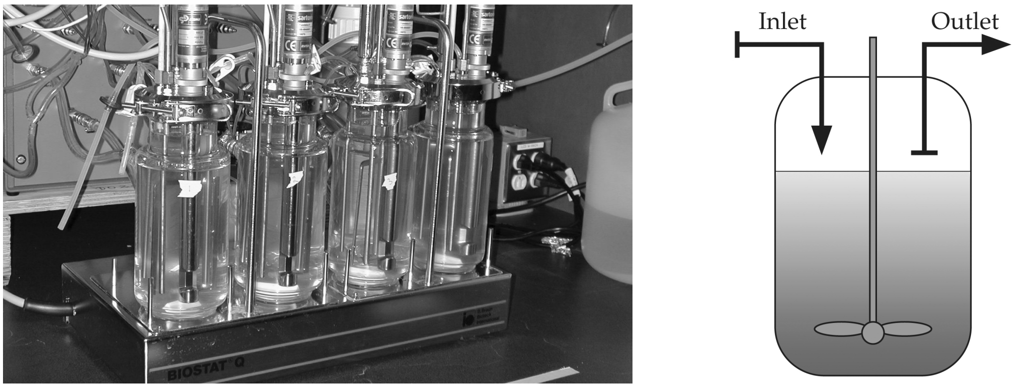


Figure C.1: Chemostats as examples of paradigm bioreactors. Left: A photography of a battery of four chemostats, used with permission from Gregor Fussmann, Fussman Lab, McGill University (http://biology.mcgill.ca/faculty/fussmann/chemostats.html). Right: A technical drawing of a chemostat.

for modeling purposes, e.g., to determine reaction constants. Cutting down the in- and out-flow turns the chemostat into a classical batch reactor.

Under this conditions, a n-th order irreversible reaction of kind

$$A \quad \rightarrow \quad B$$

with reaction rate $r = k\,[A]^n$ yields

$$\partial_t[A] \quad = \quad -k\,[A]^n\,.$$

After integrating and assuming an initial concentration $[A_0]$ we obtain

$$[A](t) \quad = \quad [A_0]\big(1+(n-1)\,k\,[A_0]^{n-1}\,t\big)^{1/(1-n)}\,.$$

Note that for $n = 1$ the upper expression is undefined. In this special case the mass balance reads

$$\partial_t[A] \quad = \quad -k\,[A]$$

and we simply obtain exponential decay

$$[A](t) \quad = \quad [A_0]\,exp(-kt)\,.$$

According to the reaction, the concentration of the product $[B]$ develops correspondingly in an increasing fashion.
When looking at a reversible reaction of kind

$$A \quad \leftrightharpoons \quad B$$

with reaction rate

$$r \quad = \quad k_+[A]-k_-[B]$$

and initial concentrations $[A] = [A_0]$ and $[B] = 0$ at $t = 0$, we finally end up with

$$[A](t) \quad = \quad [A_0]\left(1-\frac{k_+}{k_++k_-}\Big(1-exp\big(-(k_++k_-)t\big)\Big)\right),$$

where $[B](t)$ can be obtained analogue.

The Michaelis-Menten theory

With our biological processes (cf. Section 3) in mind we are particularly interested in biological motivated reactions, e.g., enzyme kinetics as already mentioned before, cf. (C.3). Now we interpret A as *free* enzyme and B,C as reactant chemical and product, respectively. The enzyme-chemical complex AB corresponds to a *bounded* enzyme. In Figure C.2 we sketched a exemplary enzymatic reaction.

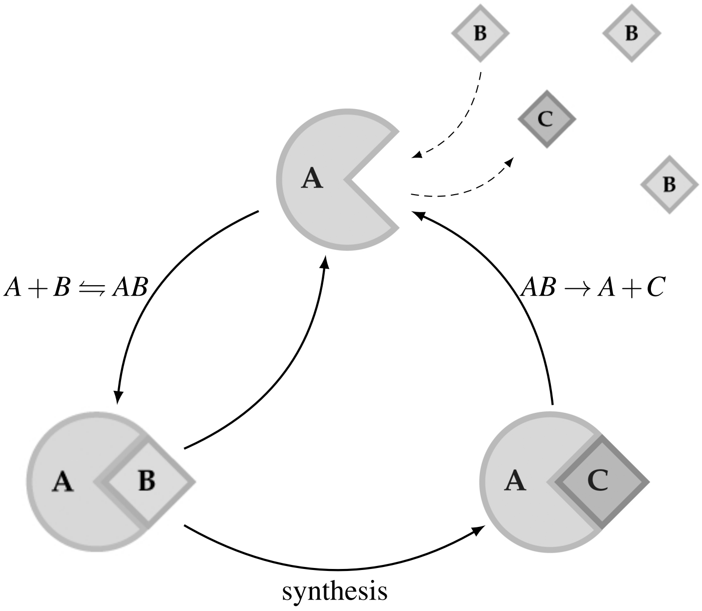


Figure C.2: An exemplary schematic enzyme reaction. Freely diffusing chemicals B can bind to the enzyme A, which results in a reversible complex AB. If this complex remains stable, the synthesis of a product C is initiated. After releasing the synthesized product, the enzyme A is freely available again.

If we study the concentrations of the involved entities at equilibrium with respect to the enzyme-reactant complex $[AB]$ – precisely speaking it is rather a dynamically equilibrium where forward and backward reaction rates are equalized, cf. [57] – then we obtain

$$\begin{aligned} \partial_t[AB] &= k^1_+[A][B] - k^1_-[AB] - k^2_+[AB] \quad = \quad 0 \\ \Rightarrow \quad [AB] &= \frac{k^1_+}{k^1_- + k^2_+}[A][B]\,. \end{aligned}$$

Together with mass conservation for the total enzyme concentration $[A]+[AB]=[A_0]$, with $[A_0]$ being the initial concentration, we write the concentration of bounded enzymes as

$$[AB] \quad = \quad K[A_0]\frac{[B]}{1+K[B]}\,,$$

where $K = k^1_+/(k^1_- + k^2_+)$. Correspondingly, the velocity for the overall reaction (C.3) reads

$$\begin{aligned} \partial_t[C] &= k[AB] \\ &= kK[A_0]\frac{[B]}{1+K[B]}\,. \end{aligned}$$

This, in turn, is exactly what *Michaelis-Menten kinetics* classically state. More often the reaction velocity γ (here $\gamma = \partial_t[C]$) is written in terms of

$$\gamma \;=\; \gamma_{\max}\frac{[B]}{K_M+[B]}\,, \tag{C.4}$$

where $\gamma_{\max} = k[A_0]$ and $K_M = K^{-1}$. These new notations are proposed for reasons of mathematical interpretation of these coefficients. Figure C.3 depicts the reaction velocity (C.4) for different values of K_M, while $\gamma_{\max} = 1$ being fixed. First of all we acknowledge fast increase of reaction velocity at low concentration levels, whereas $\gamma_{\max} = 1$ asymptotically limits the reaction velocity (hence the notation) which is accompanied by a rather inconspicuous increase at high concentration levels. In other words there is a saturation effect of enzyme-reactant binding. We see that K_M toggles the concentration for $\gamma = 0.5 = \gamma_{\max}/2$, viz., this constant is a good yardstick for measuring binding affinity and hence it determines the main characteristic of such reactions. In the literature K_M is often called the *Michaelis Constant*, cf. Steinfeld *et al.* [94].

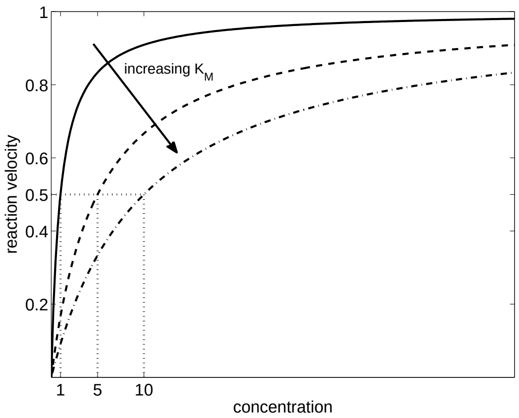


Figure C.3: Plots of the reaction velocity γ (C.4) for three values of $K_M = 1, 5, 10$.

When applying this theory to our model, i.e., to the definition of a suitable chemosensitivity (see Section 3.3), we should complement three further abstractions: (1) the enzyme concentration $[A]$ is much less than the reactant concentration $[B]$, which will be identified by the chemoattractant v, (2) the concentration of bounded enzymes, i.e., the enzyme-reactant-complex $[AB]$, is the direct measurement of chemotaxis potential, (3) the cells are able to detect the local gradient of the chemosensitivity potential. This encourages us to define the chemosensitivity as proposed in Section 3.3.

Furthermore the Michaelis-Menten kinetics provided a descent basis for other reaction terms in our governing chemotaxis model. Despite the fact that Michaelis and Menten introduced their kinetics in the context of enzyme reactions, commonly they are also practically applied when modeling saturation effects, e.g., chemical production/depletion as in Section 3.3.

Discussions

However regarding growth rates, e.g., bacteria proliferation, the Michaelis-Menten kinetics need certain revisions. To begin with, it was Monod who recovered the Michaelis-Menten equation when he was investigating the non-linear relation between growth rates of bacteria cultures and limiting resources in 1949. The Monod model basically reads as the Michaelis-Menten kinetics, modulo different notations

$$\mu \;=\; \mu_{\max}\frac{S}{S+K_S}\,, \tag{C.5}$$

where $\mu, \mu_{\max}, S$ and K_S denote growth rate, maximal rate, limiting resource (e.g., nutrient) and resource concentration at $\mu_{\max}/2$. The main difference between the Michaelis-Menten and the

Monod equation is based in their derivations. While the former was theoretically derived under certain assumptions, the latter was developed from empirical data. Furthermore the rather abstract view of Monod's bacterial growth tends to be more suspicious.
In fact, we recognize that the growth rate is positive whenever $S > 0$. However in nature the limiting resource is already consumed by cells just for maintenance concerns. This effect is commonly included in the so-called *lag phase* of bacterial growth. Monod explicitly did not consider this phase when proposing the equation (C.5), in fact his equation only models the so-called *exponential growth phase*. Hence, when applying such kind of growth rates we assume that all cells are at a "ready-to-proliferate" state. Besides the lag and exponential growth phase, moreover, the entire process of bacteria growth incorporates more distinct phases, as Monod pointed out. Without going into detail (cf. [75] for detailed references), Figure C.4 sketches these phases. To conclude, we have to keep in mind that growth terms, such as the ones introduced in Section 3.3, are most probably not capable to capture the entire growth process, viz., one specific term only models a certain phase of proliferation. We encourage the reader to keep this in mind when interpreting numerical results obtained for the underlying models.

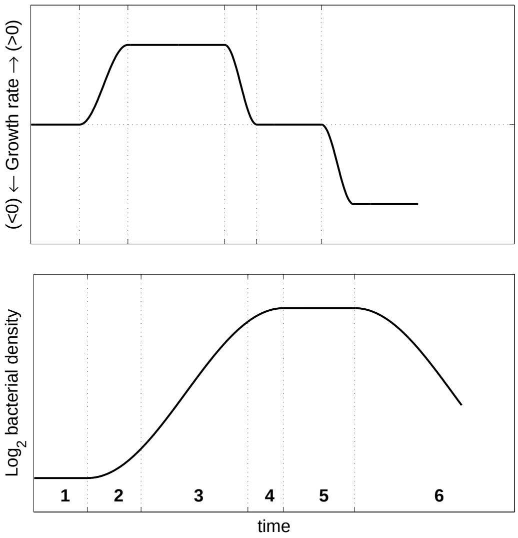


Figure C.4: Sketch of the entire bacteria growth process, cf.[75]: 1) lag phase, 2) acceleration phase, 3) exponential phase, 4) retardation phase, 5) stationary phase, 6) phase of decline.

D

About smoothers

As described in Section 5.3.1 we perform smoothing via the standard Jacobi iteration or via a preconditioned iterative solver, e.g., preconditioned BiCGStab or GMRES. As important multigrid component, a proper smoother should provide the following roughly sketched features in order to be applicable in a appropriate multigrid setup (for more details the reader is referred to [40]):

(S1) It is desirable that the smoother “damps high frequencies (w.r.t. a given mesh) of the solution”.

(S2) The smoother is computationally “efficient”.

(S3) The smoother yields consistency in terms of the exact solution $\mathbf{x}_h^*$, i.e., $\mathcal{S}_h \mathbf{x}_h^* = \mathbf{x}_h^*$.

A typical approach for the construction of such a smoother is a preconditioned iterative solver, where the number of iterations is fixed, say s. For a given system $\mathbf{A}\mathbf{x} = \mathbf{b}$ and an initial guess $\mathbf{x}^0$, the defect correction approach can be written as

$$\mathbf{x}^i = \mathbf{x}^{i-1} + \mathbf{C}^{-1}(\mathbf{b} - \mathbf{A}\mathbf{x}^{i-1}), \quad \text{for } i = 1, \dots, s.$$

It is common practice to reformulate this iteration by means of

$$\begin{aligned} \mathbf{C}\mathbf{y} &= \mathbf{b} - \mathbf{A}\mathbf{x}^{i-1}, \\ \mathbf{x}^i &= \mathbf{x}^{i-1} + \mathbf{y}. \end{aligned}$$

The matrix $\mathbf{C}$ is an appropriate approximation of $\mathbf{A}$. In terms of preconditioning, it can be associated to a preconditioning matrix such that $\mathrm{cond}(\mathbf{C}) < \mathrm{cond}(\mathbf{A})$ and the above system can be rather “easily” solved. Let us provide some well known examples for suitable preconditioning matrices.

Example D.1 *A straightforward attempt to damp the condition of the governing system matrix A is to use scaling, i.e., apply a diagonal matrix $C = diag(d_{11}, \dots, d_{nn})$. The special choice $C = D := diag(a_{11}, \dots, a_{nn})$ leads to the so-called Jacobi-preconditioner (*JAC*).*

*Other preconditioners can be derived from the associated splitting methods. For example the common Symmetric Gauß-Seidel method (*SGS*) leads to the preconditioning matrix $C = (D +$*

*$L)D^{-1}(D+R)$, where L and R denotes the lower and upper triangular part of A, respectively. Caution is advised when using non-symmetric preconditioning matrices such as the one obtained from the simple Gauß-Seidel method (*GS*), $C = (D+L)$. They do not work in the context of positive definiteness requiring algorithms such as the Conjugate Gradient Method (*CG*). Table D.1 provides the preconditioning matrices for well known iterative solver, it was extracted from Meister [71, Chapter 5].*

Method	Preconditioning matrix
Jac	D
GS	$(D+L)$
SOR	$\omega^{-1}(D+\omega L)$
SGS	$(D+L)D^{-1}(D+R)$
SSOR	$[\omega(2-\omega)]^{-1}(D+\omega L)D^{-1}(D+\omega R)$

Table D.1: Overview of the preconditioning matrices associated to the corresponding iterative methods. Extracted from [71].

The following algorithm, Algorithm D.1, sketches the defect correction approach for a simple construction of a smoother.

Algorithm D.1 Smoother variant 1, (preconditioned) defect correction

```
Given the underlying original system $\mathbf{A}\,\mathbf{x} = \mathbf{b}$ with initial solution $\mathbf{x}^0$ and an appropriate matrix $\mathbf{C}$
for $i = 1, \dots, s$ do
    Solve the system $\mathbf{C}\,\mathbf{y} = \tilde{\mathbf{b}} - \tilde{\mathbf{A}}\,\mathbf{x}^{i-1}$
    Update solution $\mathbf{x}^i = \mathbf{x}^{i-1} + \mathbf{y}$
end for
```

More generally a smoother can be constructed with a preconditioned Krylov-space solver such as BiCGStab or GMRES. Given the original system $\mathbf{A}\,\mathbf{x} = \mathbf{b}$, the idea is to transform this system into an equivalent one with a system matrix that is well conditioned before calling the underlying Krylov-space solver. With two invertible matrices $\mathbf{C}_L$ and $\mathbf{C}_R$ the transformed system reads

$$\begin{aligned} \mathbf{C}_L\,\mathbf{A}\,\mathbf{C}_R\,\mathbf{y} &= \mathbf{C}_L\,\mathbf{b}\,, \\ \mathbf{x} &= \mathbf{C}_R\,\mathbf{y}\,. \end{aligned}$$

Hereby, three situations can be distinguished. The case $\mathbf{C}_L \neq \mathbf{I} \neq \mathbf{C}_R$ is referred to as *left-right preconditioning*, whereas $\mathbf{C}_R = \mathbf{I}$ and $\mathbf{C}_L = \mathbf{I}$ are called *left preconditioning* and *right preconditioning*, respectively.

This preliminary transformed system deals as the input system for the underlying iterative solver. Algorithm D.2 depicts the pseudo code framework for a preconditioned solver cascade that can be used as a smoother.

Remark D.1 *The choice of proper preconditioning strategy (left, right or left-right preconditioning) must be seen in the context of the entire solver cascade. We like to stress two main points.*

Algorithm D.2 Smoother variant 2, preconditioned iterative solver cascade

```
Given the underlying original system $\mathbf{A}\,\mathbf{x} = \mathbf{b}$ with initial solution $\mathbf{x}^0$
Precondition the system by setting $\tilde{\mathbf{A}} := \mathbf{C}_L\mathbf{A}\mathbf{C}_R$ and $\tilde{\mathbf{b}} := \mathbf{C}_L\,\mathbf{b}$
for $i = 1,\ldots,s$ do
    Perform one iteration of the underlying solver for $\tilde{\mathbf{A}}\,\mathbf{y} = \tilde{\mathbf{b}} - \tilde{\mathbf{A}}\,\mathbf{x}^{i-1}$    ▷ in our case either BiCGStab or GMRES
    Update solution $\mathbf{x}^i = \mathbf{x}^{i-1} + \mathbf{y}$
end for
Transform the solution $\mathbf{x} = \mathbf{C}_R\mathbf{x}^s$
```

A straightforward implementation of a general preconditioned solver cascade that can be used as a solver by driving the iteration to convergence (no limit s for the number of iterations) can be troublesome. Common termination criteria monitor the residual of the transformed system, i.e., $||\tilde{\mathbf{b}} - \tilde{\mathbf{A}}\,\mathbf{x}^i||$*, rather than the residual of the original system, i.e.,* $||\mathbf{b} - \mathbf{A}\,\mathbf{x}^i||$*. This problem does not emerge for a pure right preconditioning. For more details the reader is referred to [5, 6].*

If we use a one-sided preconditioning, namely, either a right or a left preconditioning, an underlying iterative solver that requires symmetry of the iteration matrix is not directly applicable, e.g., CG*. It is therefore common practice to use full left right preconditioning in these cases. To this end we can consider a symmetric matrix of the form* $\mathbf{C} = \mathbf{E}\mathbf{E}^T$ *and set* $\mathbf{C}_L = \mathbf{E}^{-1}$ *and* $\mathbf{C}_R = \mathbf{E}^{-T}$*. The resulting iteration matrix yields symmetry and hence, solver that require symmetry are applicable.*

Bibliography

[1] J. Adler. Chemotaxis in bacteria. *Science*, 153(3737):708–716, 1966.

[2] M. Aida, T. Tsujikawa, M. Efendiev, A. Yagi, and M. Mimura. Lower estimate of the attractor dimension for a chemotaxis growth system. *Journal of the London Mathematical Society*, 74:453–474, 2006.

[3] B. Alberts, A. Johnson, J. Lewis, M. Raff, K. Roberts, and P. Walter. *Molecular biology of the cell*, volume 54. Garland Press, 2008.

[4] A.R. Anderson and M.A. Chaplain. Continuous and discrete mathematical models of tumor–induced angiogenesis. *Bulletin of Mathematical Biology*, 60(5):857–899, 1998.

[5] O. Axelsson and V.A. Barker. *Finite element solution of boundary value problems*, volume 35 of *Classics in Applied Mathematics*. Society for Industrial and Applied Mathematics, 2001.

[6] R. Barrett, M. Berry, T. F. Chan, J. Demmel, J. Donato, J. Dongarra, V. Eijkhout, R. Pozo, C. Romine, and H. Van der Vorst. *Templates for the solution of linear systems: Building blocks for iterative methods, 2nd edition*. Society for Industrial and Applied Mathematics, Philadelphia, PA, 1994.

[7] G. Bencheva. Computer modelling of haematopoietic stem cells migration. *Computers and Mathematics with Applications*, 64(3):337–349, 2012.

[8] M. Benzi and G.H. Golub. A preconditioner for generalized saddle point problems. *Society for Industrial and Applied Mathematics: Journal on Matrix Analysis and Applications*, 26 (3):20–41, 2004.

[9] P. Biler. Local and global solvability of some parabolic systems modelling chemotaxis. *Advances in Mathematical Sciences and Applications*, 8(2):715–743, 1998.

[10] M. Bittl and D. Kuzmin. An *hp*–adaptive flux–corrected transport algorithm for continuous finite elements. *Computing*, pages 1–22, 2012.

[11] J.T. Bonner. *The cellular slime molds*. Investigations in the biological sciences. Princeton University Press, 1959.

[12] D. Braess. *Finite elements. Theory, fast solvers and applications in elasticity theory. (Finite Elemente. Theorie, schnelle Löser und Anwendungen in der Elastizitätstheorie.) 4th revised and extended ed.* Springer Berlin, 2007.

[13] A.N. Brooks and T.J.R. Hughes. Streamline upwind/Petrov–Galerkin formulations for convection dominated flows with particular emphasis on the incompressible navier-stokes equations. *Computer Methods in Applied Mechanics and Engineering*, pages 199–259, 1990.

[14] E.O. Budrene and H.C. Berg. Complex patterns formed by motile cells of Escherichia coli. *Nature*, 349(6310):630–633, 1991.

[15] C. Cattaneo. Sulla conduzione de calore. *Atti del Seminario Matematico e Fisico dell' Università di Modena*, 3:83–101, 1948.

[16] A. Chertock and A. Kurganov. A second–order positivity preserving central–upwind scheme for chemotaxis and haptotaxis models. *Numerische Mathematik*, 111(2):169–205, 2008.

[17] S. Childress and J.K. Percus. Nonlinear aspects of chemotaxis. *Mathematical Biosciences*, 56:217–237, 1981.

[18] J.L. Christensen, D.E. Wright, A.J. Wagers, and I.L. Weissman. Circulation and chemotaxis of fetal hematopoietic stem cells. *Public Library of Science, Biology*, 2(3):e75, 2004.

[19] J.E. Cohen. Mathematics is biology's next microscope, only better; biology is mathematics' next physics, only better. *Public Library of Science, Biology*, 2(12):e439, 2004.

[20] B.M. DeBlois. Linearizing convection terms in the Navier–Stokes equations. *Computer Methods in Applied Mechanics and Engineering*, 143(3):289–297, 1997.

[21] R.S. Dembo, S.C. Eisenstat, and T. Steihaug. Inexact Newton methods. *Society for Industrial and Applied Mathematics: Journal on Numerical Analysis*, 19:400–408, 1982.

[22] P. Deuflhard. *Newton Methods for nonlinear problems. Affine invariance and adaptive algorithms*. Springer Series in Computational Mathematics. Springer, 2006.

[23] J.I. Diaz and T. Nagai. Symmetrization in a parabolic–elliptic system related to chemotaxis. *Advances in Mathematical Sciences and Applications*, 5:659–680, 1995.

[24] Y. Dolak and T. Hillen. Cattaneo models for chemosensitive movement: numerical solution and pattern formation. *Journal of Mathematical Biology*, 46(5):461–78, 2003.

[25] G. Dziuk and C.M. Elliott. Surface finite elements for parabolic equations. *Journal of Computational Mathematics*, 25(4):385, 2007.

[26] S.C. Eisenstat and H.F. Walker. Choosing the forcing terms in an inexact Newton method. *Society for Industrial and Applied Mathematics: Journal of Scientific Computing*, 17:16–32, 1994.

[27] C.M. Elliott, B. Stinner, and C. Venkataraman. Modelling cell motility and chemotaxis with evolving surface finite elements. *Journal of The Royal Society Interface*, 9(76):3027–3044, 2012.

[28] Y. Epshteyn. Upwind–difference potentials method for Patlak–Keller–Segel chemotaxis model. *Journal of Scientific Computing*, 53(3):689–713, 2012.

[29] Y. Epshteyn and A. Kurganov. New interior penalty discontinuous Galerkin methods for the Keller–Segel chemotaxis model. *Society for Industrial and Applied Mathematics: Journal on Numerical Analysis*, 47(1):386–408, 2008.

[30] A. Fasano, A. Mancini, and M. Primicerio. Equilibrium of two populations subject to chemotaxis. *Mathematical Models and Methods in Applied Sciences*, 14(04):503–533, 2004.

[31] E. Feireisl, P. Laurençot, and H. Petzeltová. On convergence to equilibria for the Keller–Segel chemotaxis model. *Journal of Differential Equations*, 236(2):551–569, 2007.

[32] K. Forsberg-Nilsson, T.N. Behar, M. Afrakhte, J.L. Barker, and R.D.G. McKay. Platelet–derived growth factor induces chemotaxis of neuroepithelial stem cells. *Journal of neuroscience research*, 53(5):521–530, 1998.

[33] H. Gajewski, K. Zacharias, and K. Gröger. Global behaviour of a reaction–diffusion system modelling chemotaxis. *Mathematische Nachrichten*, 195(1):77–114, 1998.

[34] A. Gerisch, D.F. Griffiths, R. Weiner, and M.A.J. Chaplain. A positive splitting method for mixed hyperbolic–parabolic systems. *Numerical Methods for Partial Differential Equations*, 17(2):152–168, 2001.

[35] S.K. Godunov. A difference method for numerical calculation of discontinuous solutions of the equations of hydrodynamics. *Matematicheskii Sbornik*, 89(3):271–306, 1959.

[36] E.P. Greenberg and E. Canale-Parola. Chemotaxis in Spirochaeta aurantia. *Journal of Bacteriology*, 130(1):485–94, 1977.

[37] L. Grüne. Numerik Dynamischer Systeme. Lecture notes, 2009.

[38] C.M. Gulberg and P. Waage. Studies concerning affinity. *C.M. Forhandlinger: Videnskabs-Selskabet i Christiana*, 35, 1864.

[39] M. Gurris, D. Kuzmin, and S. Turek. Finite element simulation of compressible particle–laden gas flows. *Journal of Computational and Applied Mathematics*, 12(233):3121–3129, 2009.

[40] W. Hackbusch. *Multi-grid methods and applications*, volume 4. Springer-Verlag Berlin, 1985.

[41] K.P. Hadeler. Reaction telegraph equations and random walk systems. *In: S. van Strien, S. Verduyn Lunel (eds), Stochastic and spatial structures of dynamical systems*, pages 133–161, 1996.

[42] M.A. Herrero and J.J.L. Velázquez. Singularity patterns in a chemotaxis model. *Mathematische Annalen*, 306(1):583–623, 1996.

[43] T. Hillen. *Transport equations and chemosensitive movement*. Habilitation thesis, University of Tübingen, Tübingen, 2001.

[44] T. Hillen and K.J. Painter. A user's guide to PDE models for chemotaxis. *Journal of Mathematical Biology*, 58(1-2):183–217, 2009.

[45] D. Horstmann. The nonsymmetric case of the Keller–Segel model in chemotaxis: Some recent results. *Nonlinear Differential Equations and Applications NoDEA*, 8(4):399–423, 2001.

[46] D. Horstmann. From 1970 until present: The Keller–Segel model in chemotaxis and its consequences. *Jahresbericht der DMV*, 105(3):103–165, 2003.

[47] D. Horstmann. Generalizing the Keller–Segel model: Lyapunov functionals, steady state analysis, and blow–up results for multi-species chemotaxis models in the presence of attraction and repulsion between competitive interacting species. *Journal of Nonlinear Science*, 21(2):231–270, 2011.

[48] D. Horstmann and M. Lucia. Uniqueness and symmetry of equilibria in a chemotaxis model. *Journal für die reine und angewandte Mathematik (Crelles Journal)*, 2011(654): 83–124, 2011.

[49] D. Horstmann and G. Wang. Blow–up in a chemotaxis model without symmetry assumptions. *European Journal of Applied Mathematics*, 12(2):159–177, 2001.

[50] W. Jäger and S. Luckhaus. On explosions of solutions to a system of partial differential equations modelling chemotaxis. *Transactions of the American Mathematical Society*, 329 (2):819–824, 1992.

[51] A. Jameson. Computational algorithms for aerodynamic analysis and design. *Applied Numerical Mathematics*, 13(5):383–422, 1993.

[52] E.F. Keller and L.A. Segel. Initiation of slime mold aggregation viewed as an instability. *Journal of Theoretical Biology*, 26(3):399–415, 1970.

[53] C.T. Kelley. *Iterative methods for linear and nonlinear equations*. Number 16 in Frontiers in Applied Mathematics. Society for Industrial and Applied Mathematics, 1995.

[54] D.I. Ketcheson, S. Gottlieb, and C.B. Macdonald. Strong stability preserving two–step Runge–Kutta methods. *Society for Industrial and Applied Mathematics: Journal of Numerical Analysis*, 49(6):2618–2639, 2012.

[55] B.S. Kirk and G.F. Carey. A parallel, adaptive finite element scheme for modeling chemotactic biological systems. *Communications in Numerical Methods in Engineering*, 25(12): 1162–1185, 2009.

[56] T.M. Konijn. Effect of bacteria on chemotaxis in the cellular slime molds. *Journal of Bacteriology*, 99(2):503–509, 1969.

[57] A.B. Koudrjavcev, R.F. Jameson, and W. Linert. *The law of mass action*. Engineering Online Library. Springer Verlag, 2001.

[58] H. Kuiper. A priori bounds and global existence for a strongly coupled quasilinear parabolic system modelling chemotaxis. *Electronic Journal of Differential Equations*, 52:1–18, 2001.

[59] D. Kuzmin. On the design of algebraic flux correction schemes for quadratic finite elements. *Journal of Computational and Applied Mathematics*, 218(1):79–87, 2008.

[60] D. Kuzmin. Explicit and implicit FEM–FCT algorithms with flux linearization. *Journal of Computational Physics*, 228(7):2517–2534, 2009.

[61] D. Kuzmin. *A Guide to Numerical Methods for Transport Equations*. University Erlangen-Nürnberg, 2010. free online book.

[62] D. Kuzmin and M. Möller. Algebraic flux correction I. Scalar conservation laws. Technical report, Fakultät für Mathematik, TU Dortmund, 2004. Ergebnisberichte des Instituts für Angewandte Mathematik, Nummer 249.

[63] D. Kuzmin and S. Turek. Flux correction tools for finite elements. *Journal of Computational Physics*, 175(2):525–558, 2002.

[64] C. Landsberg, F. Stenger, A. Deutsch, M. Gelinsky, A. Rösen-Wolff, and A. Voigt. Chemotaxis of mesenchymal stem cells within 3d biomimetic scaffolds – a modeling approach. *Journal of Biomechanics*, 44:359–364, 2011.

[65] I.R. Lapidus and R. Schiller. Model for the chemotactic response of a bacterial population. *Biophysical Journal*, 16(7):779–89, 1976.

[66] D. Le. Coexistence with chemotaxis. *Society for Industrial and Applied Mathematics: Journal on Mathematical Analysis*, 32(3):504–521, 2000.

[67] D. Le and H.L. Smith. Steady states of models of microbial growth and competition with chemotaxis. *Journal of Mathematical Analysis and Applications*, 229:295–318, 1999.

[68] A.D. Luster. Chemotaxis: Role in immune response. *Wiley Online Library, Encyclopedia of Life Sciences*, 2001.

[69] R.J. MacKinnon and G.F. Carey. Positivity–preserving, flux–limited finite-difference and finite–element methods for reactive transport. *International Journal for Numerical Methods in Fluids*, 41(2):151–183, 2003.

[70] A. Marrocco. Numerical simulation of chemotactic bacteria aggregation via mixed finite elements. *ESAIM: Mathematical Modelling and Numerical Analysis – Modélisation Mathématique et Analyse Numérique*, 37(4):617–630, 2003.

[71] A. Meister. *Numerik linearer Gleichungssysteme: Eine Einfuehrung in moderne Verfahren*. Vieweg Verlag, Braunschweig, Wiesbaden, 1999.

[72] L.J. Metheny-Barlow, S. Tian, A.J. Hayes, and L.Y. Li. Direct chemotactic action of angiopoietin-1 on mesenchymal cells in the presence of VEGF. *Microvascular Research*, 68(3):221–230, 2004.

[73] M. Mimura, T. Tsujikawa, R. Kobayashi, and D. Ueyama. Dynamics of aggregation patterns in a chemotaxis–diffusion–growth model equation. In *Proceedings of the Workshop on Principles of Pattern Formation and Morphogenesis in Biological Systems (Kasugai, 1992/93), Forma 8*, 1993.

[74] M. Möller. *Adaptive high–resolution finite element schemes*. Phd thesis, Technische Universität Dortmund, 2008.

[75] J. Monod. The growth of bacterial cultures. *Annual Review of Microbiology*, 3:371–394, 1949.

[76] B Moser and K Willimann. Chemokines: Role in inflammation and immune surveillance. *Annals of the Rheumatic Diseases*, 63(2):84–89, 2004.

[77] H. Moser. *The dynamics of bacterial populations maintained in the chemostat*. Carnegie Institution of Washington, 1958.

[78] J.D. Murray. *Mathematical biology I: An introduction*. Springer, 2002.

[79] J.D. Murray. *Mathematical biology II: Spatial models and biomedical applications*. Springer, 2003.

[80] M.R. Myerscough, P.K. Maini, and K.J. Painter. Pattern formation in a generalized chemotactic model. *Bulletin of Mathematical Biology*, 60(1):1–26, 1998.

[81] T. Nagai. Blow–up of radially symmetric solutions to a chemotaxis system. *Advances in Mathematical Sciences and Applications*, 5(2):581–601, 1995.

[82] T. Nagai, T. Senba, and K. Yoshida. Application of the Moser–Trudinger inequality to a parabolic system of chemotaxis. *Funkcialaj Ekvacioj, Serio Internacia*, 40:411–433, 1997.

[83] V. Nanjundiah. Chemotaxis, signal relaying and aggregation morphology. *Journal of Theoretical Biology*, 42(1):63–105, 1973.

[84] K. Osaki and A. Yagi. Finite dimensional attractor for one-dimensional Keller–Segel equations. *Funkcialaj Ekvacioj*, 44(3):441–470, 2001.

[85] K. Osaki, T. Tsujikawa, A. Yagi, and M. Mimura. Exponential attractor for a chemotaxis–growth system of equations. *Nonlinear Analysis: Theory, Methods and Applications*, 51 (1):119–144, 2002.

[86] H.G. Othmer and A. Stevens. Aggregation, blowup, and collapse: The abc's of taxis in reinforced random walks. *Society for Industrial and Applied Mathematics: Journal on Applied Mathematics*, 57(4):1044–1081, 1997.

[87] K.J. Painter and T. Hillen. Spatio-temporal chaos in a chemotaxis model. *Physica D: Nonlinear Phenomena*, 240(4):363–375, 2011.

[88] K.J. Painter and T. Hillen. Volume–filling and quorum–sensing in models for chemosensitive movement. *Canadian Applied Mathematics Quarterly*, 10(4):501–543, 2002.

[89] S.V. Patankar. *Numerical heat transfer and fluid flow*. Hemisphere Publishing Corporation, 1980.

[90] N. Saito. Conservative upwind finite–element method for a simplified Keller–Segel system modelling chemotaxis. *IMA Journal of Numerical Analysis*, 27(2):332–365, 2007.

[91] R. Schaaf. Stationary solutions of chemotaxis systems. *Transactions of the American Mathematical Society*, 292(2):531–556, 1985.

[92] L.D. Schmidt. *The engineering of chemical reactions*. Oxford University Press New York, NY, USA, 2005.

[93] A. Sokolov, R. Strehl, and S. Turek. Numerical simulation of chemotaxis models on stationary surfaces. Technical report, Fakultät für Mathematik, TU Dortmund, 2012. Ergebnisberichte des Instituts für Angewandte Mathematik, Nummer 463.

[94] J.I. Steinfeld, J.S. Francisco, and W.L. Hase. *Chemical kinetics and dynamics*. Prentice Hall, 1989.

[95] I. Strauss, P.D. Frymier, C.M. Hahn, and R.M. Ford. Analysis of bacterial migration II: Studies with multiple attractant gradients. *American Institute of Chemical Engineers Journal*, 41(2):402–414, 1995.

[96] R. Strehl, A. Sokolov, D. Kuzmin, and S. Turek. A flux–corrected finite element method for chemotaxis problems. *Computational Methods in Applied Mathematics*, 10(2):219–232, 2010.

[97] R. Strehl, A. Sokolov, and S. Turek. Efficient, accurate and flexible finite element solvers for chemotaxis problems. *Computers and Mathematics with Applications*, 34(3):175–189, 2011.

[98] R. Strehl, A. Sokolov, D. Kuzmin, D. Horstmann, and S. Turek. A positivity–preserving finite element method for chemotaxis problems in 3D. *Journal of Computational and Applied Mathematics*, 239:290–303, 2013.

[99] M. Tindall, E.A. Gaffney, P.K. Maini, and J.P. Armitage. Theoretical insights into bacterial chemotaxis. *Wiley Interdisciplinary Reviews: Systems Biology and Medicine*, 4(3):247–259, 2012.

[100] R. Tyson. *Pattern Formation by E. coli – mathematical and numerical investigation of a biological phenomenon*. Phd thesis, University of Washington, 1996.

[101] R. Tyson, S.R. Lubkin, and J.D. Murray. A minimal mechanism for bacterial pattern formation. *Proceedings of the Royal Society of London. Series B: Biological Sciences*, 266 (1416):299–304, 1999.

[102] R. Tyson, L.G. Stern, and R.J. LeVeque. Fractional step methods applied to a chemotaxis model. *Journal of Mathematical Biology*, 41(5):455–475, 2000.

[103] R.S. Varga. *Matrix iterative analysis*, volume 27. Springer, 2009.

[104] R. Weiss. Error-minimizing Krylov subspace methods. *Society for Industrial and Applied Mathematics: Journal on Scientific Computing*, 15(3):511–527, 1994.

[105] E. Wendland and H.E. Schulz. Numerical experiments on mass lumping for the advection–diffusion equation. *Revista Minerva*, 2(2):227–233, 2005.

[106] M. Winkler. Aggregation vs. global diffusive behavior in the higher–dimensional Keller–Segel model. *Journal of Differential Equations*, 248(12):2889–2905, 2010.

[107] Michael Winkler. Boundedness in the higher–dimensional parabolic–parabolic chemotaxis system with logistic source. *Communications in Partial Differential Equations*, 35(8):1516–1537, 2010.

[108] S.M. Wise, J.S. Lowengrub, and V. Cristini. An adaptive multigrid algorithm for simulating solid tumor growth using mixture models. *Mathematical and Computer Modelling*, 53(1): 1–20, 2011.

[109] G. Wolansky. Multi-components chemotactic system in the absence of conflicts (English summary). *European Journal of Applied Mathematics*, 13(6):641–661, 2002.

[110] A. Yagi. Norm behavior of solutions to a parabolic system of chemotaxis. *Mathematica Japonicae*, 45:241–265, 1997.

[111] Bert M Zuckerman and H Jansson. Nematode chemotaxis and possible mechanisms of host/prey recognition. *Annual Review of Phytopathology*, 22(1):95–113, 1984.