

Computational Modeling of Fluorescence Photobleaching

Christian Valdemar Hansen
Department of Mathematics and Computer Science

PhD thesis
Supervised by:
Achim Schroll & Daniel Wüstner

SDU 
Odense, December 28, 2017

Resumé

Fluorescerende proteiner, såsom grønt fluorescerende protein (GFP), har ændret og revolutioneret den måde, hvorpå videnskabsfolk i dag bruger mikroskoper. Højt opløselige live celle fluorescens billede bliver i dag brugt til at studere intracellulær trafik, som ikke før var muligt at se. En af de moderne mikroskopi metoder til visualisering af transport processen i levende celler er Fluorescence Loss In Photobleaching (FLIP). I FLIP bleger en højintensitets laser et lille område i cellen, hvor der mellem blegningerne bliver taget billede.

Denne afhandling præsentere flere metoder, som kan anvendes til at beskrive den intracellulær transport, som kan observeres på FLIP billede. Et reaktion-diffusions system, som modellere både frie og hindrede molekyler, samt forskellige modeller for kernemembran transport er opstillet. I interessen for at udvikle en automatiseret analyse a FLIP billede, bliver celle geometrien fundet fra de originale FLIP billede ved brug af algoritmen Active Contours Without Edges også kendt som Chan-Vese algoritmen.

Efter vores kendskab er det første gang at nogen præsenterer en kvantitativ beregningsmæssig FLIP metode (cFLIP), som anvendes til bestemmelse af beskrivende transport parametre ud fra FLIP data. En metode, som kan bestemme diffusions konstant, membran permeabilitet, lokale bindings rater og blegnings raten for eGFP. Ydermere er metoden blevet udvidet, således at den kan modellere intracellulær protein ophobning, som er relateret til adskillige aldersrelateret sygdomme, som Alzheimers, Huntingtons, Ataksi og Parkinsons.

I tillæg præsenteres et adaptivt E-skema for viskose bevaringslove, som udspringer fra udarbejdelse af en analyse for cFLIP metoden. Numeriske forsøg og simuleringer ved brug af det adaptive E-skema er udført for at vise den forbedrede præcision. Tilmed er det bevist at E-skemaer er monotone og TVD (Total variation diminishing).

Polyglutamin proteinophobningerne, som undersøges i denne afhandling bevæger sig ikke, det er derfor muligt at anvende cFLIP metoden på faste beregningsnet. Nylige udviklinger i FEniCS gør det muligt at have flere net, som kan bevæge sig uafhængigt. MultiMesh implementeringen i FEniCS er undersøgt og anvendt til at simulere vinden og udsigten for en anlægsplan til bebyggelse. Disse giver et mål for, hvor godt husene er placeret, som kan bruges i forbindelse med at finde og designe den mest optimale anlægsplan.

Abstract

Encoded fluorescent proteins, such as the green fluorescent protein (GFP), have changed and revolutionized the way scientists use microscopes. Today high resolution live cell fluorescence imaging is used to study intracellular traffic, which was not possible before. One of the modern microscopy methods for visualization of transport processes in living cells is Fluorescence Loss In Photobleaching (FLIP). In FLIP a high-intensity laser is used to bleach a small cellular area, while images are taken between the bleaches.

This thesis presents several methods, which can be used to model the intracellular transport observed on FLIP images. A reaction-diffusion system that models both free and hindered molecules and different models for nuclear membrane transport have been proposed. In the interest of the development of an automated analysis of the FLIP images, the cell geometry was extracted from the FLIP image by use of the Active Contours Without Edges or merely Chan-Vese algorithm.

To our knowledge, it is the first time a quantitative computational FLIP (cFLIP) method used to determine transport parameters from the FLIP image data is presented. The method have been used to determine the diffusion constant, membrane permeability, local binding rates and bleaching rates for eGFP. Further, the method is expanded to model intracellular protein aggregates, which is related to various age-associated neurodegenerative diseases, such as Alzheimer's disease, Chorea Huntington, Ataxia and Parkinson disease.

Additionally, an adaptive E-scheme for viscous balance laws originated from the work on an analysis for the cFLIP method is presented in the thesis. Numerical experiments are performed to show the improved accuracy of the adaptive scheme. Moreover, it is proved that E-schemes are monotone and TVD (Total variation diminishing).

The polyglutamine protein aggregates considered in this thesis do not move, which makes it possible to apply the cFLIP method on a fixed mesh. However, recent developments in FEniCS makes it possible to have multiple meshes, which can move independently. The MultiMesh implementation is explored and used to simulate wind flow and view of a settlement layout, which thus gives a measure on how well the houses are placed, in order to find and design the optimal settlement layout.

Acknowledgements

Chiefly, I would like to thank both my advisors. Thanks to Achim Schroll, for his great guidance, experience, drive, and avidity. His mentorship was crucial and developed me as person and mathematician. I will miss our almost daily discussions.

I would also like to thank Daniel Wüstner for his always inspiring guidance. Your ideas and enthusiasm for the project have been inestimable.

Thanks to Anders Logg for hosting me at the Chalmers University of Technology where I had a rewarding stay. In that connection, I would also like to thank Carl Lundholm for a great collaboration and for introducing me to indoor climbing.

I would like to thank all the lovely people from IMADA, especially my office mates Rojin, Nicky, Christian, Jesper, Anders, Jakob, Sophie, Thomas, and Tilde, for a great time.

Finally, and most importantly, I would like to thank Vivi for her support, encouragement, patience and for the lovely son you gave us.

From the depth of my heart, I am thankful to these people, without whom, especially my advisors Achim and Daniel who have spent a lot of time working on this topic, this project would not have been possible.

Contents

I	Introduction	1
1	Preface and publications	3
2	Introduction	7
2.1	Motivation	7
2.2	Fluorescence Microscopy	8
2.2.1	Fluorescence Recovery after Photobleaching	8
2.2.2	Fluorescence Loss in Photobleaching	9
2.3	Intracellular transport	9
2.3.1	Diffusive transport	10
2.3.2	Models for transport across the membrane	10
2.4	Analysis	11
2.4.1	Inverse-monotonicity	12
2.4.2	Stability and convergence	15
II	Computational Modeling of Fluorescence Photobleaching	19
3	Computational Modeling of Fluorescence Loss in Photobleaching	21
3.1	Introduction	22
3.2	Fluorescence loss in photobleaching	24
3.3	A reaction-diffusion model in segmented FLIP images	24
3.3.1	Segmentation of FLIP images	25
3.3.2	Transport kinetics across the membrane	27
3.3.3	Reaction-diffusion dynamics	28
3.4	Implementation and Simulation	30
3.4.1	Numerical method	30
3.4.2	Tuning reaction- and diffusion parameters	31
3.4.3	Simulation of FLIP images	32

3.4.4	A reaction-diffusion model with hindrance	36
3.5	Convergence test	38
3.6	Discussion and conclusion	39
3.7	Appendix	41
4	A Discontinuous Galerkin Model for Fluorescence Loss in Photobleaching	51
4.1	Introduction	52
4.2	Fluorescence Loss in Photobleaching	54
4.3	A reaction-diffusion model in segmented FLIP images	54
4.3.1	A reaction–diffusion model with hindrance	55
4.3.2	Compartment model with semipermeable membrane . .	57
4.3.3	The complete PDE model	58
4.4	A discontinuous Galerkin method with internal interface condition	58
4.5	FEniCS implementation	60
4.6	Calibration and simulation of FLIP images	61
4.6.1	Calibration	62
4.6.2	Simulation and visualisation	63
4.7	Discussion and conclusion	65
4.8	Supplementary Information	71
5	A discontinuous Galerkin model for fluorescence loss in photobleaching of intracellular polyglutamine protein aggregates	85
5.1	Background	86
5.2	Methods	88
5.2.1	A reaction–diffusion model on real cell geometry.	88
5.2.2	Multi-compartment modeling of eGFP-mtHtt exchange	90
5.2.3	Real cell geometry	91
5.2.4	A discontinuous Galerkin method with internal interface condition	91
5.3	Results	94
5.3.1	Calibration and simulation of intracellular transport .	94
5.3.2	Calibration test	96
5.4	Discussion	98
5.5	Conclusion	100
III	An adaptive E-scheme	107
6	An Adaptive Viscosity E-scheme for Balance Laws	109
6.1	Introduction	110
6.2	A lower bound on the effective diffusion	110
6.3	An adaptive viscosity E-scheme	112
6.4	E-schemes are monotone	113

6.5	Implicit time stepping	114
6.6	E-schemes are TVD	116
6.7	Numerical experiments	117
6.7.1	A single conservation law	117
6.7.2	Conservation laws in gas dynamics	123
6.8	Boundary conditions	127
6.9	Balance laws	128
6.10	Conclusions	131
7	An adaptive E-scheme for conservation laws	135
7.1	Introduction	136
7.2	Diffusion leads to E-schemes	136
7.3	Less diffusion gives better resolution	137
7.4	E-schemes are monotone	139
7.5	Application to compressible Navier-Stokes equations	140
7.6	Concluding remarks	142
IV	Computational design of settlement layouts	147
8	Simulation of flow and view with applications in computational design of settlement layouts	149
8.1	Introduction	150
8.2	Multimesh finite element methods	151
8.3	Algorithms	153
8.3.1	Computation of flow	153
8.3.2	Computation of view	157
8.4	Implementation	158
8.4.1	Implementation of flow computation	158
8.4.2	Implementation of view computation	161
8.5	Results	163
8.5.1	Flow	163
8.5.2	View	165
8.6	Conclusions and future work	166

Part I

Introduction

Preface and publications

In this first chapter, a short overview of the content of each section in this thesis is given. The thesis is a result of my work as a Ph.D. student at the Department of Mathematics and Computer Science (IMADA) at the University of Southern Denmark from December 2014 to December 2017. The thesis is split into four main parts of which three of them contains papers. Each of the papers produced during my Ph.D. is included as a self-contained chapter in the three parts, meaning that each chapter contains its own bibliography and that the notation can vary between the chapters. Thus the same references and definitions may appear in several of the chapters.

In part I the preface and introduction can be found. The introduction contains a short motivation, short introduction to fluorescence microscopy and a stability analysis. Part II contains results on modeling of fluorescence photobleaching. In Part III an adaptive E-scheme for viscous balance laws is presented. Part IV contains a study on flow and view for architectural settlement layouts.

The content of this thesis originates from the work on "Computational Modeling of Fluorescence Photobleaching," which is both the title of the thesis and Part II. Part II contains three papers that all focus on modeling of Fluorescence loss in photobleaching (FLIP), which is a fluorescence microscopy technique for visualization of the transport processes in living cells. In the appendix to the first article in Chapter 3, one can find an analysis on the reaction-diffusion system used in the paper; the analysis never made it into any of the papers in Part II. However, the work on the analysis inspired us to expand and generalize the analysis to viscous balance laws. The two papers presenting a new adaptive E-scheme for degenerate, viscous balance laws can thus be found in Part III. The last Part IV, contains one paper, which is a result of a collaboration that started when I was visiting Anders Logg at Chalmers University for environmental change. It is a project on architectural settlement layouts based on flow and view using multiple meshes. The work with multi-mesh finite element methods is related to the work on FLIP mod-

eling in the way that it might be used as an expansion of the paper in Chapter 5 with moving aggregates/compartments/structures in the living cells.

Chapter 3 contains the paper "Computational modeling of fluorescence loss in photobleaching," which was published in "Computing and Visualization in Science" by Springer. The paper was written in collaboration with my two advisors Achim Schroll and Daniel Wüstner and presents a framework for modeling and simulation of FLIP sequences as a reaction–diffusion system on segmented cell images, which was to our knowledge the first FLIP simulation on a real cell geometry. The solution to the system is continuous functions thus, as the intensity in the cytoplasm and nucleus are different, a thin but finite membrane is needed. The need of a thin membrane with a fine mesh has a high computational cost, which was the motivation for the next paper.

In Chapter 4 you find the paper "A Discontinuous Galerkin Model for Fluorescence Loss in Photobleaching," which is accepted for publication in Scientific Reports by Nature. It is written in cooperation with my two advisors Achim Schroll and Daniel Wüstner. In this paper, we introduce the use of a permeability model, that allows discontinuities between the cytoplasm and nucleus. Consequently, the spatial membrane can be left out and the computational complexity is drastically reduced. The complexity reduction facilitates calibration of the parameters in the reaction–diffusion system. Thus, this work contains every element needed for an automated analysis of FLIP images.

Chapter 5 contains the paper "A discontinuous Galerkin model for fluorescence loss in photobleaching of intracellular polyglutamine protein aggregates" which is submitted to BMC Bioinformatics. The paper is made in collaboration with Achim Schroll and Daniel Wüstner and uses the computational FLIP model presented in Chapter 4 combined with a compartment model for aggregates. Calibration of parameters for two different cells have been made, to investigate the precision of the method.

In Part III the first chapter; Chapter 6, contains the paper "An Adaptive E–scheme for Viscous Balance Laws." It is submitted to SIAM Journal on Numerical Analysis (SINUM). It is written in cooperation with Ebise A. Abdi and Achim Schroll and is based on ideas from the analysis presented in Chapter 2. In this paper, it is proven that E–schemes, in general, are monotone and total variation diminishing (TVD). Thus it follows that the presented adaptive E–scheme is monotone and TVD too, and in addition it is stable.

In Chapter 7 the paper "An adaptive E–scheme for conservation laws" can be found. It was presented at ENUMATH 2017 in Voss, Norway, and submitted to the ENUMATH 2017 Proceedings Volume to be published by Springer.

This proceeding is based on the adaptive E-scheme presented in Chapter 6, and numerical experiments are presented to demonstrate the robustness and improved accuracy of the adaptive scheme.

In the last part, you find Chapter 8, which contains the paper "Simulation of flow and view with application to the computational design of settlement layouts." This paper can be found on <https://arxiv.org/abs/1610.02277>. It describes the simulation of flow and view, which can be used as a measure on how well the houses are placed, and thus be used to find the optimal settlement layout. For the simulation, the MultiMesh implementation in FEniCS is used. At the time of writing the bugs in the 3D MultiMesh implementation in FEniCS are being fixed by the developers. This is also why the paper has not been submitted to a peer-reviewed journal yet.

Introduction

2.1 Motivation

Living cells are in constant movement, and one way to capture these dynamics is by fluorescence microscopy combined with quantitative image analysis. Using this approach, transport processes have been described and analyzed in detail, both in healthy and diseased cells. In fact, diseases like Alzheimer, Parkinson, Corea Huntington and Arteriosclerosis are accompanied or even caused by a jam in the intracellular traffic machinery [1]. Hence to improve diagnosis and treatment, a quantitative and automated analysis of intracellular transport is essential. Fluorescence photobleaching techniques combined with time-lapse imaging are efficient methods to obtain spatiotemporal information about intracellular transport processes. The fundamental idea behind these techniques is that one makes use of genetically encoded fluorescent protein constructions, whose steady-state distribution in the cell is locally perturbed by an intense laser pulse. The reestablishment of a new steady state is observed in techniques such as fluorescence recovery after photobleaching (FRAP), which is the most widely applied fluorescence photobleaching technique. However, even though techniques such as FRAP are widely used, there is a lack of computational tools to analyze the obtained data. One important fluorescence microscopy technique for visualization and potentially quantification of the transport processes in living cells is the so-called fluorescence loss in photobleaching (FLIP). Since in FLIP one has access to the whole spatiotemporal evolution of fluorescence changes in the cellular geometry, the motivation for making a reliable automated analysis of the image data is even higher.

To develop an automated computer analysis for FLIP images, the first step would be to identify the geometry of the cell.

In my Bachelor project "*Segmentation of fluorescence microscopy images of living cells*" I used the "Chan-Vese" method, also known as "Active Contours Without Edges," to segment images. The project successfully implemented the Chan-Vese method, such that it can be used to detect cell edges on the FLIP images.

Based on this segmentation, experimental FLIP sequences, mathematical modeling, finite element simulations, analysis and calibration we have made our effort in creating an automated analysis of FLIP images. The results of this research are presented in the first three papers presented in this thesis.

Below a short introduction to Fluorescence Microscopy, FRAP and FLIP are given. However, for a more detailed description of FLIP, the reader is referred to the FLIP introductions in Chapter 3 – 5.

2.2 Fluorescence Microscopy

Cellular transport processes are often investigated using fluorescence-tagged molecules and live-cell microscopy. Fluorescence is characterized by its ability to emit light that the substance has already absorbed. A lot of different microscopy techniques involving visualization and analysis of the dynamic in cells have been developed upon the use of fluorescence. Generally, in fluorescence microscopy, the emitted fluorescence light is detected by the same objective as used to illuminate the sample with excitation light.

To study transport dynamics in cells photobleaching is widely used. Photobleaching is a process, which permanently unable the fluorophore molecule to enter the repetitive excitation cycle, e.g., by cleaving of the covalent bonds to the surrounding molecules.

Photobleaching is here performed by a high-intensity laser on a confocal microscope. The confocal laser scanning microscopy technique makes it possible to combine high-resolution optical imaging and depth selectivity [6]. A confocal microscope uses point illuminations and thus build up the images pixel by pixel. The pinhole in front of the detector makes it possible to achieve well-focused image as it eliminates the out-of-focus emitted light.

In the recent years, many different fluorescent proteins have been developed through genetic modification. However, the most common and the one used here is the green fluorescent protein (GFP), which refers to the protein originally isolated from the jellyfish, *Aequorea victoria*.

As mentioned a lot of different fluorescence microscopy methods exists, below a short description of the two related methods, FRAP and FLIP, can be found. These are both based on selective photodestruction of fluorescence from the molecule of interest in one organelle.

2.2.1 Fluorescence Recovery after Photobleaching

As already mentioned FRAP may be designated as the most commonly applied fluorescence microscopy techniques. In FRAP a cellular region of interest is photobleached with a high-intensity laser. After bleaching, the fluorescence

recovery rate inside the bleached area is measured.

The recovery rate provides information about the transport dynamics involved when the neighboring molecules diffuse into the previously bleached area. Thus the generated recovery curve can be used to determine the diffusion as well as the mobile and immobile fractions. However, many different techniques and kinetic models have been developed through the years. Thus various studies use various models and arrive at different results for the same proteins. The difference in the results seems to arise from technical issues rather than biological differences since the recovery curve in FRAP do not hold any spatial information and thus only depends on the time the risk of overfitting is high [5].

2.2.2 Fluorescence Loss in Photobleaching

Contrary to FRAP, the area of interest in FLIP experiments is not only the bleached area but the whole cell. FLIP is a technique, where a small region is repeatedly illuminated by an intense laser pulse. A pause between the laser pulses allows for some recovery in the bleached region. A decrease in fluorescence of the labeled molecules outside the bleached area allows for assessing continuity between intracellular compartments, and in principle, for measuring the kinetics of recruitment of fluorescent proteins to the bleached region from various cellular regions. Accordingly, FLIP has the potential to include spatial information in the computational analysis an advantage that other fluorescence microscopy techniques lack.

One of the advantages of FLIP is that it can be used to investigate possible exchanges between two compartments, e.g., it may be of interest to know if there is an exchange between aggregates and the cytoplasm. FLIP can be used to investigate this by performing photobleaching in one of the compartments and then inspect whether it has any effect on the other compartment. If the fluorescence intensity in the non-bleached compartment is affected by the bleaching, there would most likely be an exchange.

2.3 Intracellular transport

Living cells use a variety of transport modes to secure the correct protein and lipid composition of their organelles. To describe this complexity in quantitative terms using mathematical models requires simplifying the observed phenomena to the essential elements, e.g., the cytoplasm has long been considered as a homogenous fluid. Evidence obtained using quantitative fluorescence and electron microscopy, however, has revealed a hierarchically organized entangled network architecture of the cytoplasm. As a consequence, realistic transport models have to account for this structural organization including

space-dependent diffusion constants and complex cell shapes.

2.3.1 Diffusive transport

The main transport process modeled in this thesis is the diffusive transport process. Diffusion is a result of the fact that solvent molecules collide stochastically with the solute molecules which causes them to move in random directions. These random movements on the molecular scale can nevertheless outbalance eventual concentration gradients of the solute molecules, e.g., as generated by bleaching with fluorescent photobleaching techniques. The macroscopic description of diffusion as a mechanism to counteract concentration gradients of solute molecules was for the first time expressed as an equation in 1855 by Adolf Fick. His second law, which balances the temporal evolution of concentration changes with the spatial change of diffusive fluxes is known as Fick's law of diffusion or simply the Diffusion Equation.

Looking at a glass of water with a drop of ink in it; the ink molecules would slowly start to diffuse out into the surrounding water because of the Brownian motion created by the water molecules. Looking at this example, the hypothesis that the ink molecules go from a region of higher concentration to a region of lower concentration could arise. Based on a similar hypothesis Fick's first law arose:

$$\mathbf{J}(t, \mathbf{x}, u) = -d\nabla u(t, \mathbf{x}),$$

where \mathbf{J} is the diffusive flux of the substance, with the proportion or amount u dependent on space \mathbf{x} and time t . The interesting term here is the diffusion coefficient d , which can be space dependent, but as the case in this thesis, d is often kept constant for simplicity. From Fick's first law and the principle of mass conservation, Fick's second law can be derived as:

$$u_t = d\Delta u(t, \mathbf{x}).$$

Here Δ denotes the Laplacian operator and the partial derivative is denoted with the usual subscript such that $\partial_t = \frac{\partial}{\partial t}$.

2.3.2 Models for transport across the membrane

An other important transport process which has been considered is the transport across the membrane.

In the paper "Computational Modeling of Fluorescence Loss in Photobleaching" in Chapter 3 we present a model for membrane transport which combines diffusion and first-order reaction kinetic with space-dependent logistic reaction rates. A model, which requires a rather dense and fine mesh in the membrane region, but it allows for transport against the concentration gradient. Even though it allows for transport against the gradient, it is driven by the diffusive transport processes across the membrane. In "A Discontinuous Galerkin

Model for Fluorescence Loss in Photobleaching" which can be found in Chapter 4 a semi-permeable model is presented. This model is built on top of Fick's first law which by construction allows a diffusive flux across the nuclear membrane. A model that is suitable for molecular diffusion through nuclear pore complexes, i.e., particles that are typically smaller than $\sim 30 - 60$ kDa [7].

The model in "Computational Modeling of Fluorescence Loss in Photobleaching," Chapter 3 allows for transport against the gradient. However, another approach, which has not been implemented in the papers, is to use the same approach as presented for the aggregate traffic in "A discontinuous Galerkin model for fluorescence loss in photobleaching of intracellular polyglutamine protein aggregates," Chapter 5. This method can be seen as an interior boundary condition, and thus it does not require a finite membrane with a fine mesh, implying the computation time to be lowered.

2.4 Analysis

The typical system considered throughout this thesis is a reaction-diffusion system on the form

$$\begin{aligned} u_t &= u_{xx} + k^-v - k^+u , \\ v_t &= v_{xx} + k^+u - k^-v , \quad x \in (0, 1) , \quad t > 0 . \end{aligned} \quad (2.1)$$

In this chapter, an analysis of the system in one spatial variable and with diffusion coefficients scaled to unity is presented. A similar analysis can be found in the appendix to the paper "Computational Modeling of Fluorescence Loss in Photobleaching" found in Chapter 3. This section is a more explanatory expansion of that, thus some of the sentences and clauses here are similar to the one used in the analysis from appendix B in Chapter 3. For this analysis we consider homogenous Dirichlet and Neumann conditions for u and v , respectively

$$\begin{aligned} u(t, 0) &= u(t, 1) = 0 , \\ v_x(t, 0) &= v_x(t, 1) = 0 , \end{aligned}$$

and initial data $u(0, x) = u^0(x)$ and $v(0, x) = v^0(x)$, for $x \in (0, 1)$. The analysis is based on elements from [2], which originally go back to R.S. Varga [8], L. Collatz [4] and E. Bohl [3].

The bilinear form

$$\int_0^1 u_{xx}\phi \, dx = - \int_0^1 u_x \cdot \phi_x \, dx$$

acting on a finite dimensional approximation $u(x) = \sum_j u_j \phi_j(x)$ with piecewise linear elements $\phi_j(x_i) = \delta_{ij}$ is expressed as matrix-vector product

$$- \int_0^1 u_x \cdot (\phi_j)_x \, dx = (\mathbf{A}_h \mathbf{u})_j .$$

The vector \mathbf{u} holds approximations on inner grid points $\mathbf{u} = (u_i)$, $u_i = u(x_i)$, $x_i = ih$, $i = 1, 2, \dots, m$, and $(m+1)h = 1$. The matrix is given by

$$\mathbf{A}_h = (a_{ij}) \quad , \quad a_{ij} = - \int_0^1 (\phi_i)_x \cdot (\phi_j)_x \, dx \quad .$$

Due to the piecewise constant gradients $(\phi_j)_x$, the matrix becomes a well-known (finite difference) approximation to the second derivative. With homogenous Dirichlet conditions the linear operator reads

$$-u_{xx} \approx \mathbf{A}_h \mathbf{u} = \frac{1}{h^2} \begin{pmatrix} 2 & -1 & & \\ -1 & \ddots & \ddots & \\ & \ddots & \ddots & -1 \\ & & -1 & 2 \end{pmatrix} \mathbf{u} \quad ,$$

where the vector u holds approximations on inner grid points $u_i \approx u(x_i)$, $x_i = ih$, $i = 1, 2, \dots, m$, and $(m+1)h = 1$. In the case of homogenous Neumann conditions $v_0 = v_1$ and $v_m = v_{m+1}$, the second derivative on inner grid points is approximated by

$$-v_{xx} \approx \mathbf{B}_h \mathbf{v} = \frac{1}{h^2} \begin{pmatrix} 1 & -1 & & & \\ -1 & 2 & -1 & & \\ & \ddots & \ddots & \ddots & \\ & & -1 & 2 & -1 \\ & & & -1 & 1 \end{pmatrix} \mathbf{v} \quad .$$

Semi-implicit time stepping results in a linearly coupled, discrete system

$$\begin{aligned} [(1 + \Delta t k^+) \mathbf{I} + \Delta t \mathbf{A}_h] \mathbf{u}^{n+1} &= \mathbf{u}^n + \Delta t k^- \mathbf{v}^n \quad , \\ [(1 + \Delta t k^-) \mathbf{I} + \Delta t \mathbf{B}_h] \mathbf{v}^{n+1} &= \mathbf{v}^n + \Delta t k^+ \mathbf{u}^n \quad . \end{aligned} \quad (2.2)$$

Naturally, step sizes are positive Δt , $h > 0$ and reaction rates are non-negative $k^\pm \geq 0$. Again, u_i^n is thought as approximation to $u(t_n, x_i)$. In fact, one can prove that the numerical approximation converges to the solution of the PDE system.

2.4.1 Inverse-monotonicity

The key argument relies on the fact that the finite difference operators $\mathbf{I} + \Delta t \mathbf{A}_h$ as well as $\mathbf{I} + \Delta t \mathbf{B}_h$ are inverse-monotone with bounded inverse. In fact, \mathbf{A}_h inherits its inverse-monotonicity from the negative Laplacian, while \mathbf{B}_h is obviously singular.

Lemma 2.4.1. *With non-negative parameters $\Delta t \geq 0$, $h > 0$ and reaction rates $k^\pm \geq 0$ both system matrices $(1 + \Delta t k^+) \mathbf{I} + \Delta t \mathbf{A}_h$ and $(1 + \Delta t k^-) \mathbf{I} + \Delta t \mathbf{B}_h$ are regular with non-negative and uniformly bounded inverse.*

An obvious consequence is:

Corollary 2.4.2. *With non-negative initial data $u^0 \geq 0$ and $v^0 \geq 0$ and reaction rates $k^\pm \geq 0$, approximations computed by (2.2) remain non-negative for all time $n\Delta t \geq 0$.*

The proof of Lemma 2.4.1 is based on inverse-monotone Z -matrices. A Z -matrix is characterized by its non-positive off-diagonal elements:

$$\mathbf{Z} = (z_{ij}) , \quad z_{ij} \leq 0 , \quad i \neq j .$$

A matrix \mathbf{A} is called inverse-monotone if it is invertible with elementwise non-negative inverse $\mathbf{A}^{-1} \geq 0$. In this case $\mathbf{A}\mathbf{e} \geq 0$ implies $\mathbf{e} \geq 0$; i.e., solving the system preserves positivity. An inverse-monotone Z -matrix is called M -matrix.

Any strictly positive vector $\mathbf{p} > 0$ defines a weighted max-norm

$$\|\mathbf{x}\|_p = \max_i \{|x_i|/p_i\} .$$

The corresponding operator norm for a monotone (i.e. elementwise positive) matrix is $\|\mathbf{A}\|_p = \|\mathbf{A}\mathbf{p}\|_p$:

Lemma 2.4.3. *For $0 < \mathbf{p} \in \mathbb{R}^n$ and $0 \leq \mathbf{A} \in \mathbb{R}^{n \times n}$*

$$\|\mathbf{A}\|_p = \|\mathbf{A}\mathbf{p}\|_p . \quad (2.3)$$

Proof of Lemma 2.4.3: Let $\mathbf{x} \in \mathbb{R}^n$ be arbitrary. As $\mathbf{A} \geq 0$,

$$\frac{|(\mathbf{Ax})_i|}{p_i} \leq \sum_j \left(\frac{a_{ij}p_j}{p_i} \frac{|x_j|}{p_j} \right) \leq \sum_j \frac{a_{ij}p_j}{p_i} \|\mathbf{x}\|_p .$$

Hence,

$$\|\mathbf{Ax}\|_p = \max_i \frac{|(\mathbf{Ax})_i|}{p_i} \leq \max_i \frac{(\mathbf{Ap})_i}{p_i} \|\mathbf{x}\|_p .$$

Again by positivity $\mathbf{Ap} \geq 0$, $\|\mathbf{Ax}\|_p \leq \|\mathbf{Ap}\|_p \|\mathbf{x}\|_p$ and $\|\mathbf{A}\|_p \leq \|\mathbf{Ap}\|_p$. The reversed inequality is obvious, and the proof is complete. \square

Lemma 2.4.4. *A Z -matrix \mathbf{A} is M -matrix if and only if \mathbf{A} is semi-positive. That is, there exists $\mathbf{e} > 0$ with $\mathbf{A}\mathbf{e} > 0$.*

Proof of Lemma 2.4.4: The proof of Lemma 2.4.4 (in German) can be found in E. Bohl, Satz 5.1 [2].

1. Let \mathbf{A} be M -matrix. Then $\mathbf{e} = \mathbf{A}^{-1}\delta > 0$ and $\mathbf{A}\mathbf{e} = \delta > 0$. Here $\delta = (1, \dots, 1)^T$ denotes the vector consisting of just ones.

2. Let $\mathbf{e} > 0$ and $\mathbf{A}\mathbf{e} > 0$. As \mathbf{A} is Z -matrix

$$(\mathbf{A}\mathbf{e})_i = a_{ii}e_i - \sum_{j \neq i} |a_{ij}|e_j > 0$$

and hence $a_{ii} > 0$. Consequently, $\mathbf{D} = \text{diag}(a_{ii})$ is regular and

$$\mathbf{P} = \mathbf{D}^{-1}(\mathbf{D} - \mathbf{A}) \geq 0 , \quad \mathbf{A} = \mathbf{D}(\mathbf{I} - \mathbf{P}) .$$

Further, $(\mathbf{I} - \mathbf{P})\mathbf{e} = \mathbf{D}^{-1}\mathbf{A}\mathbf{e} > 0$ and $\mathbf{P}\mathbf{e} < \mathbf{e}$. By (2.3) $\|\mathbf{P}\|_e < 1$. By the Banach fixed-point theorem $\mathbf{I} - \mathbf{P}$ is regular and

$$(\mathbf{I} - \mathbf{P})^{-1} = \sum_{j=0}^{\infty} \mathbf{P}^j \geq 0 .$$

Therefore,

$$\mathbf{A}^{-1} = (\mathbf{I} - \mathbf{P})^{-1}\mathbf{D}^{-1} = \sum_{j=0}^{\infty} \mathbf{P}^j \mathbf{D}^{-1} \geq 0 .$$

The proof is complete. \square

Lemma 2.4.5. *Let \mathbf{A} be M -matrix and $\mathbf{D} = \text{diag} \geq 0$. Then $\mathbf{A} + \mathbf{D}$ is M -matrix.*

Proof of Lemma 2.4.5: As \mathbf{A} is M -matrix, there is $\mathbf{e} > 0$ such that $\mathbf{A}\mathbf{e} > 0$. As \mathbf{D} is diagonal, $\mathbf{A} + \mathbf{D}$ is Z -matrix. As $\mathbf{D} \geq 0$, $(\mathbf{A} + \mathbf{D})\mathbf{e} > \mathbf{D}\mathbf{e} \geq 0$. By Lemma 2.4.4, $\mathbf{A} + \mathbf{D}$ is M -matrix. \square

Proof of Lemma 2.4.1: Note $\mathbf{A}_h\delta = (1, 0, \dots, 0, 1)/h^2$ and $B_h\delta = 0$. Obviously \mathbf{B}_h is singular. Both $\mathbf{I} + \Delta t\mathbf{A}_h$ and $\mathbf{I} + \Delta t\mathbf{B}_h$ are Z -matrices. Both are semi-positive

$$(\mathbf{I} + \Delta t\mathbf{A}_h)\delta \geq (\mathbf{I} + \Delta t\mathbf{B}_h)\delta \geq \delta > 0 .$$

By Lemma 2.4.4 $\mathbf{I} + \Delta t\mathbf{A}_h$ and $\mathbf{I} + \Delta t\mathbf{B}_h$ are M -matrices. We may add any positive diagonal matrix $\Delta t k \mathbf{I}$ and the M -property is preserved by Lemma 2.4.5. Thus both system matrices are M -matrices, even with non-constant, non-negative reaction rates k^\pm . It remains to show the uniform bound for the inverse. We have that

$$\begin{aligned} [(1 + \Delta t k^+)\mathbf{I} + \Delta t\mathbf{A}_h]\delta &\geq \delta , \\ [(1 + \Delta t k^-)\mathbf{I} + \Delta t\mathbf{B}_h]\delta &\geq \delta . \end{aligned}$$

Multiplying by the non-negative inverse, taking norms and applying Lemma 2.4.3 the desired bound follows

$$\begin{aligned} \left\| [(1 + \Delta t k^+)\mathbf{I} + \Delta t\mathbf{A}_h]^{-1} \right\|_\delta &\leq 1 , \\ \left\| [(1 + \Delta t k^-)\mathbf{I} + \Delta t\mathbf{B}_h]^{-1} \right\|_\delta &\leq 1 \end{aligned}$$

and the proof is complete. \square

Note that $\|\cdot\|_\delta$ is the well-known max-norm. Via the eigensystem of \mathbf{A}_h , one can also show that \mathbf{A}_h itself is a M-matrix.

2.4.2 Stability and convergence

The uniform bound of the inverse system matrix implies stability of the numerical method with respect to errors in the initial data $\Delta\mathbf{u}^0$ and $\Delta\mathbf{v}^0$. By linearity these errors are governed by the system (2.2) itself

$$\begin{aligned} [(1 + \Delta t k^+) \mathbf{I} + \Delta t \mathbf{A}_h] \Delta \mathbf{u}^{n+1} &= \Delta \mathbf{u}^n + \Delta t k^- \Delta \mathbf{v}^n , \\ [(1 + \Delta t k^-) \mathbf{I} + \Delta t \mathbf{B}_h] \Delta \mathbf{v}^{n+1} &= \Delta \mathbf{v}^n + \Delta t k^+ \Delta \mathbf{u}^n . \end{aligned}$$

Uniform boundedness implies

$$\begin{aligned} \|\Delta \mathbf{u}^{n+1}\|_\delta &\leq \|\Delta \mathbf{u}^n\|_\delta + \Delta t k^- \|\Delta \mathbf{v}^n\|_\delta , \\ \|\Delta \mathbf{v}^{n+1}\|_\delta &\leq \|\Delta \mathbf{v}^n\|_\delta + \Delta t k^+ \|\Delta \mathbf{u}^n\|_\delta , \end{aligned}$$

and

$$\|\Delta \mathbf{u}^{n+1}\|_\delta + \|\Delta \mathbf{v}^{n+1}\|_\delta \leq (1 + \Delta t K) (\|\Delta \mathbf{u}^n\|_\delta + \|\Delta \mathbf{v}^n\|_\delta) ,$$

where $K = \max(k^+, k^-)$. Finally,

$$\|\Delta \mathbf{u}^n\|_\delta + \|\Delta \mathbf{v}^n\|_\delta \leq (1 + \Delta t K)^n (\|\Delta \mathbf{u}^0\|_\delta + \|\Delta \mathbf{v}^0\|_\delta) , \quad (2.4)$$

with $(1 + \Delta t K)^n \leq \exp(n \Delta t K) \leq \exp(TK)$, $n \Delta t \leq T$.

The inequality (2.4) expresses stability in the sense that errors at later time depend continuously on initial errors. Convergence towards a smooth solution of the reaction-diffusion system (2.1) follows from the discrete Gronwall lemma.

The approach for this analysis is generalized and applied in Part III. In section 6.9 a reaction-diffusion system in the form of a balance law is considered.

Bibliography

- [1] Aridor M., Hannan L.A.: Traffic Jams II: An Update of Diseases of Intracellular Transport. *Traffic*, 3(11), pp. 781-790. (2002)
- [2] Bohl E.: Finite Modelle gewöhnlicher Randwertaufgaben, *Teubner, Stuttgart.* (1981)
- [3] Bohl E.: Monotonie: Lösbarkeit und Numerik bei Operatorgleichungen, *Springer Tracts in Natural Philosophy 25, Berlin.* (1974)
- [4] Collatz L.: Funktionalanalysis und Numerische Mathematik, *Springer, Berlin.* (1964)
- [5] Mueller F., D. Mazza D., Stasevich T.J., McNally J.G.: FRAP and kinetic modeling in the analysis of nuclear protein dynamics: what do we really know? *Curr. Opin. Cell Biol.* 22, 403-411. (2010)
- [6] Pawley, J.B. *Handbook of Biological Confocal Microscopy*, 3rd ed. Springer US. (2006)
- [7] Timney B.L., Raveh B., Mironksa R., Trivedi J.M., Kim S.J., Russel D., Wente S.R., Sali A., Rout M.P.: Simple rules for passive diffusion through the nuclear pore complex, *The Journal of Cell Biology.* (2016)
- [8] Varga R.S.: Matrix iterative analysis, *Prentice-Hall, Englewood Cliffs, N.J.* (1962)

Part II

Computational Modeling of Fluorescence Photobleaching

Computational Modeling of Fluorescence Loss in Photobleaching

Christian V. Hansen, Hans J. Schroll, Daniel Wüstner

Abstract

Fluorescence loss in photobleaching (FLIP) is a modern microscopy method for visualization of transport processes in living cells. Although FLIP is widespread, an automated reliable analysis of image data is still lacking. This paper presents a framework for modeling and simulation of FLIP sequences as reaction–diffusion systems on segmented cell images. The cell geometry is extracted from microscopy images using the Chan–Vese active contours algorithm [8]. The PDE model is subsequently solved by the automated Finite Element software package FEniCS [20]. The flexibility of FEniCS allows for spatially resolved reaction diffusion coefficients in two (or more) spatial dimensions.

3.1 Introduction

Many human diseases involve or are even caused by malfunction in intracellular protein and membrane trafficking. Examples are Huntington and Parkinson disease, Dubin-Johnson syndrome and various lysosomal storage disorders [1]. Monitoring of intracellular transport in health and disease heavily relies on quantitative fluorescence microscopy. In particular, photobleaching studies of the protein of interest tagged to enhanced green fluorescent protein (eGFP) are increasingly acknowledged as tool for deciphering transport kinetics and dysfunction [7, 19]. In fluorescence recovery after photobleaching (FRAP) a selected region in the cell is rapidly bleached with a high intensity laser pulse followed by sequential imaging with reduced laser power to determine recovery kinetics due to protein transport into the bleached region. Extensive mathematical modeling has been applied to recover diffusion constants and binding parameters from fluorescence recovery curves [23].

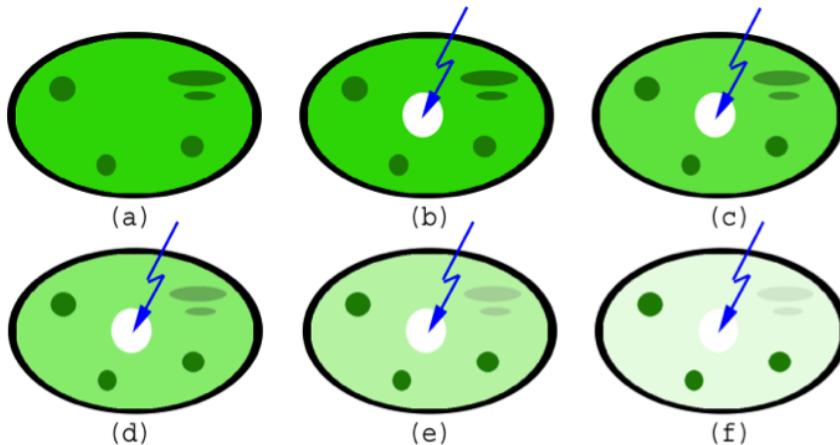


Figure 3.1: Schematic illustration of a FLIP experiment. A cell contains fluorescent molecules (green) (a). A small region is repeatedly bleached by laser light (blue flash) (b-f). Fluorescence begins to fall in the cytoplasm and in some organelles (ellipses) but not in others (circles). This demonstrates diffusive transport of molecules towards the bleached region with recruitment from the elliptical organelle but not from the circular organelles, suggesting that the fluorescent molecules are confined in the latter.

Since the recovery curve in FRAP only depends on time, any spatial information of the acquired image sequence is lost with the consequence, that overfitting often takes place, and the recovered parameters are of low reliability [23]. In fact, it has been shown that several mechanistically differing diffusion/binding models are capable of fitting the same FRAP curve with comparable accuracy [23]. This phenomenon is at least partly a consequence

of the fact that a complex spatiotemporal reaction-diffusion model is fitted to a time-dependent fluorescence recovery curve, only. Thus, the spatial aspect of the transport process enters only indirectly such that the fitting process can provide ambiguous results. In other cases, as for binding-dominated FRAP modeling (i.e. in which diffusion is much faster than binding resulting in well-mixed binding partner distribution), the recovery process is modeled as a sum of exponentials [25]. In the continuum limit of infinitely many binding partners, this would lead to a Fredholm integral equation of the 1st kind, which is known to be ill-posed [16, 22]. Unequivocal determination of all binding parameters would not be possible in this case from a single recovery curve. To overcome these limitations, line profiles recorded over the bleached area are often included in the FRAP analysis [17, 26]. This approach results in much improved parameter sensitivity and thereby higher fitting accuracy. However, as in spot-bleaching FRAP experiments, no information about the intracellular origin of the fluorescent molecules being recruited to the bleached region can be discerned. Accordingly, the different binding sites are not considered explicitly resulting in summing them as exponential terms in the fitting function. To overcome this problem, the whole spatiotemporal image information must be taken into account, thereby being able to discern individual binding sites directly from the images. Fluorescence loss in photobleaching (FLIP) is a variation of FRAP, in which the intense laser bleaching is performed repeatedly in a selected region, while fluorescence loss is monitored inside and outside the bleached area. This bears the potential to include pixel-wise information about the whole spatiotemporal dynamics of tagged molecules in the modeling process. Parameter estimation from FLIP data would, of course, suffer from the same fundamental limitations as explained above for the FRAP analysis. However, including all available spatial information given in a FLIP image set should allow for reducing the parameter space in a later regression of given reaction diffusion models compared to fitting models with multiple parameters to a single FRAP which just depends on time as variable. Since, one records the whole spatiotemporal profile of probe distribution in FLIP image sequences, local heterogeneity as distributed binding sites can be explicitly considered in the modeling process. Thus, FLIP provides two important advantages compared to FRAP: (1) in FLIP, the user can better interfere with the dynamics of the measurement process by varying the duration of pauses between individual bleach pulses in addition to altering the strength and duration of the bleaching pulse. (2) Information about the whole compartment in which the transport process takes place can enter the FLIP analysis. Given the heterogeneity of intracellular structures, this will be of advantage in future studies inferring diffusion and binding parameters from FLIP image sequences. Recently, we have presented a pixel-by-pixel fitting of an empirical decay law to diffusion- and binding-limited FLIP image sequences of eGFP and eGFP-tagged Huntingtin protein [29]. Although a step in the right direction, our

method did not allow us to model the spatiotemporal protein dynamics based on an underlying physical mechanism. This, however, will be instrumental for achieving our long-term goal, namely estimation of local binding and diffusion parameters of fluorescent molecules directly from the experimental FLIP image sequences.

In this paper, we present a framework to simulate FLIP sequences based on reaction–diffusion models on the cell geometry being extracted from two dimensional confocal image sequences. The model preserve positivity of intensities and averaged intensities appear correlated with experimental data. Introducing aggregation of molecules in prescribed areas, the simulated images reproduce basic structures as seen in experimental FLIP sequences.

The outline of the paper is as follows: Fluorescence photobleaching technology is explained in Section 3.2. In Section 3.3 our reaction–diffusion model is developed. A finite element simulation of a FLIP experiment is presented in Section 3.4. Second order convergence of numerical approximations is verified in a numerical experiment in Section 3.5. We summarize our findings in Section 3.6. Shape gradients are derived in Appendix A. Well-posedness and positivity for a simplified model in one-dimension are proven in Appendix B.

3.2 Fluorescence loss in photobleaching

Cellular transport processes are often investigated using fluorescence–tagged molecules and live–cell microscopy [19]. In FLIP experiments, a region of the cell is repeatedly illuminated by an intense laser pulse destroying the fluorescence of the molecule of interest in that area. A pause between the laser pulses allows for some recovery in the bleached region, see Fig. 3.1 for illustration. Repeating this protocol several times creates a sink for the fluorescent molecules in the local environment being in continuous exchange with the bleached region. A decrease of fluorescence of the labeled molecules outside the bleached area allows for assessing continuity between intracellular compartments, and in principal, for measuring the kinetics of recruitment to the bleached region from various cellular areas. Accordingly, FLIP has the potential to include spatial information in the computational analysis [29].

3.3 A reaction–diffusion model in segmented FLIP images

Living cells are in continuous exchange with their environment and use a variety of transport modes to secure the correct protein and lipid composition of their organelles. Simple Brownian motion and anomalous diffusion of proteins in the cytoplasm, active transport in form of vesicular shuttling along cytoskeleton tracks, budding, fission and fusion of membrane tubules as well

as cytoplasmic streaming, all contribute to the complex intracellular traffic network. To describe this complexity in quantitative terms using mathematical models requires simplifying the observed phenomena to the most essential elements. For example, the cytoplasm has long been considered as a homogeneous fluid with Newton-viscosity. Accordingly, transport models describing, for example intracellular diffusion of macromolecules contained only a single, spatially invariant diffusion constant. Recent evidence obtained using quantitative fluorescence and electron microscopy, however, has revealed a hierarchically organized entangled network architecture of the cytoplasm. As a consequence, realistic transport models have to account for this structural organization including space-dependent diffusion constants and complex cell shapes.

3.3.1 Segmentation of FLIP images

The first step toward a realistic simulation is to segment the FLIP image into its nucleus Ω_N , the cytoplasm Ω_C and the nuclear membrane Ω_M . For later reference we denote the boundary of the nuclear membrane as $\partial\Omega_M$ and split $\partial\Omega_M$ into two disjoint sets $\partial\Omega_{MC}$ and $\partial\Omega_{MN}$, such that $\partial\Omega_{MC}$ is the boundary part that faces the cytoplasm and $\partial\Omega_{MN}$ faces nucleus. The bleaching area is located within the cytoplasm $\Omega_B \subset \Omega_C$. In this way, the computational domain Ω covers the full geometric complexity of the two-dimensional FLIP image. See Fig. 3.2 for a schematic illustration.

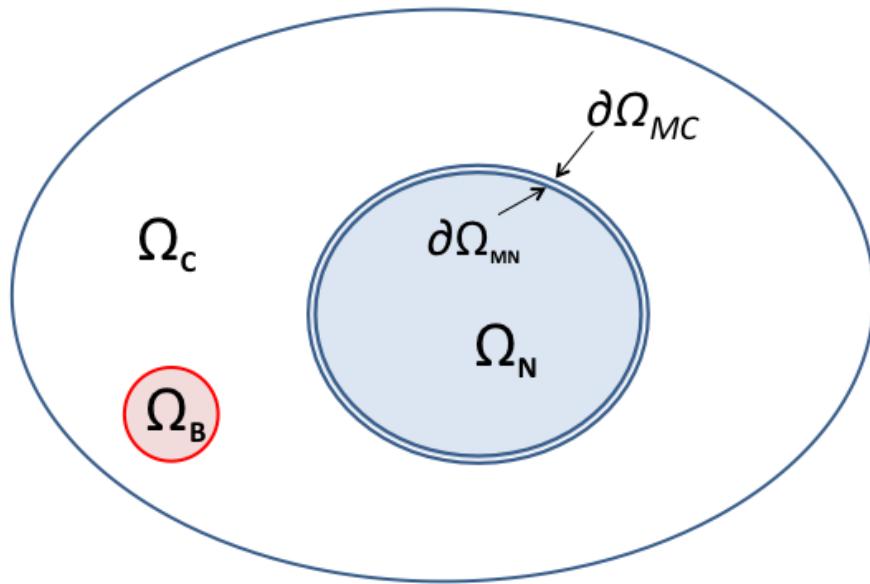


Figure 3.2: Schematic illustration of segmented cell.

For segmentation the Chan-Vese active contours algorithm [8] is applied which detects different gray-levels c_- and c_+ separated by a closed curve Γ . The separation line is represented as a zero-level curve

$$\Gamma = \{\mathbf{x} \in \mathbb{R}^2 | L(\mathbf{x}) = 0\} ,$$

such that $L < 0$ inside Γ . Note that

$$\mathbf{n} = \frac{\nabla L}{|\nabla L|} \quad (3.1)$$

is the outward unit normal and $\kappa = \nabla \cdot \mathbf{n}$ is the signed curvature. The optimal contour is characterized by minimizing the Chan-Vese energy functional

$$\begin{aligned} E(c_-, c_+, \Gamma) = & \mu \int_{\Gamma} d\sigma \\ & + \lambda_- \int_{L < 0} (I(\mathbf{x}) - c_-)^2 d\mathbf{x} \\ & + \lambda_+ \int_{L > 0} (I(\mathbf{x}) - c_+)^2 d\mathbf{x} . \end{aligned} \quad (3.2)$$

Here $I(\mathbf{x})$ is the gray-scale FLIP image and $\mu \geq 0$, $\lambda_{\pm} > 0$ are fixed parameters. The first integral on the right hand side measures the length of the curve and thereby has a smoothing effect. The remaining two integrals are minimized by matching the curve to the boundary of the object as well as the average gray-levels in fore- and background respectively.

The idea of Chan and Vese is to *activate* the curve by introducing an artificial time-like parameter $\Gamma(t) = \{\mathbf{x}(t) \in \mathbb{R}^2 : L(\mathbf{x}(t), t) = 0\}$ and let the curve *evolve* towards minimal energy. By definition $L_t = -\nabla L \cdot \mathbf{x}'$. Evolution towards less energy means $\mathbf{x}' = -\nabla_{\partial\Omega} E$, where the shape gradient of the energy (3.2), see Appendix A, is given by

$$\nabla_{\Gamma} E = [2\mu\kappa + \lambda_-(I(\mathbf{x}) - c_-)^2 - \lambda_+(I(\mathbf{x}) - c_+)^2] \cdot \mathbf{n} . \quad (3.3)$$

Consequently, the level set equation for the Chan-Vese active contour reads

$$L_t = [2\mu\kappa + \lambda_-(I(\mathbf{x}) - c_-)^2 - \lambda_+(I(\mathbf{x}) - c_+)^2] \cdot |\nabla L| .$$

The implementation of the Chan-Vese algorithm described in [14] is a time-stepping method for this evolutionary PDE. It is used to localize boundaries of the cell, nucleus and bleaching area in FLIP images, respectively. The iteration is stopped, when by visual inspection the contour reached the boundary of the object. The strength of the algorithm is that it can distinguish several disjoint objects in one image. The method does detect objects whose boundaries are not necessarily defined by gradients. Like in *snake-methods* [18] the energy is minimized until the desired boundary is detected.

The Chan–Vese algorithm applies to segment the cell from the background as well as the nucleus and the bleaching area from the cytoplasm. Both the cell– and the bleaching area are segmented from the first image while the nucleus is taken from the last image of the sequence depicted in Fig. 3.10 – 3.13 .

To simulate transport kinetics across the nuclear membrane using space dependent reaction rates, the nuclear boundary is "blown up" to a small but finite thickness of 9 nm which resembles closely the thickness of the nuclear membrane [30]. From the Chan–Vese method we obtain the boundary of the nucleus represented as a polygon. Each point on this polygon is shifted in both inward– and outward normal direction to form two new polygons $\partial\Omega_{MN}$ and $\partial\Omega_{MC}$, respectively. Finally, the nuclear membrane is bounded by $\partial\Omega_M = \partial\Omega_{MN} \cup \partial\Omega_{MC}$.

3.3.2 Transport kinetics across the membrane

The in FLIP images observed fluorescence intensity is a direct indicator for the amount of fluorescence–tagged molecules. A typical marker is eGFP, whose fluorescence intensity, denoted by c , is the quantity of interest in our model and subsequent simulation. Typical FLIP images show that the fluorescence intensity of eGFP is highest in the nuclear compartment even in the prebleach image. This suggests a higher amount of eGFP in the nucleus than in the cytoplasm at steady state (see Fig. 3.10). To model reactive transport of eGFP tagged molecules across the nuclear membrane, we need to distinguish between the intensities in the cytoplasm u and in the nucleus v . Consequently, $c = u + v$ is a "direct" sum in the sense that $u(t, \mathbf{x}) = 0$ in the nucleus $\mathbf{x} \in \Omega_N$ and $v(t, \mathbf{x}) = 0$ in Ω_C . Transport across the membrane is modeled by a first order reversible molecular reaction



where k^+ and k^- are positive reaction coefficients. Obviously the reaction (3.4) is just the simplest of its kind, but it will be sufficient for a proof of concept. By reaction kinetics, the dynamics of (3.4) is described by a differential system

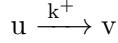
$$\frac{\partial}{\partial t} u(t, \mathbf{x}) = k^-(\mathbf{x})v(t, \mathbf{x}) - k^+(\mathbf{x})u(t, \mathbf{x}) , \quad (3.5)$$

$$\frac{\partial}{\partial t} v(t, \mathbf{x}) = k^+(\mathbf{x})u(t, \mathbf{x}) - k^-(\mathbf{x})v(t, \mathbf{x}) , \quad (3.6)$$

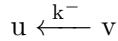
$$\mathbf{x} \in \Omega_M , \quad t > 0.$$

Clearly, $c = u + v$ is constant and mass is conserved.

To guarantee the complete transformation of u into v across the membrane, space dependent reaction rates are needed: Near the nucleus the reaction



dominates $k^+ \gg 1$, while the reversed reaction is slow $k^- \ll 1$. On the other side of the membrane, close to the cytoplasm



dominates $k^- \gg 1$ and $k^+ \ll 1$. Thus, both reaction coefficients do vary with the position \mathbf{x} throughout the membrane; $\pm k^\pm$ is falling from Ω_{MN} towards Ω_{MC} . This is justified not only from a numerical but also from a biophysical point of view of modeling nucleo-cytoplasmic transport. Indeed, by introducing a potential energy, Zilman and colleagues have recently modeled the transport of cargo through nuclear pore complexes as diffusion in an energy landscape [30]. Also, we introduce stiff reaction rates forcing (3.5) and (3.6) into equilibrium

$$k^+(\mathbf{x})u = k^-(\mathbf{x})v .$$

On the boundary $\partial\Omega_{MN}$ $k^-(\mathbf{x}) = 0$ and $k^+(\mathbf{x}) \gg 1$ such that $u = 0$ and all the tagged protein is present as v . Similarly, on the other side, $\partial\Omega_{MC}$, $k^+(\mathbf{x}) = 0$ and $k^-(\mathbf{x}) \gg 1$ such that $v = 0$ and all protein is transformed into u . Examples for rate coefficients k^\pm are logistic functions

$$k^\pm(\mathbf{x}) = \frac{1}{\epsilon} \frac{1}{1 + \exp(\pm k \cdot \text{dist}(\mathbf{x} - \mathbf{x}_m))} , \quad (3.7)$$

$$0 < \epsilon \ll 1 , \quad k \gg 1 .$$

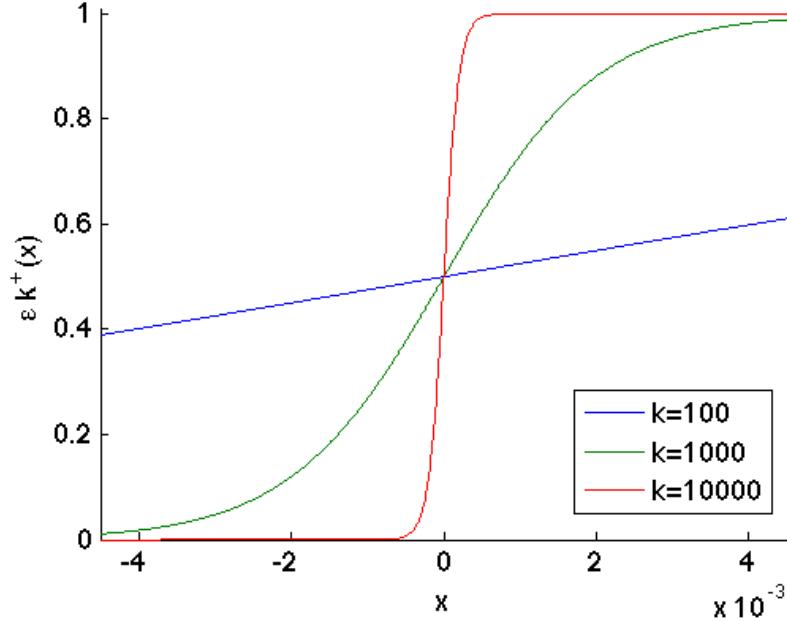
Here \mathbf{x}_m denotes the orthogonal projection of \mathbf{x} onto center of the membrane, such that $\text{dist}(\mathbf{x} - \mathbf{x}_m)$ is the signed distance from the center. This distance is positive in direction of the outward normal (3.1) pointing from the nucleus into the cytoplasm. Fig. 3.3 displays ek^+ for different values of the shape parameter k and over the typical width of a membrane: 9 nm. Note that the outward normal \mathbf{n} is positive in negative \mathbf{x} -direction.

For a discussion of alternative, single variable but discontinuous models we refer to Section 3.6.

3.3.3 Reaction-diffusion dynamics

From [21] we find that the photobleaching may be modeled by the differential equation:

$$\frac{\partial}{\partial t} u(t, \mathbf{x}) = -b \frac{q}{q+1} \cdot u(t, \mathbf{x}) , \quad \mathbf{x} \in \Omega_B , t > 0 , \quad (3.8)$$

Figure 3.3: Reaction rate ϵk^+ centered at origin.

where b is the intrinsic bleaching rate constant and q is the equilibrium constant for the reaction between the ground and excited state for a fluorophore. Since the excitation rate constant depends on laser intensity, q also depend on the intensity. This model allows to study the effect of fluorophore photophysics and eventual fluorescence saturation [21].

Using common notation, our time–dependent PDE model reads:

$$\begin{aligned} u_t &= \nabla \cdot (\alpha \nabla u) + \left(k^- v - k^+ u \right) \Big|_{\Omega_M} - \theta b \frac{q}{1+q} u \Big|_{\Omega_B}, \\ v_t &= \nabla \cdot (\beta \nabla v) + \left(k^+ u - k^- v \right) \Big|_{\Omega_M}, \\ \mathbf{x} \in \Omega, \quad t > 0, \end{aligned} \tag{3.9}$$

where θ is a 0–1 step function simulating the high–intensity laser pulses occurring at a frequency given by the frame rate. Diffusion coefficients are inhomogeneous and piecewise defined

$$(\alpha, \beta) = \begin{cases} (\alpha, \beta)_C & \text{if } \mathbf{x} \in \Omega_C, \\ (\alpha, \beta)_M & \text{if } \mathbf{x} \in \Omega_M, \\ (\alpha, \beta)_N & \text{if } \mathbf{x} \in \Omega_N. \end{cases} \tag{3.10}$$

To prevent flow across the cell boundary and thereby maintain mass conservation, the model comes with non-flow, Neumann boundary conditions

$$\mathbf{n} \cdot \nabla u = \mathbf{n} \cdot \nabla v = 0 , \quad \mathbf{x} \in \partial\Omega . \quad (3.11)$$

Finally, initial concentrations need to be specified

$$u(0, \mathbf{x}) = u^0(\mathbf{x}) , \quad v(0, \mathbf{x}) = v^0(\mathbf{x}) , \quad \mathbf{x} \in \Omega . \quad (3.12)$$

3.4 Implementation and Simulation

3.4.1 Numerical method

In complex geometries, like segmented FLIP cell images, it is convenient to generate finite element meshes using Gmsh [11]. The Finite Element Method is automated in the FEniCS software [20], which uses the UMFPACK sparse, direct solver via PETSc [10, 24]. Optional iterative and parallel solvers are available. All one needs to specify to the FEniCS compiler is the weak form of our FLIP model (3.9).

We discretize the time derivative by an backward Euler step. Any higher order but L -stable method, like f.ex. certain SDIRK schemes [13], are appropriate as well. Let ϕ and ψ be bilinear Lagrange test-functions for u and v respectively. After multiplication and partial integration the weak form reads:

$$\begin{aligned} & \int_{\Omega} \frac{u^{n+1} - u^n}{\Delta t} \phi \, d\mathbf{x} + \int_{\Omega} \alpha \nabla u^{n+1} \cdot \nabla \phi \, d\mathbf{x} \\ & + \int_{\Omega_M} (k^+ u^{n+1} - k^- v^{n+1}) \phi \, d\mathbf{x} \\ & + \int_{\Omega_B} \theta b \frac{q}{q+1} u^{n+1} \phi \, d\mathbf{x} = 0 , \end{aligned} \quad (3.13)$$

$$\begin{aligned} & \int_{\Omega} \frac{v^{n+1} - v^n}{\Delta t} \psi \, d\mathbf{x} + \int_{\Omega} \beta \nabla v^{n+1} \cdot \nabla \psi \, d\mathbf{x} \\ & + \int_{\Omega_M} (k^- v^{n+1} - k^+ u^{n+1}) \psi \, d\mathbf{x} = 0 . \end{aligned} \quad (3.14)$$

For faster execution, the bilinear form is pre-assembled. The pulsating laser beam is realized by pre-assembling two systems, with and without the bleaching term and iterating both. To resolve the effect of bleaching, the bleaching interval is a multiple of the time step: $\Delta t_b = n\Delta t$.

3.4.2 Tuning reaction– and diffusion parameters

To find reasonable reaction–diffusion rates, consider a one-dimensional model problem along the cross-section of a cell membrane of 9nm width.

$$\begin{aligned} u_t &= (\alpha u_x)_x + k^- v - k^+ u , \\ v_t &= (\beta v_x)_x + k^+ u - k^- v , \quad x \in \Omega , \quad t > 0 . \end{aligned}$$

Measuring α, β in $\mu\text{m}^2/\text{s}$, k^\pm in s^{-1} and x in μm , the domain of interest is the interval $\Omega = [0, 0.029]$. The membrane of width 9 nm is located at $\Omega_M = [0.01, 0.019]$, the cytoplasm $\Omega_C = [0, 0.01)$, and the nucleus $\Omega_N = (0.019, 0.029]$. As initial data we set

$$u^0 = \begin{cases} 1 & \text{if } x \in \Omega_C \\ 0.5 & \text{if } x \in \Omega_M , \\ 0 & \text{if } x \in \Omega_N \end{cases} \quad v^0 = \begin{cases} 0 & \text{if } x \in \Omega_C \\ 0.5 & \text{if } x \in \Omega_M \\ 1 & \text{if } x \in \Omega_N \end{cases} .$$

Note that $u^0 + v^0 = 1$. From the literature [29] we know that eGFP diffuses with $\approx 25 \mu\text{m}^2/\text{s}$, thus the diffusion rates are set to

$$\alpha = \begin{cases} 25 & \text{if } x \in \Omega_C \\ 5 \cdot 10^{-4} & \text{if } x \in \Omega_M , \\ 10^{-3} & \text{if } x \in \Omega_N \end{cases} , \quad (3.15)$$

$$\beta = \begin{cases} 10^{-3} & \text{if } x \in \Omega_C \\ 2.5 \cdot 10^{-4} & \text{if } x \in \Omega_M , \\ 25 & \text{if } x \in \Omega_N \end{cases} . \quad (3.16)$$

The stiffness parameter ϵ in (3.7) is set to $\epsilon = 10^{-4}$. For different values of

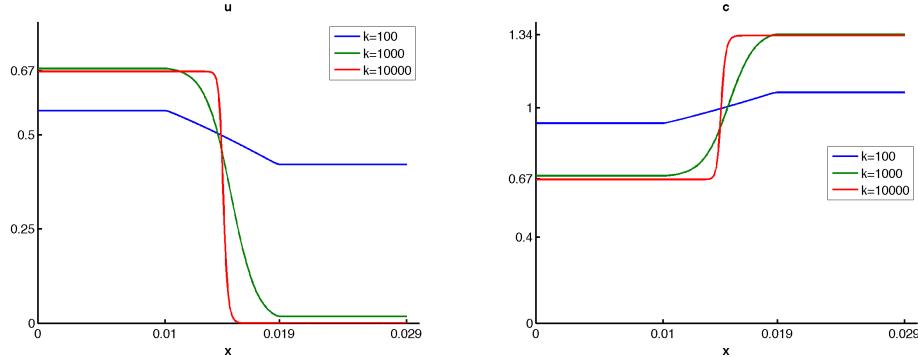


Figure 3.4: u at time $t = 5$ sec.

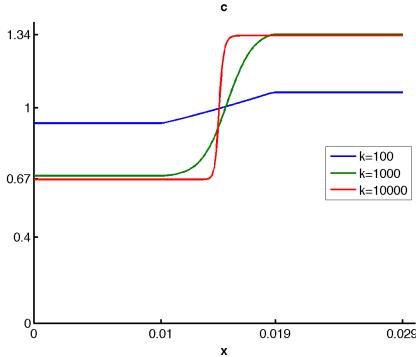


Figure 3.5: $c = u + v$ at $t = 5$ sec.

the shape coefficient we observe the eGFP fluorescence intensities as depicted in Fig. 3.4 and 3.5 at time $t = 5$.

In agreement with Appendix B Corollary 3.7.3, positivity is preserved — a convenient property when computing intensities. From Fig. 3.4 we conclude that for u and v to vanish in the cytoplasm and the nucleus respectively, values of $k \approx 10^4$ and $\epsilon \lesssim 10^{-4}$ are sufficient. Rates of $k \approx 10^3$ and less cannot completely transform u into v across the membrane.

The higher amount of total eGFP in the nucleus as seen in Fig. 3.5, is caused by the trans–membrane kinetics. It is confirmed in our two–dimensional simulation and agrees with typical FLIP images, see Section 3.4.3. It might be worth noting that the relation of fluorescence intensities in cytoplasm and nucleus agrees with the ratio of diffusion rates in the membrane

$$2 = \frac{\alpha_M}{\beta_M} \approx \frac{[c_N]}{[c_C]} . \quad (3.17)$$

3.4.3 Simulation of FLIP images

Fig. 3.10 – 3.13 shows a sequence of FLIP images in McArdle RH7777 cells. Enhanced green fluorescent protein (eGFP) stably expressed in the cells was repeatedly bleached on temperature controlled ($35 \pm 1^\circ\text{C}$) stage of a Zeiss LSM 510 confocal microscope using the 488-nm line of an Argon laser as excitation source. Bleaching was performed for 5 iterations in the indicated region (circle), while images were acquired with 0.5% laser power to avoid bleaching during image recording. The total frame rate inclusive bleaching was 1.6 sec. and the size of the displayed cell is approximately $10 \times 16 \mu\text{m}$.

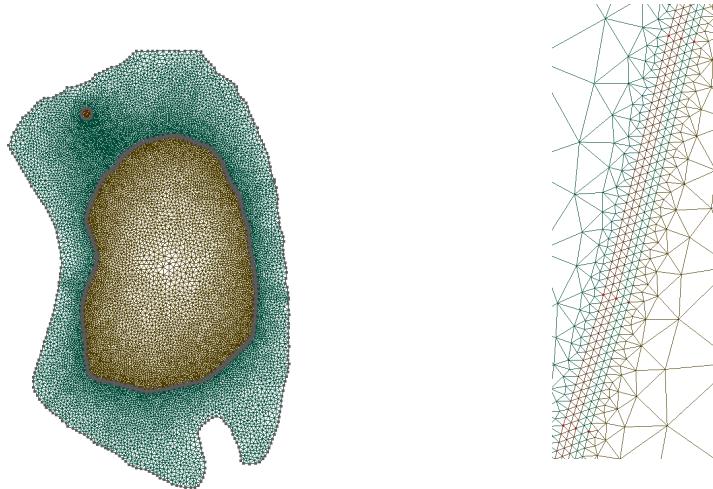


Figure 3.6: Finite element mesh on Chan–Vese active contours generated with gmsh [11].

Figure 3.7: Close-up illustration of the mesh at the membrane.

To simulate the experiment, we compute the FLIP model (3.7), (3.9) and (3.10) on the mesh displayed in Fig. 3.6. The mesh, consisting of 221558 trian-

gles (68534 located in cytoplasm, 114 in bleaching area, 89077 in membrane, and 63833 in nucleus), was generated using Gmsh [11] on the Chan–Vese contours. The bleaching area is visible in the upper left corner. The model parameters are: $\epsilon = 10^{-4}$, $k = 10^4$, $bq/(1 + q) = 160 \text{ s}^{-1}$ with diffusion rates given by (3.15) and (3.16). Initial data as seen in Fig. 3.14 are logistic functions across the membrane

$$u^0 = \frac{53.9}{1 + \exp(-k \cdot \text{dist}(\mathbf{x} - \mathbf{x}_m))},$$

$$v^0 = \frac{119.4}{1 + \exp(k \cdot \text{dist}(\mathbf{x} - \mathbf{x}_m))}.$$

The max-values are taken as intensity averages from the initial FLIP image. The bleaching time interval is $\Delta t_b = 0.4 \text{ sec}$, the recovery phase lasts for 1.2 sec resulting in a total frame rate of 1.6 sec. The time step is chosen as $\Delta t = 0.1 \text{ sec}$.

In both the FLIP images and the simulation shown in Fig. 3.14-3.17, we observe a stronger effect of bleaching in the cytoplasm than in the nucleus — the membrane shows its protective effect.

In FLIP images we observe spatially resolved structures especially within the nucleus. Our model however, based on piecewise constant diffusion coefficients approximates averages within the different compartments.

Simulated average intensities are depicted in Fig. 3.8 and 3.9. Even though simulated intensities follow the observations, initially the intensity in the cytoplasm drops faster than observed in FLIP images. This defect may be attenuated by further (automated) tuning of reaction and diffusion rates in future work.

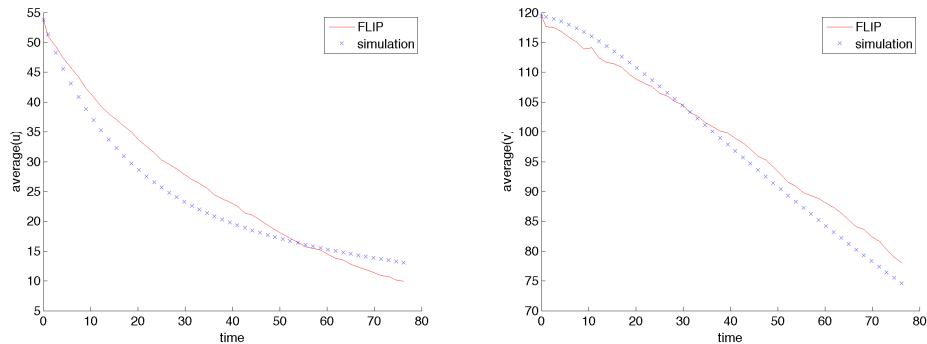


Figure 3.8: Average fluorescence intensity in cytoplasm over time.

Figure 3.9: Average fluorescence intensity in nucleus over time.

Observed FLIP intensities are linearly correlated with the computational model, as seen in Fig. 3.22 and 3.23. The correlation coefficients for cytoplasm and nucleus are 0.980 and 0.998, respectively.

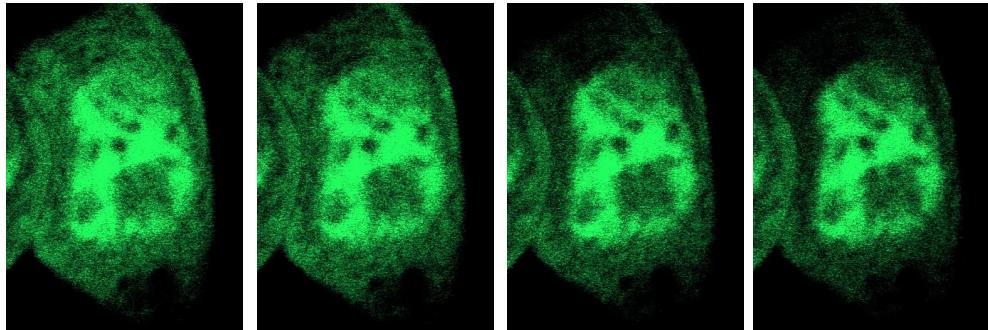


Figure 3.10: $t=0\text{s}$ Figure 3.11: $t=2.6\text{s}$ Figure 3.12: $t=17\text{s}$ Figure 3.13: $t=33\text{s}$

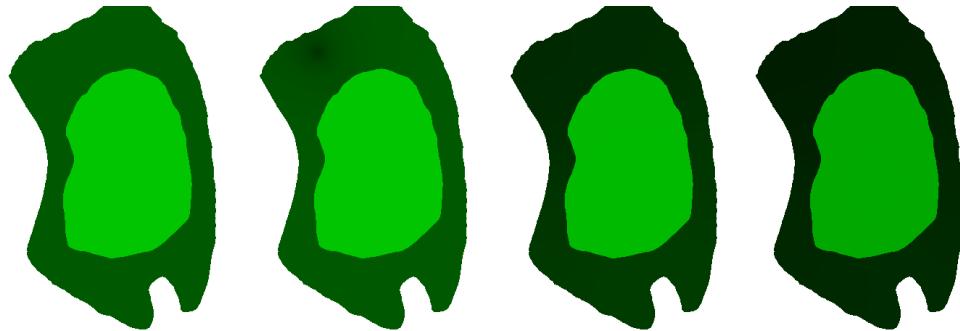


Figure 3.14: $t=0\text{s}$ Figure 3.15: $t=2.6\text{s}$ Figure 3.16: $t=17\text{s}$ Figure 3.17: $t=33\text{s}$

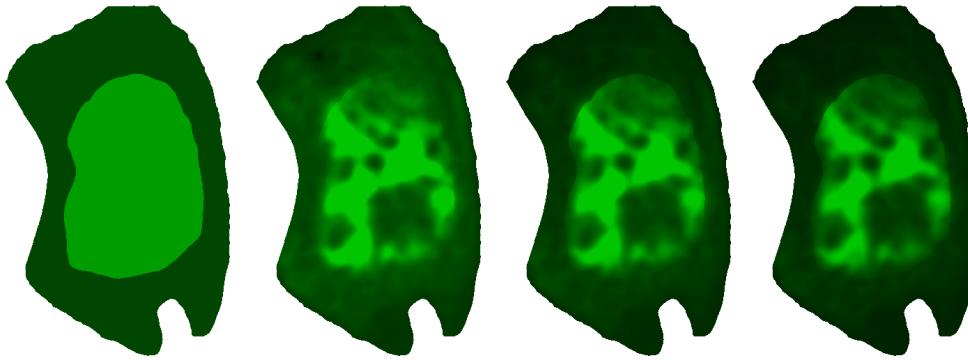


Figure 3.18: $t=0\text{s}$ Figure 3.19: $t=2.6\text{s}$ Figure 3.20: $t=17\text{s}$ Figure 3.21: $t=33\text{s}$

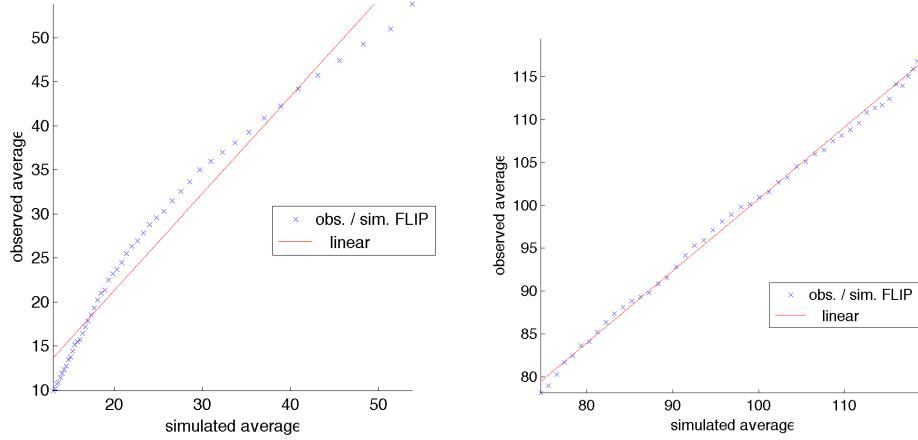


Figure 3.22: Correlation of the average fluorescence intensity in cytoplasm. Figure 3.23: Correlation of the average fluorescence intensity in nucleus.

In Section 3.4.2 we observed that the ratio of average fluorescence intensities in the cytoplasm and nucleus is approximately determined by the ratio of membrane diffusion rates (3.17). This observation is confirmed in a 2D simulation when bleaching is turned off $b = 0$. In Fig. 3.24 it is seen that the ratio of averaged intensities approaches the ratio of diffusion rates $\frac{\alpha_M}{\beta_M} = 2$.

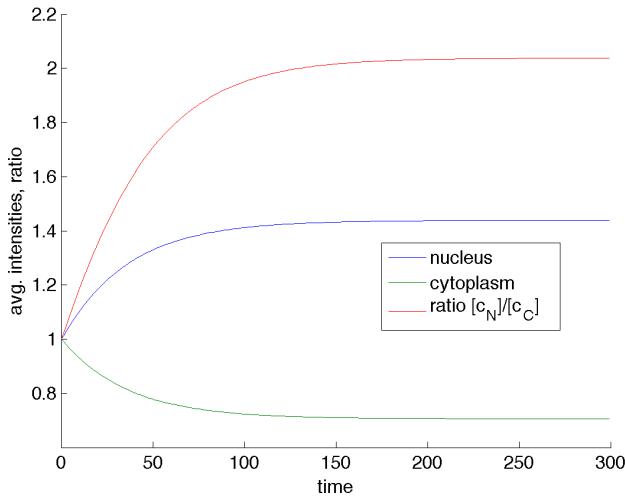


Figure 3.24: Average intensities and ratio over time.

3.4.4 A reaction-diffusion model with hindrance

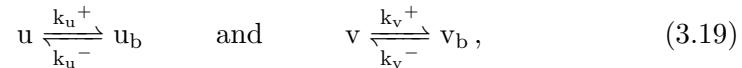
Currently, much research effort is on understanding the phenomenon of molecular crowding in living cells and on elucidating the underlying causes and the consequences of crowding. By a variety of fluorescence based imaging techniques including FRAP and FLIP, evidence for restricted motion of even non-interacting proteins, as eGFP has been provided [3, 4, 15, 29]. Especially in the nucleus, eGFP was found to perform diffusive motion as on a fractal, since the compact ribonuclein structure severely hindered its mobility [3, 15].

Pixel-wise FLIP analysis gave support for this model and found a negative correlation between DNA content (visualized by Hoechst staining) in the nucleus and fluorescence intensity of eGFP [29]. On the other side, nuclear areas with high amounts of eGFP experienced a slower fluorescence loss than areas with low eGFP, suggesting that the surrounding DNA and nuclear proteins hinder escape of eGFP from these regions. Such nuclear compartmentalization is supposed to play a central role in gene regulation [4]. Realistic FLIP simulations therefore need to account for an uneven distribution of nuclear proteins. We therefore extended the model described in the previous section by introducing u_b and v_b which represent molecules that are hindered in their diffusion and will, thus, have a lower diffusion rate than u and v .

Like for u and v , u_b and v_b only resides in the cytoplasm and nucleus, respectively. Therefore they also follow the membrane transport process presented in Section 3.3.2 where:



In areas with high intensities, eGFP is hindered in its motion, thus in these areas u and v will be transformed into u_b and v_b , which is described by the following reaction mechanism



where k_u^+ , k_u^- , k_v^+ and k_v^- are spatially resolved positive reaction rates, modeling binding and release, respectively. Extending our PDE model from (3.9) with the reaction kinetics from (3.18) and (3.19) yields the system

$$\begin{aligned}
u_t &= \nabla \cdot (\alpha \nabla u) + k_u^- u_b - k_u^+ u \\
&\quad + (k^- v - k^+ u) \Big|_{\Omega_M} - \theta b \frac{q}{1+q} u \Big|_{\Omega_B} , \\
(u_b)_t &= \nabla \cdot (\alpha_b \nabla u_b) + k_u^+ u - k_u^- u_b \\
&\quad + (k_b^- v_b - k_b^+ u_b) \Big|_{\Omega_M} - \theta b \frac{q}{1+q} u_b \Big|_{\Omega_B} , \\
v_t &= \nabla \cdot (\beta \nabla v) + k_v^- v_b - k_v^+ v + (k^+ u - k^- v) \Big|_{\Omega_M} , \\
(v_b)_t &= \nabla \cdot (\beta_b \nabla v_b) + k_v^+ v - k_v^- v_b \\
&\quad + (k_b^+ u_b - k_b^- v_b) \Big|_{\Omega_M} , \\
\mathbf{x} \in \Omega \quad t > 0 \quad .
\end{aligned} \tag{3.20}$$

The diffusion rates α and β are as in (3.15) and (3.16) and for the hindered molecules the diffusion rates are chosen to be significant smaller than the typical rate $25\mu\text{m}^2/\text{s}$ for eGFP. The values for k_u^+ and k_v^+ is chosen to be proportional to the pixel intensities in Fig. 3.10, which is the equilibrium stage in the cell before bleaching. Thus, let I be the normalized intensities from Fig. 3.10, then $k_u^+ = k_v^+ = \nu I$, where $\nu > 0$ is a constant. Furthermore k_u^- and k_v^- is set to be $k_u^- = k_v^- = \nu - \rho I$, where $\nu - \rho > 0$ to ensure that the reaction processes from (3.19) is reversible in high intensity areas. By use of the approach from Section 3.4 and based on visual inspection choosing $\nu = 30$, $\rho = 25$ and the diffusion constants as

$$\alpha_b = \begin{cases} 0.1 & \text{if } x \in \Omega_C \\ 5 \cdot 10^{-4} & \text{if } x \in \Omega_M \\ 10^{-3} & \text{if } x \in \Omega_N \end{cases} , \tag{3.21}$$

$$\beta_b = \begin{cases} 10^{-3} & \text{if } x \in \Omega_C \\ 2.5 \cdot 10^{-4} & \text{if } x \in \Omega_M \\ 0.1 & \text{if } x \in \Omega_N \end{cases} , \tag{3.22}$$

we obtain the simulation results presented in Fig. 3.18 – 3.21. Apparently the hindrance mechanism described above generates fluorescence intensities that agree well with the original FLIP sequence.

The simulation cannot only reconcile the experimentally observed heterogeneity in eGFP distribution (compare Figs. 3.11 - 3.13 with Fig. 3.19 - 3.21). They also show establishment of a depletion zone surrounding the bleached

area; a prominent sign for a diffusion-limited bleaching process (see Table 1 and accompanying text in [29] for the possible scenarios in FLIP experiments). For the same experimental FLIP sequence as shown in Figs. 3.10 - 3.13 we demonstrated previously, that transport of eGFP to the bleach spot is diffusion-limited, as a pixel-wise fitting of a stretched/compressed exponential function gave a gradient to lower values of the stretching parameter ($h < 1$) far from the bleach spot (see Fig. 3C and E in [29]). Crossover from a stretched exponential to a compressed exponential process as function of distance from the bleach spot is an empirical measure of diffusion limited FLIP, as we verified in additional simulations in our previous study [29]. By performing a pixel-by-pixel fit of this function, i.e. $f(\mathbf{x}, t) = A(\mathbf{x}) \exp\left(-\left(\frac{t}{\tau(\mathbf{x})}\right)^{\frac{1}{h(\mathbf{x})}}\right)$, to the simulated FLIP image sequence based on the reaction-diffusion model with hindrance, we could well recapitulate the stretching parameter map $h(\mathbf{x})$ observed in experiments (not shown). Thus, with our model and the used parameter combination, we can reconcile the scenario of diffusion-limited FLIP experiments giving the high coincidence of experimental data and simulation.

3.5 Convergence test

Consider a rectangle domain which is $1.3 \mu m$ long and $0.2 \mu m$ high with a membrane width of $9 nm$ centred in $x = 0.6045 \mu m$. The rectangle represents a subsection of the cell geometry. For the reference function a mesh with 289,996 triangles as shown in Fig. 3.25 have been used.

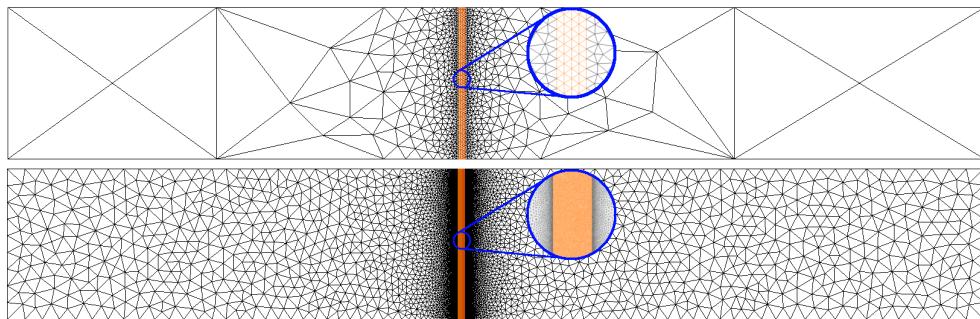


Figure 3.25: Coarsest and finest mesh with respectively 1,908 and 289,996 triangles for convergence test.

h_{max}	triangles	est. error	time ¹ [sec]
0.448469	1,908	0.568714	6.84
0.107540	20,444	0.073679	49.78
0.067564	56,266	0.022497	156.61
0.047757	112,496	0.008966	383.03
0.039125	193,606	0.003697	738.13

Table 3.1: Convergence test results.

The system tested is the system with backward Euler step from (3.13) and (3.14) where $\Delta t = 0.05$, $b = 0$ and final time $T = 10$. The diffusion and reaction parameters are as in Section 3.4.3 and $c^0 = u^0 + v^0 = 100$ where u^0 and v^0 are logistic functions as described in previous sections. The results from the convergence test is presented in Table 3.1 and Fig. 3.26, where the estimated error clearly decays as the number of gridpoints grows and the maximal celldiameter h_{max} gets smaller. A least square fit of the data in Fig. 3.26 gives an estimated order of convergence at 1.97.

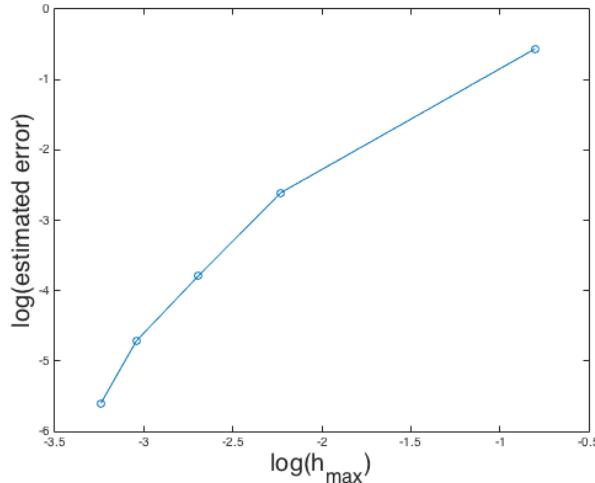


Figure 3.26: Log-log-plot of estimated errors.

3.6 Discussion and conclusion

We describe a computational reaction-diffusion model of FLIP image sequences of eGFP transport between the cytoplasm and nucleus of mammalian cells.

¹Computation time on a 2,4 GHz i5, 8 GB DDR3, MacBook ultimo 2013.

Our model performs these simulations in the 2D cell geometry extracted from microscopy data gathered on a confocal microscope.

Using a 2D model for the real 3D cell geometry is a common assumption being valid as long as the cell is flat and the image acquisition gathered data from a thin cellular region (typical settings for the pinhole are about 3 Airy disks corresponding to a depth of field of the recorded fluorescence of about $2.5\mu m$). However, we are aware of the fact, that local variation in probe diffusivity and abundance can be a consequence of 2D imaging of the 3D trajectories [12]. As the FEniCS software applies to higher dimensions, also 3D simulations are feasible in future work. Then it might be possible to investigate 2D imaging effects.

In agreement with theoretical analysis i.e. Corollary 3.7.3 in Appendix B, the model preserves positivity of intensities. Also, we found that our simulation results are correlated with experimental observations. The reaction diffusion model with hindrance (3.20) shows spatially resolved intensities that agree well with original FLIP images. We may conclude that our simulations support the observation that compact ribonuclein structure locally reduces diffusive mobility of eGFP.

Interestingly, the simulations suggest that a slowed diffusion of eGFP from the nucleus into the cytoplasm compared to the opposite direction can account for a higher residence of the protein in the nuclear compartment at steady state.

Piecewise linear, continuous Finite Elements are second order accurate in space. To establish a uniformly second order method in space and time, the backward Euler–step is to be replaced by any L–stable scheme, for example the two stage SDIRK method [13].

The simple first order model for membrane transport (3.4) is only a starting point. To resolve the stiff reaction in space and time is questionable and modeling the fluorescence intensity of eGFP by two variables u and v is redundant. Alternatively, the jump in intensity across the membrane may be treated as discontinuity. Due to the additional degrees of freedom discontinuous Galerkin methods are about 5–20 times more computational expensive than continuous Finite Elements [28]. In return the membrane does not need to be resolved and large reaction coefficients are avoided. The investigation of advanced and nonlinear membrane transport models in the context of discontinuous Galerkin methods is subject of future work.

The FLIP simulator presented in this paper can serve as forward solver in an inverse analysis to quantify diffusion and transport rates and thus to calibrate the model to experimentally observed images. Given the fact that the cytoplasm is highly heterogeneous and diffusion or binding parameters vary locally, likely on a nano scale, computational analysis of FLIP experiments discussed in this paper bear potential for a better quantitative description of intracellular diffusion and transport processes.

3.7 Appendix

Appendix A: Shape gradients

The Chan-Vese energy (3.2) consists of two parts: Minimizing the length of the curve has some regularizing effect $E_{\text{reg}}(\Gamma) = \int_{\Gamma} d\sigma$. Without loss of generality, we may assume parametrization with respect to arc-length

$$\Gamma = \gamma(\sigma) : [0, l] \rightarrow \mathbb{R}^2$$

such that $|\gamma'| = 1$ and $l = \int_{\Gamma} d\sigma = \int_0^l |\gamma'(\sigma)|^2 d\sigma$. Let ω be another closed, C^1 -curve $\omega : [0, l] \rightarrow \mathbb{R}^2$ such that $\omega(0) = \omega(l)$ and $\omega'(0) = \omega'(l)$. Consider an ϵ -variation of the curve Γ given by $\gamma_{\epsilon} = \gamma + \epsilon\omega$. The variation of $E_{\text{reg}} = \int_0^l |\gamma'(\sigma)|^2 d\sigma$ is defined as

$$\delta E_{\text{reg}}(\gamma, \omega) = \frac{d}{d\epsilon} E_{\text{reg}}(\gamma_{\epsilon}) \Big|_{\epsilon=0}$$

By differentiation we find

$$\begin{aligned} \delta E_{\text{reg}}(\gamma, \omega) &= \frac{d}{d\epsilon} \int_0^l |\gamma'(\sigma) + \epsilon\omega'(\sigma)|^2 d\sigma \Big|_{\epsilon=0} \\ &= 2 \int_0^l (\gamma'(\sigma) + \epsilon\omega'(\sigma)) \omega'(\sigma) d\sigma \Big|_{\epsilon=0} \\ &= 2 \int_0^l \gamma'(\sigma) \omega'(\sigma) d\sigma . \end{aligned}$$

By partial integration, using periodicity of γ' and ω

$$\delta E_{\text{reg}}(\gamma, \omega) = -2 \int_0^l \gamma''(\sigma) \omega(\sigma) d\sigma .$$

By definition, the L^2 -shape gradient of E_{reg} is

$$\nabla_{\Gamma} E_{\text{reg}}(\gamma) = -2\gamma'' .$$

In arc-length parametrization $\gamma'' = -\kappa \cdot \mathbf{n}$, where \mathbf{n} is the outward unit normal. Finally

$$\nabla_{\Gamma} E_{\text{reg}}(\gamma) = 2\kappa \mathbf{n} . \quad (3.23)$$

The fitting energy is

$$E_{\text{fit}}(\gamma) = \int_{\text{int}\Gamma} (I(\mathbf{x}) - c_-)^2 d\mathbf{x} + \int_{\text{ext}\Gamma} (I(\mathbf{x}) - c_+)^2 d\mathbf{x} .$$

The integral over the exterior segment equals the integral over the entire image without the interior part

$$\begin{aligned} E_{\text{fit}}(\gamma) &= \int_{\text{int}\Gamma} (I(\mathbf{x}) - c_-)^2 - (I(\mathbf{x}) - c_+)^2 \, d\mathbf{x} \\ &\quad + \int_{\text{int}\Gamma \cup \text{ext}\Gamma} (I(\mathbf{x}) - c_+)^2 \, d\mathbf{x} . \end{aligned}$$

The last integral does not depend on the curve Γ and the first one is subject of the next Lemma.

Lemma 3.7.1. *Let γ and ω be closed C^1 -curves. Let $V : \mathbb{R}^2 \rightarrow \mathbb{R}$ be integrable. The variation of*

$$E(\gamma) = \int_{\text{int}\gamma} V(\mathbf{x}) \, d\mathbf{x}$$

is

$$\delta E(\gamma, \omega) = \int_0^l V(\gamma(\sigma)) \mathbf{n}(\sigma) \cdot \omega(\sigma) \, d\sigma .$$

The proof below shows that for Lemma 3.7.1 to hold true, arc-length parametrization is not necessary but convenient. Otherwise the unit normal n is to be replaced by the non-unit outward normal. As a result we have that the L^2 -shape gradient is given by

$$\nabla_\Gamma E(\gamma) = V(\gamma) \mathbf{n} . \quad (3.24)$$

Proof of Lemma 3.7.1:

Let $P(x_1, x_2) = 0$ and $Q(x_1, x_2) = \int_0^{x_1} V(\xi, x_2) \, d\xi$.

$$E(\gamma) = \int_{\text{int}\gamma} V(\mathbf{x}) \, d\mathbf{x} = \int_{\text{int}\gamma} \frac{\partial Q}{\partial x_1} - \frac{\partial P}{\partial x_2} \, dx_1 \, dx_2 .$$

By Green's formula

$$E(\gamma) = \int_\gamma Q \, dx_2 .$$

In the last curve integral $\mathbf{x} = \gamma$, $dx_2 = \gamma'_2 \, d\sigma$ and thus

$$E(\gamma) = \int_0^l Q(\gamma(\sigma)) \gamma'_2(\sigma) \, d\sigma .$$

Consider the ϵ -variation $\gamma_\epsilon = \gamma + \epsilon\omega$ and find

$$\begin{aligned} \delta E(\gamma, \omega) &= \frac{d}{d\epsilon} E(\gamma_\epsilon) \Big|_{\epsilon=0} \\ &= \frac{d}{d\epsilon} \int_0^l Q(\gamma_\epsilon) (\gamma'_\epsilon)_2 \, d\sigma \Big|_{\epsilon=0} \\ &= \int_0^l \nabla Q(\gamma_\epsilon) \cdot \omega(\gamma'_\epsilon)_2 + Q(\gamma_\epsilon) \omega'_2 \, d\sigma \Big|_{\epsilon=0} \\ &= \int_0^l \nabla Q(\gamma) \cdot \omega \gamma'_2 + Q(\gamma) \omega'_2 \, d\sigma . \end{aligned}$$

By partial integration and periodicity

$$\begin{aligned}
\delta E(\gamma, \omega) &= \int_0^l \nabla Q(\gamma) \cdot \omega \gamma'_2 - \frac{d}{d\sigma} (Q(\gamma)) \omega_2 d\sigma \\
&= \int_0^l \nabla Q(\gamma) \cdot \omega \gamma'_2 - \nabla Q(\gamma) \cdot \gamma' \omega_2 d\sigma \\
&= \int_0^l \frac{\partial Q}{\partial x_1} \mathbf{n} \omega d\sigma \\
&= \int_0^l V(\gamma) \mathbf{n} \omega d\sigma .
\end{aligned}$$

Here the (unit) normal is given by $\mathbf{n} = (\gamma'_2, -\gamma'_1)^T$ and our proof is complete. \square

The shape gradient of the Chan–Vese energy (3.3) follows from (3.23) and (3.24) where $V(\mathbf{x}) = (I(\mathbf{x}) - c_-)^2 - (I(\mathbf{x}) - c_+)^2$.

Appendix B: Stability analysis

For simplicity of presentation we consider a reaction–diffusion system of type (3.9), but in one spatial variable and with diffusion coefficients scaled to unity

$$\begin{aligned}
u_t &= u_{xx} + k^- v - k^+ u , \\
v_t &= v_{xx} + k^+ u - k^- v , \quad x \in (0, 1) , \quad t > 0 .
\end{aligned} \tag{3.25}$$

For illustration we consider homogenous Dirichlet and Neumann conditions for u and v , respectively

$$\begin{aligned}
u(t, 0) &= u(t, 1) = 0 , \\
v_x(t, 0) &= v_x(t, 1) = 0 .
\end{aligned}$$

The interested reader can easily replace the Dirichlet condition by a Neumann condition as in (3.11). Initial data is given as in (3.12). The analysis uses elements by E. Bohl [5] which originally go back to R.S. Varga [27], L. Collatz [9] and E. Bohl [6]. We have chosen to present this concept, because it generalizes to two dimensions and applies to non–linear reaction diffusion problems.

The bilinear form

$$\int_{\Omega} \Delta u \phi dx = - \int_{\Omega} \nabla u \cdot \nabla \phi dx$$

acting on a finite dimensional approximation $u(x) = \sum_j u_j \phi_j(x)$ with piecewise linear elements $\phi_j(x_i) = \delta_{ij}$ is expressed as matrix–vector product

$$- \int_{\Omega} \nabla u \cdot \nabla \phi_j dx = (\mathbf{A}_h \mathbf{u})_j .$$

The vector \mathbf{u} holds approximations on inner grid points $\mathbf{u} = (u_i)$, $u_i = u(x_i)$, $x_i = ih$, $i = 1, 2, \dots, m$, and $(m+1)h = 1$. The matrix is given by

$$\mathbf{A}_h = (a_{ij}) \quad , \quad a_{ij} = - \int_{\Omega} \nabla \phi_i \cdot \nabla \phi_j \, dx \quad .$$

Due to the piecewise constant gradients $\nabla \phi_j$, the matrix becomes a well-known (finite difference) approximation to the second derivative. With homogenous Dirichlet conditions the linear operator reads

$$-u_{xx} \approx \mathbf{A}_h \mathbf{u} = \frac{1}{h^2} \begin{pmatrix} 2 & -1 & & & \\ -1 & \ddots & \ddots & & \\ & \ddots & \ddots & -1 & \\ & & -1 & 2 & \end{pmatrix} \mathbf{u} \quad .$$

In the case of homogenous Neumann conditions $v_0 = v_1$ and $v_m = v_{m+1}$, the second derivative on inner grid points is approximated by

$$-v_{xx} \approx \mathbf{B}_h \mathbf{v} = \frac{1}{h^2} \begin{pmatrix} 1 & -1 & & & \\ -1 & 2 & -1 & & \\ & \ddots & \ddots & \ddots & \\ & & -1 & 2 & -1 \\ & & & -1 & 1 \end{pmatrix} \mathbf{v} \quad .$$

Semi-implicit time stepping results in a linearly coupled, discrete system

$$\begin{aligned} [(1 + \Delta t k^+) \mathbf{I} + \Delta t \mathbf{A}_h] \mathbf{u}^{n+1} &= \mathbf{u}^n + \Delta t k^- \mathbf{v}^n \quad , \\ [(1 + \Delta t k^-) \mathbf{I} + \Delta t \mathbf{B}_h] \mathbf{v}^{n+1} &= \mathbf{v}^n + \Delta t k^+ \mathbf{u}^n \quad . \end{aligned} \quad (3.26)$$

Naturally, step sizes are positive $\Delta t, h > 0$ and reaction rates are non-negative $k^\pm \geq 0$. Again, u_i^n is thought as approximation to $u(t_n, x_i)$. In fact one can prove that the numerical approximation converges to the solution of the PDE system.

The key argument relies on the fact that the finite difference operators $\mathbf{I} + \Delta t \mathbf{A}_h$ as well as $\mathbf{I} + \Delta t \mathbf{B}_h$ are inverse-monotone with bounded inverse. While \mathbf{A}_h inherits its inverse-monotonicity from the negative Laplacian, \mathbf{B}_h is obviously singular.

Lemma 3.7.2. *With non-negative parameters $\Delta t \geq 0$, $h > 0$ and reaction rates $k^\pm \geq 0$ both system matrices $(1 + \Delta t k^+) \mathbf{I} + \Delta t \mathbf{A}_h$ and $(1 + \Delta t k^-) \mathbf{I} + \Delta t \mathbf{B}_h$ are regular with non-negative and uniformly bounded inverse.*

An obvious consequence is:

Corollary 3.7.3. *With non-negative initial data $u^0 \geq 0$ and $v^0 \geq 0$ and reaction rates $k^\pm \geq 0$, approximations computed by (3.26) remain non-negative for all discrete time levels $n \Delta t \geq 0$.*

The proof of Lemma 3.7.2 is based on inverse-monotone Z -matrices. A Z -matrix is characterized by its non-positive off-diagonal elements:

$$\mathbf{Z} = (z_{ij}) \text{ , } z_{ij} \leq 0 \text{ , } i \neq j .$$

A matrix \mathbf{A} is called inverse-monotone if it is invertible with elementwise non-negative inverse $\mathbf{A}^{-1} \geq 0$. In this case $\mathbf{A}\mathbf{e} \geq 0$ implies $\mathbf{e} \geq 0$; i.e. solving the system preserves positivity. An inverse-monotone Z -matrix is called M -matrix.

Any strictly positive vector $\mathbf{p} > 0$ defines a weighted max-norm

$$\|\mathbf{x}\|_p = \max_i \{|x_i|/p_i\} .$$

The corresponding operator norm for a monotone (i.e. elementwise positive) matrix is $\|\mathbf{A}\|_p = \|\mathbf{A}\mathbf{p}\|_p$.

The following M -criterion can be found in E. Bohl [5], Chapter I, Theorem 5.1:

Lemma 3.7.4. *A Z -matrix \mathbf{A} is M -matrix if and only if \mathbf{A} is semi-positive. That is, there exists $\mathbf{e} > 0$ with $\mathbf{A}\mathbf{e} > 0$.*

Obviously, adding a non-negative diagonal matrix to a M -matrix preserves the M -property. With these preparations, we are ready to prove the main stability result.

Proof of Lemma 3.7.2:

Let $\delta = (1, \dots, 1)^T$. Note that $\mathbf{A}_h\delta = (1, 0, \dots, 0, 1)/h^2$ and $B_h\delta = 0$. Obviously \mathbf{B}_h is singular. Both $\mathbf{I} + \Delta t\mathbf{A}_h$ and $\mathbf{I} + \Delta t\mathbf{B}_h$ are Z -matrices. Both are semi-positive

$$(\mathbf{I} + \Delta t\mathbf{A}_h)\delta \geq (\mathbf{I} + \Delta t\mathbf{B}_h)\delta \geq \delta > 0 .$$

By Lemma 3.7.4 $\mathbf{I} + \Delta t\mathbf{A}_h$ and $\mathbf{I} + \Delta t\mathbf{B}_h$ are M -matrices. We may add any positive diagonal matrix $\Delta t k \mathbf{I}$ and the M -property is preserved. Thus both system matrices are M -matrices, even with non-constant, non-negative reaction rates k^\pm . It remains to show the uniform bound for the inverse. We have that

$$\begin{aligned} [(1 + \Delta t k^+) \mathbf{I} + \Delta t \mathbf{A}_h] \delta &\geq \delta , \\ [(1 + \Delta t k^-) \mathbf{I} + \Delta t \mathbf{B}_h] \delta &\geq \delta . \end{aligned}$$

Multiplying by the non-negative inverse, taking norms the desired bound follows

$$\begin{aligned} \|(1 + \Delta t k^+) \mathbf{I} + \Delta t \mathbf{A}_h\|_\delta^{-1} &\leq 1 , \\ \|(1 + \Delta t k^-) \mathbf{I} + \Delta t \mathbf{B}_h\|_\delta^{-1} &\leq 1 \end{aligned} \tag{3.27}$$

and the proof is complete. \square

Note that $\|\cdot\|_\delta$ is the well-known max–norm. Via the eigensystem of \mathbf{A}_h one can also show that \mathbf{A}_h itself is a M–matrix.

The uniform bound of the inverse system matrix implies stability of the numerical method: By linearity, errors are governed by the system (3.26) itself. Uniform boundedness (3.27) implies stability in the sense that errors at later time depend continuously on initial errors. Convergence towards a smooth solution of the reaction–diffusion system (3.25) follows from the discrete Gronwall lemma.

Acknowledgement

The authors want to thank Niels Christian Overgaard from Lund University for introducing us to level set methods in image segmentation.

Bibliography

- [1] Aridor M., Hannan L.A.: Traffic Jams II: An Update of Diseases of Intracellular Transport. *Traffic*, 3(11), pp. 781-790. (2002)
- [2] Babuška I., Banerjee U., Osborn J.E.: Survey of meshless and generalized finite element methods: A unified approach, *Acta Numerica* 12, 1-125. (2003)
- [3] Bancaud A., Huet S., Daigle N., Mozziconacci J., Beaudouin J., Ellenberg J.: Molecular crowding affects diffusion and binding of nuclear proteins in heterochromatin and reveals the fractal organization of chromatin. *EMBO J.* 28, 3785-3798. (2009)
- [4] Beaudouin J., Mora-Bermúdez F., Klee T., Daigle N., Ellenberg J.: Dissecting the contribution of diffusion and interactions to the mobility of nuclear proteins. *Biophys. J.* 90, 1878-1894. (2006)
- [5] Bohl E.: Finite Modelle gewöhnlicher Randwertaufgaben, *Teubner, Stuttgart.* (1981)
- [6] Bohl E.: Monotonie: Lösbarkeit und Numerik bei Operatorgleichungen, *Springer Tracts in Natural Philosophy* 25, *Berlin.* (1974)
- [7] Braga J., Desterro J.M., Carmo-Fonseca M.: Intracellular macromolecular mobility measured by fluorescence recovery after photobleaching with confocal laser scanning microscopes, *Mol. Biol. Cell* 15: 4749-4760. (2004)
- [8] Chan T. and Vese L.: Active Contours Without Edges, *IEEE transactions on image processing* vol. 10, no. 2, pp. 266-277. (2001)
- [9] Collatz L.: Funktionalanalysis und Numerische Mathematik, *Springer, Berlin.* (1964)
- [10] Davis, T.A.: Algorithm 832. *ACM Transactions on Mathematical Software* 30 (2): pp. 196–199. (2004)
- [11] Geuzaine Ch. and Remacle J.F.: Gmsh, <http://geuz.org/gmsh>

- [12] Ghosh S.K., Cherstvy A.G., Metzler R.: Non-universal tracer diffusion in crowded media of non-inert obstacles. *Phys. Chem. Chem. Phys.* **17**, 1847-1858. (2015)
- [13] Hairer E. and Wanner G.: Solving Ordinary Differential Equations II, *Springer, Heidelberg.* (1996)
- [14] Hansen Ch.V.: Segmentation of Fluorescent Microscopy Images of Living Cells, Bachelor Project, IMADA, SDU (2012)
- [15] Hinde E., Cardarelli F., Digman M.A., Gratton E.: In vivo pair correlation analysis of EGFP intranuclear diffusion reveals DNA-dependent molecular flow. *Proc. Natl. Acad. Sci. U. S. A.* **107**, 16560-16565. (2010)
- [16] Istratov A.A. and Vyvenko O.F.: Exponential analysis in physical phenomena. *Rev. Sci. Instr.* **70**, 1233-1257. (1999)
- [17] Jönsson P., Jonsson M.P., Tegenfeldt J.O., Höök F.: A method improving the accuracy of fluorescence recovery after photobleaching analysis. *Biophys. J.* **95**, 5334-5348. (2008)
- [18] Kass M., Witkin A., Terzopoulos D.: Snakes: Active Contour Models, *Int Journal of Computer Vision*, pp.321-331. (1987)
- [19] Lippincott-Schwartz J., Altan-Bonnet N., Patterson G.H.: Photobleaching and photoactivation: following protein dynamics in living cells., *Nat. Cell Biol., Suppl.* **7** (2003), S7-14. (2003)
- [20] Logg A., Mardal K.-A., Wells G.N. et al.: Automated Solution of Differential Equations by the Finite Element Method, *Springer, Heidelberg.* (2012)
- [21] Lund F.W., Wüstner D.: A comparison of single particle tracking and temporal image correlation spectroscopy for quantitative analysis of endosome motility, *Journal of Microscopy*, Vol. 252, pp. 169-188. (2013)
- [22] McWhirter, J.G., Pike, E.R.: On the numerical inversion of the Laplace transform and similar Fredholm integral equations of the first kind. *J. Phys. A: Math. Gen.* **11**, 1729-1745. (1978)
- [23] Mueller F., D. Mazza D., Stasevich T.J., McNally J.G.: FRAP and kinetic modeling in the analysis of nuclear protein dynamics: what do we really know? *Curr. Opin. Cell Biol.* **22**, 403-411. (2010)
- [24] Portable, Extensible Toolkit for Scientific Computation:
<http://www.mcs.anl.gov/petsc>

- [25] Sprague B.L., Pego R.L., Stavreva D.A., and McNally J.G.: Analysis of binding reactions by fluorescence recovery after photobleaching. *Biophys. J.* **86**, 3473-3495. (2004)
- [26] Stasevich T.J., Müller F., Brown D.T., McNally, J.G.: Dissecting the binding mechanism of the linker histone in live cells: an integrated FRAP analysis. *EMBO J.* **29**, 1225-1234. (2010)
- [27] Varga R.S.: Matrix iterative analysis, *Prentice-Hall, Englewood Cliffs, N.J.* (1962)
- [28] Vo G.D.: Comparison of continuous and discontinuous Galerkin finite element methods for parabolic partial differential equations with implicit time stepping. *M.S. thesis, Montana State University* (2012)
- [29] Wüstner D., Solanko L.M., Lund F.W., Sage D., Schroll H.J., Lomholt M.A.: Quantitative Fluorescence Loss in Photobleaching for Analysis of Protein Transport and Aggregation. *BMC Bioinformatics*, **13**:296. (2012)
- [30] Zilman A., Talia S.D., Chait B.T., Rout M.P., Magnasco M.O.: Efficiency, selectivity, and robustness of nucleocytoplasmic transport. *PLoS Comp. Biol.* **3**(7), e125. (2007)

A Discontinuous Galerkin Model for Fluorescence Loss in Photobleaching

Christian V. Hansen, Hans J. Schroll, Daniel Wüstner

Abstract

Fluorescence loss in photobleaching (FLIP) is a modern microscopy method for visualization of transport processes in living cells. This paper presents the simulation of FLIP sequences based on a calibrated reaction–diffusion system defined on segmented cell images. By the use of a discontinuous Galerkin method, the computational complexity is drastically reduced compared to continuous Galerkin methods. Using this approach on green fluorescent protein (GFP), we can determine its intracellular diffusion constant, the strength of localized hindrance to diffusion as well as the permeability of the nuclear membrane for GFP passage, directly from the FLIP image series. Thus, we present for the first time, to our knowledge, a quantitative computational FLIP method for inferring several molecular transport parameters in parallel from FLIP image data acquired at commercial microscope systems.

4.1 Introduction

Analysis of protein mobilities within living cells heavily relies on quantitative fluorescence microscopy. The protein of interest is either tagged with a green fluorescent protein (GFP) or its color variants. Alternatively, linkage tags are introduced genetically (as HaLo or SNAP tags) for subsequent labeling with suitable organic dyes [30, 34, 36]. The intracellular dynamics of such tagged proteins can be followed and quantified by three principal approaches a) measurement of fluorescence fluctuations in the steady state, as in fluorescence correlation spectroscopy and its imaging variants [17, 32], b) single molecule tracking (SMT) to gather an ensemble of trajectories of individual molecules [29, 53] and c) local disturbance of the steady state by photobleaching followed by measurement of establishing a new steady state [34, 41]. Here, we are concerned with the last approach only. The disturbance by localized photobleaching can be singular in time, as in fluorescence recovery after photobleaching (FRAP), continuous, as in continuous photobleaching (CP) or repeatedly pulsed, as in fluorescence loss in photobleaching (FLIP). In FRAP and CP, the fluorescence dynamics is typically only monitored at the site of bleaching [41, 59]. Accordingly, only one temporal profile of fluorescence change is gathered in conventional FRAP and CP and can be used for subsequent modeling of binding and diffusion processes. This comes at the risk of parameter uncertainty and overfitting [41], which is why more recent FRAP studies include the whole spatiotemporal profile involved in the bleach and recovery [6, 20, 28, 42, 55]. In FLIP, the whole cell is automatically monitored, i.e., inside and outside the bleached domain, thereby naturally providing a temporal fluorescence profile (i.e., fluorescence loss) at each pixel position. Thus, FLIP provides comprehensive quantitative data on fluorescence dynamics for the whole cell as a precondition for reliable data modeling. However, only a few attempts have been made so far, to infer transport parameters from FLIP image data [23, 37, 62]. Luedke et al. used a compartment model in their FLIP data modeling, in which a Heaviside function was used to describe the FLIP cycle of bleaching and scanning [37]. This lead to a non-linear ordinary algebraic-differential equation system, which was solved numerically. Diffusion was not explicitly included in this model. Gruebele and colleagues (2014) performed numerical simulations of the underlying reaction-diffusion model, in which the reaction term described the localized bleaching process [23]. They discretized the whole cellular domain into a few subdomains and fitted the experimental fluorescence loss in each subdomain to several diffusion models. To include the complete spatiotemporal fluorescence loss profile, we presented previously a quantitative FLIP model using a pixel-by-pixel analysis with an empirical fitting function available as a plugin to the popular image analysis program ImageJ [62, 48]. This analysis method allowed for detecting local heterogeneities in fluorescence loss kinetics, but the underlying causes

could not be inferred from the empirical model used.

In [25] we presented a reaction–diffusion compartment model for intracellular transport observed in FLIP images, which can describe both diffusion, nucleo-cytoplasmic transport, and local binding mechanistically. We focused on GFP, as many reference measurements by a variety of techniques are available, and because GFP is known to interact minimally, i.e., only non-specifically with intracellular structures [5, 13, 16, 18]. Transport of GFP across the nuclear membrane is assumed to be passive, as the GFP sequence lacks nuclear import or export signals [13, 39, 50]. We modeled that process as a passive exchange of GFP between nucleus and cytoplasm by a stiff reaction. Resolving the rate coefficients and the strong gradient in intensity across the membrane by a continuous Galerkin method required fine meshes near and inside the membrane. Also, our previous model [25] did not allow for parameter estimation, i.e., iterative refinement of the model parameter values given the data. This, however, is most wanted in using FLIP modeling as a quantitative tool for analysis of experimental image data. In contrast to [25], in the present paper, we have improved upon previous work in the following respects: 1) the jump in intensity across the membrane is treated as discontinuity. A semipermeable membrane separates the nucleus from the cytoplasm, and transport across the membrane is established by an interface condition. The spatial resolution of the membrane is avoided by a discontinuous Galerkin (DG) method working on essentially smaller meshes. The DG–mesh for a typical FLIP image consists of 1500 triangles only in comparison to 221000 triangles in the continuous Galerkin simulation presented in [25]. Even though the DG method is about 5–20 times more computationally expensive than continuous Finite Elements [58], one may expect up to 29 times faster execution for the DG simulation. 2) we have implemented a parameter estimation method, which allows us to directly determine diffusion constant, spatially varying binding constants and the nuclear membrane permeability. For that, we compare various optimization routines for iterative minimization of the error between FLIP image data and the DG model of the postulated underlying reaction–diffusion process. This is, to our knowledge, the first attempt of direct parameter inference from FLIP image data.

The outline of the paper is as follows: The next section presents a schematic description of the FLIP protocol. A reaction–diffusion PDE model is developed in Section 4.3. Incorporation of the membrane interface condition (Subsection 4.3.2) into the DG method is described in Section 4.4 while Section 4.5 comments on aspects of the implementation in the FEniCS project [35]. Calibrated simulations of FLIP sequences are presented in Section 4.6. A discussion including FLIP simulation of differently sized inert permeation probes combined with concluding remarks is provided in Section 4.7.

4.2 Fluorescence Loss in Photobleaching

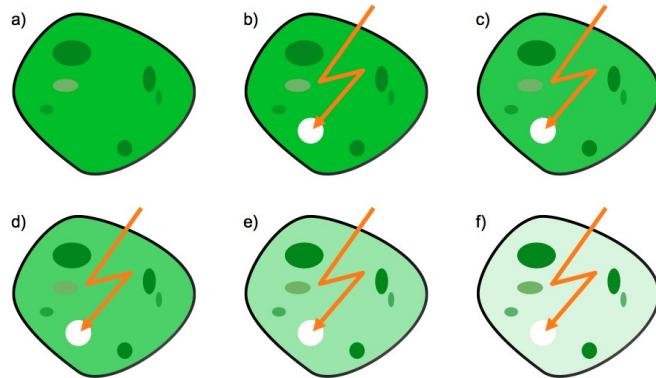


Figure 4.1: Schematic illustration of a FLIP experiment, for a cell containing green fluorescent molecules. (a) The cell at steady state. (b-f) A small region is repeatedly bleached by laser light (orange flash). Fluorescence begins to fall in the cytoplasm and in some organelles (ellipses). This demonstrates diffusive transport of molecules towards the bleached region with recruitment from some of the organelles.

In FLIP a selected cell-area is repeatedly bleached using the intense laser beam of a confocal microscope. In between the bleaches, an image scan is made to observe the transport process, see Figure 4.1 for illustration. The bleaching induces a decrease in fluorescence, not only in the bleaching area but in the whole cell due to the transport processes towards the repeatedly bleached area. This in principle allows evaluating the transport in the cell and between the intracellular compartments.

Thus, any delayed fluorescence loss in a particular cellular region outside the bleach spot indicates hindrance to molecular transport, either due to steric barriers to diffusion (for example the nuclear membrane separating cytosol from the nucleus), due to binding or because of crowding. The latter has been shown to cause excluded volume effects and, in the case of the nucleus, fractal diffusion as a consequence of the complex DNA folding and topology [5, 27, 62].

4.3 A reaction-diffusion model in segmented FLIP images

The PDE model of the FLIP process is a reduced version of the system in [25] defined on two compartments, namely nucleus and cytoplasm. To obtain a realistic simulation, the compartment boundaries are found via segmentation of the FLIP images. For this and later references, we will use Ω as a notation

for the whole cell domain, and $\partial\Omega$ denotes the boundaries. Furthermore, we let Γ_M represent the nuclear membrane, Ω_N and Ω_C represent nucleus and cytoplasm respectively. In this paper we let the bleaching area be located within the cytoplasm $\Omega_B \subset \Omega_C$, see Figure 4.2.

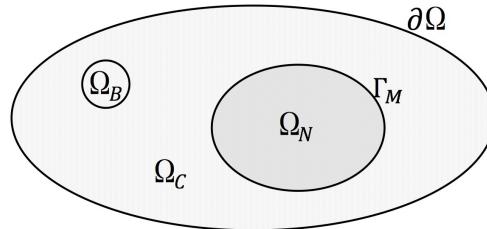


Figure 4.2: Schematic illustration of cell domains. The full cell domain is denoted Ω with boundary $\partial\Omega$. Here $\Omega = \Omega_N \cup \Omega_C$, where Ω_N and Ω_C represent the nucleus and cytoplasm, respectively. Ω_B is the bleaching domain located in the cytoplasm, such that $\Omega_B \subset \Omega_C$. Γ_M is the boundary between Ω_N and Ω_C , which represent the nuclear membrane.

The segmentation of the FLIP images is produced by the Chan-Vese active contours algorithm [12]. The algorithm is based on level set functions where the goal is to minimize the Chan-Vese energy functional by activating the level set function through an artificial time-like parameter. By minimizing the energy functional one minimizes the total deviation from the average gray-levels in for- and background, respectively. The energy functional also takes the length and thereby the smoothness of the curve into account. The implementation and further description of the Chan-Vese algorithm can be found in [24, 25]. In this paper, the algorithm is applied to localize boundaries of the cell, nucleus and bleaching area in our FLIP images. Here the cell and bleaching area are segmented from the first FLIP image, while the nucleus is segmented from frame number 45, where the nucleus geometry is clearest.

4.3.1 A reaction–diffusion model with hindrance

Inspecting the FLIP images, one of the most conspicuous things would be the architecture of especially the nucleus. There is currently put a lot of research effort on characterizing spatial heterogeneities in intracellular diffusion and transport processes. Especially within the nucleus, it is observed that molecular crowding hinder GFP’s diffusion in dense nuclear compartments [5, 27]. GFP is considered as minimally interacting protein, such that specific binding to intracellular structures can likely be ignored. However, the spatial heterogeneity of GFP distribution, which we observed especially in the nucleus, indicates that the mesoscopic cellular organization together with non-specific interactions of eGFP can cause local enrichment or depletion of this protein. Such locally varying heterogeneous distribution of eGFP can be the

consequence of protein partitioning into aqueous nuclear phases with differing properties [4]. Alternatively, it is the result of the fractal organization of diffusion barriers, for example stemming from the nuclear DNA content [5, 27]. Such barriers to diffusion have been detected in the nucleus by pair correlation analysis of intensity fluctuations of eGFP [25]. Similarly, the heterochromatin-euchromatin border has been shown to form a barrier for protein diffusion [56]. In [62] the pixel-wise FLIP analysis shows a negative correlation between DNA content and the fluorescence intensity and fluorescence loss kinetics of GFP in the nucleus. The computational FLIP model therefore needs to account for the uneven distributions of nuclear proteins.

We model the spatially varying eGFP distribution using rate constants and classical mass-action kinetics. It should be emphasized that this is a significant simplification, as diffusion of eGFP in the bounded state is ignored, and the underlying causes of local protein enrichment are not explicitly considered. However, as they are only partly understood, and we find good agreement of our simulation results with the experimental FLIP data, we use this pragmatic modeling approach here. More complicated modeling approaches including confined or anomalous diffusion will be discussed in section 4.7, below.

Thus the model consists of both hindered and free fluorescence proteins and we define the observed fluorescence intensity as:

$$c = u + u_b, \quad (4.1)$$

where u and u_b is the intensities of the free and hindered molecules, respectively. The high-intensity areas are the areas in which we find that GFP is hindered in its motion. Thus, in these areas, u has been transformed into u_b , in contrast to areas of low intensity. This is described by the reversible, first order reaction mechanism:



where k^+ and k^- are spatially resolved positive reaction constants; i.e. we account for the above mentioned diffusion barriers by a mean field approach using reaction rate constants k^+ and k^- .

Assuming diffusive transport of the free (but not the hindered) GFP-tagged molecules according to Fick's law, the time-dependent PDE model reads:

$$\begin{aligned} u_t &= \nabla \cdot (\alpha \nabla u) + k^- u_b - k^+ u - \chi_{\Omega_B} \theta b \frac{q}{1+q} u , \\ (u_b)_t &= k^+ u - k^- u_b - \chi_{\Omega_B} \theta b \frac{q}{1+q} u_b , \end{aligned} \quad \mathbf{x} \in \Omega, \quad t > 0,$$

where α is the diffusion coefficient for free GFP molecules, b is the intrinsic bleaching rate constant, q is the equilibrium constant for the reaction between

the ground and excited state for a fluorophore [38], thus $b \frac{q}{1+q}$ is the total rate at which the fluorophores are bleached inside the bleaching area Ω_B . Further, θ and χ_{Ω_B} are both characteristic functions, θ is time-dependent and simulates when the high-intensity laser bleaches, χ_{Ω_B} is space dependent and ensures that bleaching only occurs in the bleaching area:

$$\chi_{\Omega_B} = \begin{cases} 1 & \text{if } \mathbf{x} \in \Omega_B, \\ 0 & \text{else.} \end{cases}$$

At the initial time, before bleaching, the system is in equilibrium and the free molecules are uniformly distributed $u^0 = \text{const}$. Any higher fluorescence intensity is due to accumulation of hindered molecules $c^0(\mathbf{x}) = u^0 + u_b^0(\mathbf{x}) \geq u^0$. Thus, the initial intensity of free molecules is the uniform background of the observed initial intensity

$$u^0 = \min_{\mathbf{x} \in \Omega} c^0(\mathbf{x}) . \quad (4.3)$$

The equilibrium state for (4.2) is given by $u^0 k^+ = k^- u_b^0(\mathbf{x}) = k^- (c^0(\mathbf{x}) - u^0)$. It is reasonable to model $k^+ = k^+(\mathbf{x})$ to be positive where increased fluorescence intensity indicates the presence of hindered molecules

$$k^+(\mathbf{x}) = \gamma u_b^0(\mathbf{x}) = \gamma (c^0(\mathbf{x}) - u^0) , \quad (4.4)$$

and γ is a proportionality constant. Consequently, k^- is constant

$$k^- = \frac{k^+(\mathbf{x})}{u_b^0(\mathbf{x})} u^0 = \gamma u^0 . \quad (4.5)$$

4.3.2 Compartment model with semipermeable membrane

To obtain a realistic FLIP simulation at least two compartments are needed, i.e., the cytoplasm and nucleus. These compartments are separated by the nuclear membrane. According to Fick's first law the diffusive flux is anti-proportional to the gradient $\mathbf{J} = -\alpha \nabla u$. To model diffusive transport across a semipermeable membrane interface where u may jump, we integrate Fick's law across the membrane to obtain $\mathbf{J} = -p(u^- - u^+) \mathbf{n}^-$. Here, p denotes the solute permeability of the membrane measured in $\mu\text{m}/\text{s}$. The membrane separates the domain into two compartments labeled by \pm superscripts. In our cell model see Figure 4.2 for example, the nucleus Ω_N is the minus-compartment and the cytoplasm Ω_C is the plus-compartment. The outward unit normal vectors n^\pm along the common interface point into the opposite compartment. If the concentration outside is greater than inside $u^+ > u^-$, then the flux points back into the minus-compartment resulting in a damping effect in agreement with Fick's law. As the outward normals along a common interface are opposite, the flux may be written as a jump bracket

$$\mathbf{J} = p(u^- - u^+) \mathbf{n}^- = p[u] . \quad (4.6)$$

Despite the fact that biological transport across a membrane may be complex, it is common practice to approximate the permeability experimentally by dividing the measured flux by the jump in concentration [9, 21]. At this point, we are ready to summarize the mathematical model.

4.3.3 The complete PDE model

The fluorescence intensities of both free and hindered molecules are governed by the reaction diffusion system

$$\begin{aligned} u_t &= \nabla \cdot (\alpha \nabla u) + k^- u_b - k^+ u - \chi_{\Omega_B} \theta b \frac{q}{1+q} u , \\ (u_b)_t &= k^+ u - k^- u_b - \chi_{\Omega_B} \theta b \frac{q}{1+q} u_b , \quad \mathbf{x} \in \Omega , \quad t > 0 . \end{aligned} \quad (4.7)$$

The reaction rates are taken from (4.4) and (4.5). Along the membrane, the diffusive flux (4.6) is expressed as interface condition

$$\mathbf{J} \cdot \mathbf{n}^- = -\alpha \frac{\partial u^-}{\partial \mathbf{n}^-} = p[u] \cdot \mathbf{n}^- \quad \mathbf{x} \in \Gamma_M . \quad (4.8)$$

Focusing on the intracellular architecture and diffusive transport of GFP, we may assume there is no transport of GFP across the cell membrane $\partial\Omega$

$$\mathbf{n} \cdot \nabla u = \mathbf{n} \cdot \nabla u_b = 0 , \quad \mathbf{x} \in \partial\Omega . \quad (4.9)$$

The normalised initial intensity $0 \leq c(0, \mathbf{x}) \leq 1$ is extracted from the first FLIP image and

$$u(0, \mathbf{x}) = u^0 = \min_{\mathbf{x} \in \Omega} c(0, \mathbf{x}) , \quad u_b(0, \mathbf{x}) = c(0, \mathbf{x}) - u^0 , \quad \mathbf{x} \in \Omega . \quad (4.10)$$

4.4 A discontinuous Galerkin method with internal interface condition

To effectively simulate the abrupt change in fluorescence intensity as seen in FLIP images, it is desirable that the numerical method can represent discontinuous functions. The Discontinuous Galerkin (DG) method was first introduced by Reed and Hill [49] in 1973 to resolve shocks in hyperbolic conservation laws. Independently, Babuska [3], Wheeler [60] and Arnold [1] developed interior penalty discontinuous Galerkin (IPDG) methods for elliptic and parabolic problems. Since then the interest and the development of DG methods have been growing. The interested reader is referred to [14] where the history of their development until 1999 can be found.

In this paper, the interface condition along the nuclear membrane (4.8) is implemented into the IPDG method based on [1, 2].

To describe the method, we introduce some notation. Let \mathcal{T}_h denote the discretization of Ω into disjoint open elements $\mathcal{K} \in \mathcal{T}_h$. In connection, let Γ denote the union of the boundaries of all \mathcal{K} . Note that the mesh should be constructed such that $\Gamma_M \subset \Gamma$. Further we decompose Γ into three disjoint subsets $\Gamma = \partial\Omega \cup \Gamma_{\text{int}} \cup \Gamma_M$, where Γ_{int} holds all internal edges $\Gamma_{\text{int}} := \Gamma \setminus (\partial\Omega \cup \Gamma_M)$. Further, let u^+ and u^- denote a single valued function on two adjacent elements \mathcal{K}^+ and \mathcal{K}^- . As usual, \mathbf{n}^\pm denote the outward unit vectors on along $\partial\mathcal{K}^\pm$. Then average and jump term are defined as $\{u\} = (u^+ + u^-)/2$, $\llbracket u \rrbracket = u^+ \mathbf{n}^+ + u^- \mathbf{n}^-$. For piecewise defined vector valued functions \mathbf{q} let $\{\mathbf{q}\} = (\mathbf{q}^+ + \mathbf{q}^-)/2$, $\llbracket \mathbf{q} \rrbracket = \mathbf{q}^+ \cdot \mathbf{n}^+ + \mathbf{q}^- \cdot \mathbf{n}^-$. Note that the jump of a scalar gives a vector, while the jump of a vector is a scalar, moreover

$$\llbracket \mathbf{q}u \rrbracket = \llbracket \mathbf{q} \} \{ u \} + \{\mathbf{q}\} \cdot \llbracket u \rrbracket . \quad (4.11)$$

Consider the div–grad operator $\nabla \cdot (\alpha \nabla u)$ on two adjacent elements \mathcal{K}^\pm . By partial integration (Green's first identity)

$$\int_{\mathcal{K}^\pm} \nabla \cdot (\alpha \nabla u) v \, dx = - \int_{\mathcal{K}^\pm} \alpha \nabla u \cdot \nabla v \, dx + \int_{\partial\mathcal{K}^\pm} \alpha \nabla u \cdot \mathbf{n}^\pm v \, ds ,$$

where v denotes a suitable test function. Along the common edge $e = \partial\mathcal{K}^+ \cap \partial\mathcal{K}^-$, normal derivatives sum up to a jump

$$\int_e (\alpha^+ \nabla u^+ v^+ - \alpha^- \nabla u^- v^-) \mathbf{n}^+ \, ds = \int_e \llbracket \alpha \nabla u v \rrbracket \, ds .$$

Summing up over all elements $\mathcal{K} \in \mathcal{T}_h$ we thus find

$$\int_\Omega \nabla \cdot (\alpha \nabla u) v \, dx = - \int_\Omega \alpha \nabla u \cdot \nabla v \, dx + \int_{\Gamma_{\text{int}}} \llbracket \alpha \nabla u v \rrbracket \, ds - \int_{\Gamma_M} p \llbracket u \rrbracket \cdot \llbracket v \rrbracket \, ds ,$$

where both the membrane flux condition (4.6) and the zero flux boundary condition (4.9) have been used. The IPDG method enforces continuity across internal edges by a penalty term [1, 2, 3]. Using (4.11) the formula

$$\int_{\Gamma_{\text{int}}} \llbracket \mathbf{J}(u)v \rrbracket \, ds = \int_{\Gamma_{\text{int}}} \{\mathbf{J}(u)\} \cdot \llbracket v \rrbracket \, ds + \int_{\Gamma_{\text{int}}} \{\mathbf{J}(v)\} \cdot \llbracket u \rrbracket \, ds - \int_{\Gamma_{\text{int}}} \frac{\sigma}{h} \llbracket u \rrbracket \cdot \llbracket v \rrbracket \, ds$$

is symmetric and consistent for continuous solutions $\llbracket u \rrbracket = \llbracket \mathbf{J}(u) \rrbracket = 0$. Here h denotes the average diameter of two adjacent elements, and σ is the Nitsche parameter [44].

The bilinear form for the div–grad operator based on the IPDG method reads

$$\begin{aligned} D(u, v, \alpha) := & \int_\Omega \alpha \nabla u \cdot \nabla v \, dx - \int_{\Gamma_{\text{int}}} \{\alpha \nabla v\} \cdot \llbracket u \rrbracket \, ds \\ & - \int_{\Gamma_{\text{int}}} \{\alpha \nabla u\} \cdot \llbracket v \rrbracket \, ds + \int_{\Gamma_{\text{int}}} \frac{\sigma}{h} \llbracket u \rrbracket \cdot \llbracket v \rrbracket \, ds . \end{aligned} \quad (4.12)$$

The last integral in (4.12) is the internal penalty; a large enough Nitsche parameter enforces continuity across internal edges [1, 2, 3, 44]. Let v and w be discontinuous, piecewise bilinear test-functions for u and u_b respectively. The semi-discrete PDE with boundary condition and interface conditions (4.9), (4.8) reads

$$\begin{aligned} \int_{\Omega} u_t v \, dx + D(u, v, \alpha) &= R(u, u_b, v) - B(u, v) - p \int_{\Gamma_M} [\![u]\!] \cdot [\![v]\!] \, ds , \\ \int_{\Omega} (u_b)_t w \, dx &= -R(u, u_b, w) - B(u_b, w) , \end{aligned} \quad (4.13)$$

with $R(u, u_b, v) := \int_{\Omega} (k^- u_b - k^+ u) v \, dx$ and $B(u, v) := \int_{\Omega_B} \theta b \frac{q}{1+q} u v \, dx$.

We discretize the time derivative by a backward Euler step. Any higher order but L-stable method, like certain SDIRK schemes [26], are appropriate as well.

4.5 FEniCS implementation

Applying a backward Euler time step to (4.13) results in the weak form for the time step

$$\begin{aligned} \int_{\Omega} \frac{u^{n+1} - u^n}{\Delta t} v \, dx + D(u^{n+1}, v, \alpha) &= R(u^{n+1}, u_b^{n+1}, v) - B(u^{n+1}, v) \\ &\quad - p \int_{\Gamma_M} [\![u^{n+1}]\!] \cdot [\![v]\!] \, ds , \\ \int_{\Omega} \frac{u_b^{n+1} - u_b^n}{\Delta t} w \, dx &= -R(u^{n+1}, u_b^{n+1}, w) - B(u_b^{n+1}, w) . \end{aligned}$$

This weak form is conveniently implemented using the automated Finite Element package FEniCS [35]. For faster execution, it is recommended to pre-assemble the system matrix, which FEniCS can do automatically based on the given mesh and weak formulation. The pulsating laser is realized by pre-assembling two systems, with and without the bleaching term $B(u, v)$. To resolve the effect of bleaching, the bleaching interval is a multiple of the time step: $\Delta t_b = m \Delta t$, $m \in \mathbb{N}$. To use FEniCS a high-level Python script is written, where the weak formulation is expressed in the UFL form language. UFL is a domain specific language for defining weak formulations in a notation close to the one presented in this paper [57]. DOLFIN then interprets the script and passes the UFL to the Variational Form Compiler (FFC). Then Instant (build on top of SWIG) turns it into a C++ function callable from Python. In the end, the linear systems are solved by the UMFPACK sparse, direct solver via PETSc [15, 47]. Optional iterative and parallel solvers are available. A test on the given mesh and the system from this paper showed that the iterative generalized minimal residual method with PETSc algebraic multigrid preconditioner was overall 20 – 30% slower than the direct solver.

4.6 Calibration and simulation of FLIP images

The discontinuous Galerkin method approximates the solution to the PDE model (4.7)–(4.10) as a piecewise bilinear and possibly discontinuous function defined on a triangulation of the cell. The discrete geometry from the segmented FLIP images is written into a `.geo` geometry file. By Gmsh [22] the mesh is constructed on the segmented cell geometry found in the `.geo` file and displayed in Figure 4.3. It consists of 1523 triangles; 991 located in the cytoplasm, 503 in the nucleus and 29 in the bleaching area.

The initial fluorescence intensity $0 \leq c(0, \mathbf{x}) \leq 1$ is extracted from the first FLIP image. The original images as displayed in the top row in Figure 4.6 (a-d) are affected by some noise, however. Therefore, the FLIP images are pre-conditioned by Gaussian smoothing (with a radius of one pixel = 0.05467326 μm) within the cell domain. The blurred images are seen in Figure 4.6 (e-h).

The intensity of free and hindered molecules is initialized according to (4.10) i.e., the intensity pattern as seen in the first blurred FLIP image is carried by the hindered molecules.

The bleaching time interval was $\Delta t_b = 0.8$ s followed by a recovery phase of 1.8 s, resulting in a total frame rate of 2.6 s. For the simulation the discrete time step is set to be $\Delta t = 0.2$ s.

Not yet defined model parameters are: the diffusion coefficient α , the bleaching term $\beta = b \frac{q}{1+q}$ both appearing in the PDE model (4.7), the proportionality factor γ in reaction rates (4.4) and (4.5), as well as the permeability constant p in the interface condition (4.8).

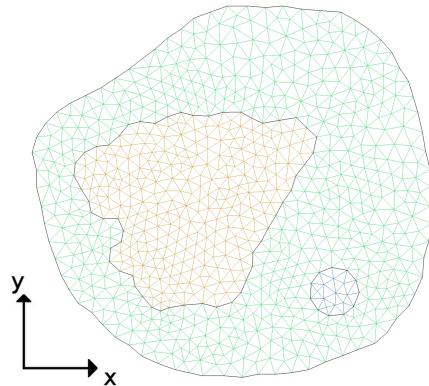


Figure 4.3: Finite element mesh on Chan-Vese active contours generated by Gmsh. Green: cytoplasm, orange: nucleus, blue: bleaching area.

4.6.1 Calibration

The remaining parameters are identified by calibrating the simulation to observed FLIP images. To this end, a misfit functional is minimized with respect to the parameters. At discrete times $t_i = 2.6(i - 1) + 2.0$ seconds $i = 1, 2, 3, \dots, 50$ we measure the difference between the simulated intensity and the preconditioned (blurred) FLIP images represented as a piecewise linear finite element function on the mesh. For tests regarding the number of FLIP images used, see Supplementary S.3. Thus, the misfit functional is expressed as

$$E = \frac{1}{50} \sum_{i=1}^{50} \int_{\Omega} |u(t_i, \mathbf{x}) + u_b(t_i, \mathbf{x}) - c_g(t_i, \mathbf{x})|^2 dx , \quad (4.14)$$

where c_g denotes the intensity of the goal function. Squaring the deviation puts a strong penalty on outliers and results in a more even distribution of residuals. The PDE constrained calibration problem reads:

$$(\bar{\alpha}, \bar{\beta}, \bar{\gamma}, \bar{p}) = \underset{0 < \alpha, \beta, \gamma, p}{\operatorname{argmin}} E(\alpha, \beta, \gamma, p) ,$$

where u and u_b solve the PDE model (4.7). To perform the optimization, we apply the Nelder–Mead downhill simplex algorithm [43] which is part of the SciPy library [54]. It calls the semipermeable membrane FLIP model (4.13) implemented as a FEniCS function. Initially the Nelder–Mead search constructs with five initial guess vectors $\xi_k = (\alpha_k, \beta_k, \gamma_k, p_k)$ forming a four dimensional simplex. The misfit functional (4.14) is evaluated in all five vertices $E_k = E(\xi_k)$ and the vertices are renumbered in ascending order $E_1 < E_2 < \dots < E_5$. The least optimal simplex vector ξ_5 is replaced by a (hopefully) better approximation. The iteration stops if both the progress in the optimal parameters $\|\xi_1^{(n+1)} - \xi_1^{(n)}\|$ and the variation of the misfit functional $E_5 - E_1$ are small. Default tolerances are 10^{-4} . For details, we refer to [43, 52, 54]. The performance of alternative algorithms and norms is discussed in Supplementary S.4 and S.5, respectively.

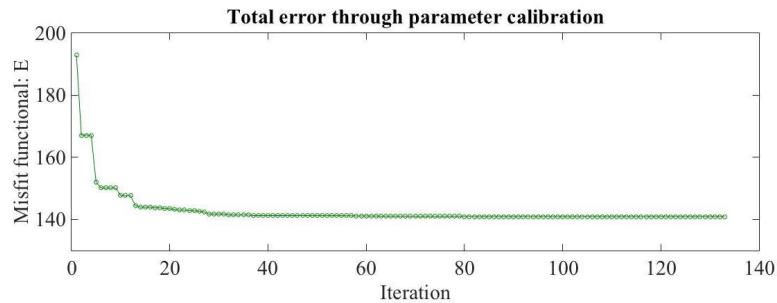


Figure 4.4: Plot of error E during the optimization process for the semipermeable membrane model.

Based on the literature [62, 46] and numerical experiments in [25] a qualified initial guess is: $\alpha_0 = 25$, $\beta_0 = 20$, $\gamma_0 = 0.5$ and $p_0 = 0.05$. Figure 4.4 depicts the progress of the optimization process with corresponding parameters shown in Figure 4.5. We clearly observe the monotone decrease of the misfit functional from initially $E = 193$ down to $E = 140$ after 133 iterations with totally 231 function evaluations. The estimated parameters are

$$\bar{\alpha} = 16.1, \quad \bar{\beta} = 35.6, \quad \bar{\gamma} = 0.319, \quad \text{and} \quad \bar{p} = 0.111. \quad (4.15)$$

The Nelder–Mead algorithm can call the FLIP solver multiple times per iteration, here resulting in 231 function evaluations in form of forward solutions of the PDE system (4.7)–(4.10). The calibration process takes approximately 3 hours on an Intel Core i5 processor at 3.2 GHz with 8 GB memory running Ubuntu 14.04.5.

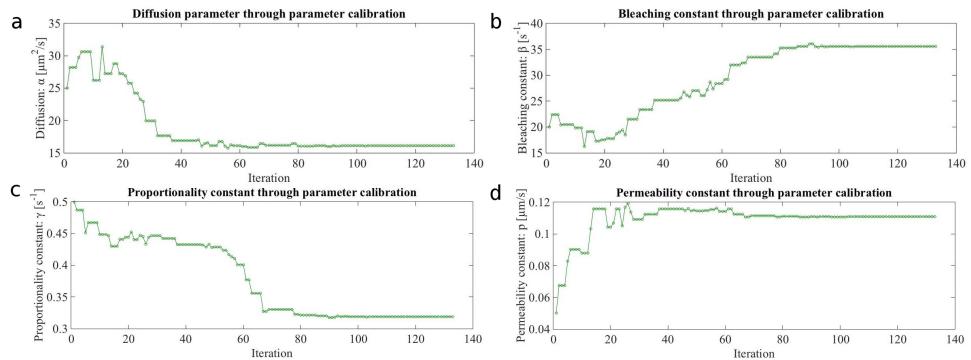


Figure 4.5: Parameters during the optimization process for the semipermeable membrane model. a) shows the evolution of the diffusion coefficient α through the calibration process, b) the evolution of the bleaching constant β , c) the proportionality constant γ , and d) the permeability constant p .

4.6.2 Simulation and visualisation

With the optimized parameters (4.15) our FLIP model as stated in Section 4.3.3 is completely determined. Recall that reaction rates k^\pm as well as initial intensities are extracted from the first (denoised) FLIP image. A sequence of FLIP images in McArdle RH7777 cells is displayed in the top row of Figure 4.6 (a-d). Green fluorescent protein (GFP) was repeatedly bleached with full laser power at a 30 pixel ($1.64 \mu\text{m}$) diameter circular region in the cytoplasm (green circle), in a temperature controlled ($35 \pm 1^\circ\text{C}$) environment of a Zeiss LSM 510 confocal microscope using the 488-nm line of an Argon laser. The entire images were scanned with 0.5% laser power between each bleach. The total frame rate inclusive bleaching was 2.6 s and the image area is approximately $15 \times 15 \mu\text{m}$. As mentioned earlier, we use Gaussian blur with

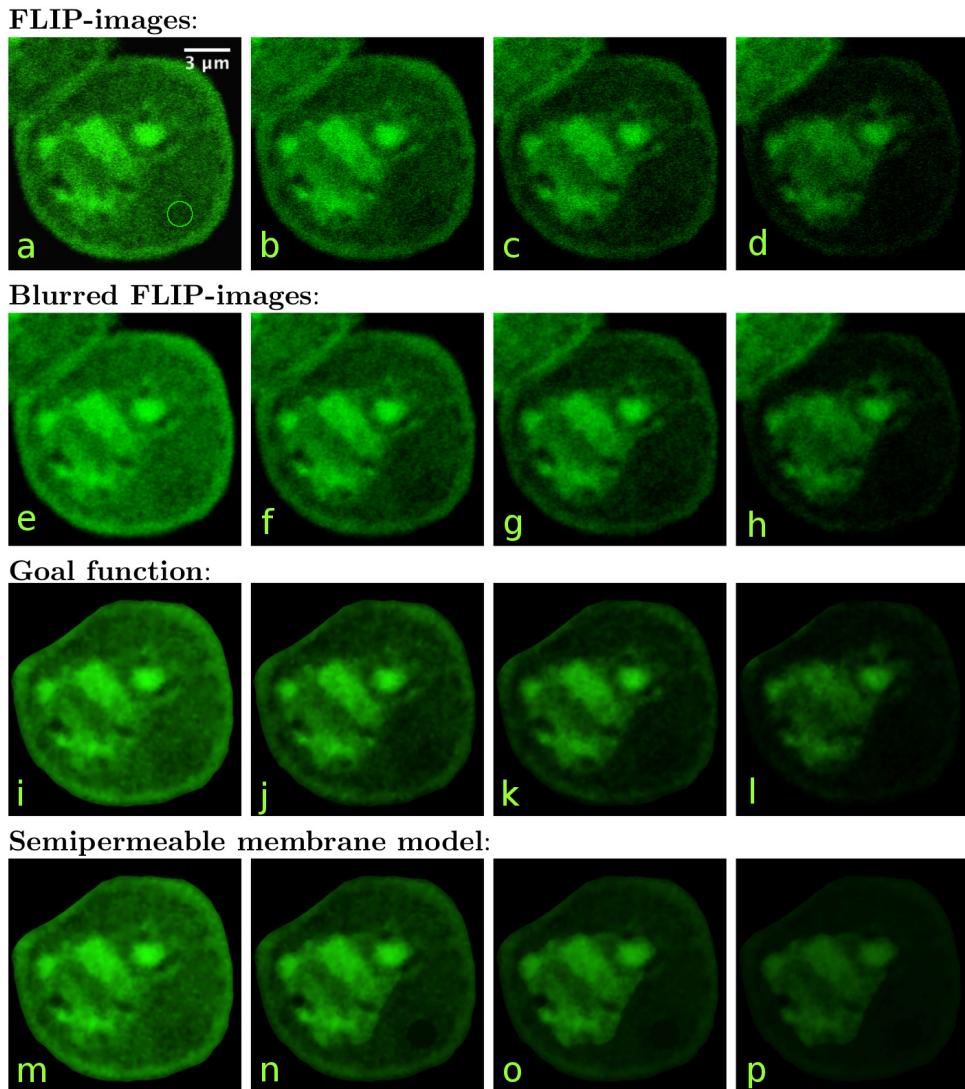


Figure 4.6: The first four images (a-d) are the original FLIP images of the McArdle RH7777 cells expressing GFP in the cytoplasm and nucleus. The green circle on the image (a) shows the 30-pixel wide bleaching area. The left most FLIP image (a) is taken before bleaching, the next image (b) is taken after it has been bleached 10 times i.e. time $t = 26$ s. The third FLIP image (c) is the 20'th FLIP image in the sequence (time $t = 52$ s) and the last (d) is at time $t = 104$ s which correspond to FLIP frame 40. The second row (e-h) shows the corresponding Gaussian blurred (radius = 1 px) FLIP images. The third row (i-l) shows the goal function and the last row (m-p) shows the simulation results, all at times corresponding to the displayed FLIP images.

radius 1 pixel to denoise the FLIP image. The blurred FLIP sequence is presented in the second row of Figure 4.6 (e-h). The first blurred FLIP image is used to create k^+ and the subsequent is used to generate goal functions. A goal function is a piecewise linear discontinuous Galerkin function defined on the mesh, based on the pixel values from the blurred FLIP images. The goal functions displayed in the third row of Figure 4.6 (i-l) were used to calibrate the FLIP model. Finally, the simulation results of our calibrated FLIP model can be seen in the lowest row of Figure 4.6 (m-p).

The structure established in the simulation mainly originates from the reaction kinetics given in (4.2). In Figure 4.7 k^+ is illustrated based on the estimated proportional factor $\bar{\gamma} = 0.319$. One can clearly see that the spatial map of k^+ resembles the structure from the first intensity image as stated in (4.4). Hindrance to free diffusion is clearly higher in the nucleus compared to the cytoplasm, which is in accordance with earlier studies [5, 6].

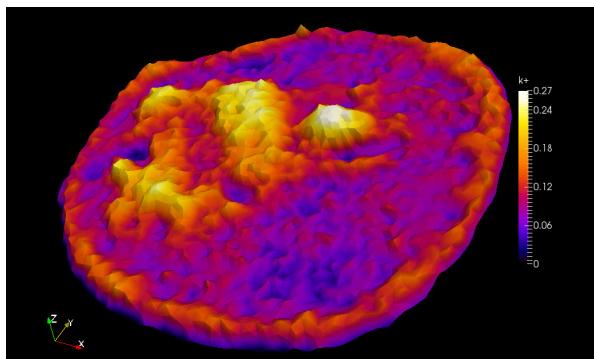


Figure 4.7: Plot of the reaction coefficient k^+ with $\gamma = 0.319$. Note that in areas where a low number of hindered molecules are observed on the FLIP images k^+ is low and in areas with high intensity k^+ is also high.

4.7 Discussion and conclusion

To compare the spatiotemporal profile of fluorescence loss between experiment and simulation, we make use of our previously developed method, namely to fit a stretched/compressed exponential (StrExp) function to each pixel position in the data and simulation outputs [62]. This function is an extension of the exponential function, as it can be considered as the sum of exponentials with a distribution of rate constants, rather than a single rate constant. This leads to a time-dependent rate coefficient, suitable for modeling delays and long-tail kinetics, not addressable using a single exponential decay function. The StrExp function is widely used for modeling physico-chemical processes and is used here to provide an independent assessment of the quality of our FLIP

model. The StrExp function provides an accurate description of fluorescence loss kinetics and reads with amplitude map $I_0(\mathbf{x})$, time constant map, $\tau(\mathbf{x})$, heterogeneity map, $h(\mathbf{x})$ and a background term, $I_b(\mathbf{x})$ [7, 31, 62, 63]

$$I(\mathbf{x}, t) = I_0(\mathbf{x}) \exp \left[- \left(\frac{t}{\tau(\mathbf{x})} \right)^{\frac{1}{h(\mathbf{x})}} \right] + I_b(\mathbf{x}) .$$

The heterogeneity parameter describes the shape of the intensity decay with $0 \leq h < 1$ modeling a delayed (compressed) exponential and $1 < h \leq 2$ modeling a stretched exponential, which is faster than exponential initially and slower for long times compared to τ . For $h = 1$, one recovers a mono-exponential function. We showed previously, that the StrExp function can accurately model diffusional transport in FLIP simulations, both in 2D and in 3D. We found that the shape of the fluorescence loss profile is well approximated with $1 < h \leq 2$ inside the bleach spot and a gradient of h -values as function of distance from the bleaching spot in the range $0.5 \leq h < 1$ outside the bleached region [62]. We demonstrated also that binding/release-dominated transport can be fitted with a StrExp function as well. Finally, we found that local heterogeneity in the h -map between neighboring pixels for GFP FLIP experiments indicates deviation from classical diffusional transport with space-invariant diffusion constant in living cells. In fact, we found for exactly the same experimental FLIP sequence used in the current study, that pixel-to-pixel variation of h -values, either larger or smaller than one exist in the cytoplasm and in the nucleus (see Figures 4 and 5 in [62]). This can be seen particularly clearly when calculating the rate coefficient map, which is defined as:

$$k(\mathbf{x}, t) = - \frac{\partial \ln I_n(\mathbf{x}, t)}{\partial t} = \frac{1}{h(\mathbf{x})\tau(\mathbf{x})} \left(\frac{t}{\tau(\mathbf{x})} \right)^{\frac{1}{h(\mathbf{x})}-1} \quad (4.16)$$

Here, $I_n(\mathbf{x}, t) = \exp(-(t/\tau(\mathbf{x}))^{(1/h(\mathbf{x}))})$ refers to the intensity decay normalized to the initial fluorescence given an amplitude equal to one [7, 62]. For a stretched decay, the rate coefficient decreases over time, while for a compressed decay, the rate coefficient increases, indicating respective slowing and accelerating fluorescence loss kinetics at a given position [62]. We fitted this function to the experimental and calibrated FLIP sequence using a plugin, which we presented previously to the popular image analysis program ImageJ [48] named PixBleach [51, 61]. As shown in Figure 4.8, the outcome of the FLIP simulation and calibration coincides nicely with the experimental FLIP data including spatially heterogeneous amplitude and time constant maps. As for the experimental data, fluorescence loss in the nucleus is significantly slowed, and the nucleus shows spatially varying fluorescence loss kinetics in experiment and FLIP simulation. From that, we conclude that our model, using spatially varying binding/release rate constants can accurately describe the

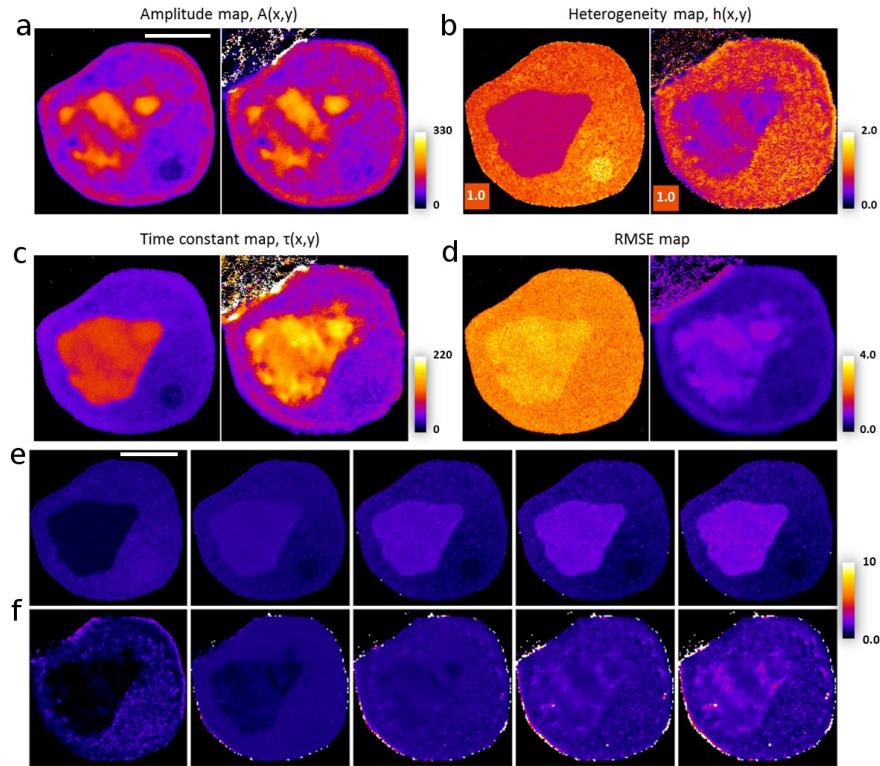


Figure 4.8: Pixel-wise comparison of temporal evolution of fluorescence loss between experiment and model (a-d); Pixel-wise fitting of a StrExp function to the calibrated model FLIP data (left panels) or to the experimental FLIP sequence (right panels) for the amplitude maps (a), the heterogeneity maps (b), the time constant maps (c) or the RMSE maps (d). The range is indicated, and the rectangular inset in panel (b) color-codes $h=1$, as a reference value for a mono-exponential decay. (e, f); selected frames of the temporal profile of rate coefficients (4.16) for the calibrated (e) and experimental FLIP data (f). The scale bars in (a) and (e) are $5 \mu\text{m}$. See text for further explanations.

experimentally known heterogeneity of nuclear diffusion of GFP, even though, we do not explicitly model spatially varying diffusion (i.e., we kept D spatially invariant and varied local binding affinities to unknown subcellular structures) [18, 62]. The spatially varying intensity of GFP is observed at steady state in living McArdle cells and has been reported in many other studies as well [18, 33]. Local differences in diffusion of GFP have been measured by fluorescence correlation spectroscopy (FCS) in the nucleus of HeLa cells, ranging from $D \approx 10 \mu\text{m}^2/\text{s}$ to $D \approx 35 - 50 \mu\text{m}^2/\text{s}$, but those differences in diffusion were not correlated with GFP intensity in the same regions [18]. It is likely that the compact nuclear DNA creates local barriers to diffusion [27], which

we detect as locally delayed fluorescence loss profiles [62].

As long as diffusion barriers are penetrable for GFP on the time scale of its cellular turnover by synthesis and degradation no spatial gradients of this protein should be expected. In other words, barriers can cause protein confinement on a short time scale but should lead to normal diffusion on a long time scale and therefore to a complete exploration of the three-dimensional nuclear space. As a consequence, any concentration gradients will be smoothed out and a homogeneous nuclear intensity of GFP would be expected. If on the other hand, the affinity of GFP for various nuclear subregions varies, a heterogeneous steady state distribution can be expected. Coexisting phases due to differences in polyelectrolyte concentration and properties have been proposed to contribute to the nuclear organization[4], and GFP could show different affinities for such nuclear domains. Thus, our simplified mass-action model, while ignoring intradomain diffusion, emphasizes exchange of GFP between nuclear areas of different affinities for this protein. The same is true, though to a lower extent, for the cytoplasm. Similarly, the nuclear membrane can be seen as a barrier to diffusion, detectable by a variant of FCS [10]. The time constant map inferred from fitting the StrExp function to the experimental FLIP sequence or to the calibrated FLIP model data changes abruptly at the nuclear membrane, demonstrating that our computational FLIP model can detect barriers to diffusion as well (Figure 4.8c). Also, the heterogeneity map and the maps of rate coefficients indicate delayed fluorescence loss in the nucleus for the experimental and calibrated FLIP sequence (compare Figure 4.8b and Figure 4.8e and f). This delay, characterized by a compressed StrExp function with increasing rate coefficients as function of time is a direct consequence of the presence of two effects: i) the nuclear membrane, acting as stringent barrier to diffusion and ii) hindrance to diffusion combined with partitioning preference of GFP in domains in the nucleus, which also causes the higher overall accumulation of GFP in that compartment compared to the cytoplasm. Both, the comparable shape of the fluorescence loss kinetics and the nuclear accumulation of GFP despite passive permeation across the nuclear membrane, are important validations of our reaction-diffusion FLIP model. Interestingly, on a smaller spatial scale (i.e., in the range of a few microns) the heterogeneity map is more structured for the experimental FLIP data than for the calibrated model (Figure 4.8b). This leads to a larger spatial variation of the bleaching rate coefficients in the experimental FLIP sequence compared to the FLIP model (compare Figure 4.8e and f, especially in the nucleus). It is likely that this minor discrepancy is a result of anomalous diffusion processes, which are not taken into account in our model [11].

For further validating our model of passive permeation across the nuclear membrane, we made use of the data by Mohr et al. [39], who compared the size dependence of nuclear permeation of various inert and spherical probe molecules[39]. The passive (i.e. not receptor mediated) influx of each studied

molecular species followed first order kinetics, and the measured influx rate constant in permeabilized HeLa cells could be used to estimate the membrane permeability as $p = k \cdot V/A$ (nuclear volume, $V = 1130 \mu\text{m}^3$ and nuclear area, $A = 540 \mu\text{m}^2$). With these values and the Stokes-Einstein relation, we have performed a forward simulation of a FLIP experiment with selected probe molecules of very different Stokes radius (Supplemental Figure S8). Clearly, increasing the Stokes radius from 0.67 nm for Fluorescein-tagged cysteine (Fl-Cys), over 1.69 nm for Ubiquitin (Ubq) to 2.85 nm for maltose-binding protein (MBP) had a dramatic effect on the fluorescence loss kinetics in the nucleus. While the nuclear membrane presented not much of a barrier for the nucleo-cytoplasmic exchange of Fl-Cys, permeation of MBP was strongly hindered. On the same time scale, lateral diffusion of all three probe molecules to the bleached area caused complete fluorescence loss in the cytoplasm (Supplemental Figure S8). Together, these simulation results are in line with the experimental findings of Görlich and colleagues[39], and shows the potential of our reaction-diffusion FLIP model to study nuclear transport and intracellular diffusion of other cargo molecules than GFP.

The simulation results of the calibrated FLIP model agree very well with the goal function and even the FLIP images in Figure 4.6. The internal structure of the cell is accurately reproduced by the remarkably simple reaction-diffusion model. It might be worth noting that the FLIP images and hence also the goal function reflect a time interval of 1.8 s what it takes the confocal microscope to scan the image during the recovery phase after bleaching. The simulated images, however, display snapshots at discrete times $t = 0, 26, 52$ and 104 seconds.

By applying a discontinuous Galerkin method, it is possible to model the nuclear membrane as an internal interface instead of resolving the internal membrane dynamics as in [25]. As a consequence not only the DG mesh consists of 147 times fewer triangles, but also the PDE model is simpler replacing the internal membrane dynamics by the interface condition (4.8). The typical runtime for the simulation a FLIP sequence is about 108 times faster than for the continuous Galerkin method. Also, this result exceeds the expectation formulated in the introduction. One reason is that the PDE model (4.7) consists of only two equations instead of four as in [25].

In the literature, one can find several papers using a semipermeable membrane model, see [9, 19, 46]. Peters [46] measures the permeability constant for a liver cell with a different size of dextrans. The article presents results for dextrans with a molecular mass of 19.5, 39.0 and 62.0 kDa. Although only three measurements are presented, it is clear that the correlation between the mass of the molecules and the respective measured permeabilities 0.705, 0.027 and $0.0036 \mu\text{m}/\text{s}$ is nonlinear. As we only have three data points, a fit would be strongly biased by the error in the data. For GFP with its estimated Stokes radius of 2.42 nm and a molecular mass of 27 kDa however, one may expect

a permeability in the lower range of the interval $(0.027, 0.705)$. The in Section 4.6.1 estimated permeability for GFP of $p = 0.111$ clearly matches with that expectation.

It is also possible to model active transmembrane dynamics in the framework of a discontinuous Galerkin method. In that case, the semipermeable membrane condition (4.8) will be replaced by an active membrane condition based on reaction kinetics. A follow-up article is in preparation.

4.8 Supplementary Information

S.1 Verification of the calibration algorithm

To verify the constrained optimization process, we perform a calibration based on synthetic data. The goal function is emulated by computing the FLIP model (7)-(10) on the mesh displayed in Figure 3 and with the parameters

$$\alpha = 20 , \quad \beta = 15 , \quad \gamma = 0.4 , \quad p = 0.1 . \quad (4.17)$$

When started at the initial guess

$$\alpha_0 = 25 , \quad \beta_0 = 20 , \quad \gamma_0 = 0.5 , \quad p_0 = 0.05 , \quad (4.18)$$

the Nelder-Mead algorithm detects the reference parameters (4.17) after 165 iteration steps with 185 evaluations of the misfit function to 5 digits precision and with a misfit functional $E = 0.00013$. Repeated calibrations from a variation of initial guesses confirm the robustness of the method.

S.2 Convergence test

To test convergence under mesh refinement, we consider the FLIP model (7)-(10) on a simplified, rectangular domain of size $10 \times 8\mu\text{m}$ consisting of two compartments and a bleaching area within the right compartment as illustrated in Figure 4.9. The model parameters are (4.17) and the initial intensity is set to

$$c_0(x) = \sin(x/2) + 2 .$$

On a fine mesh consisting of 21150 triangles and with fixed time step $\Delta t = 0.2$

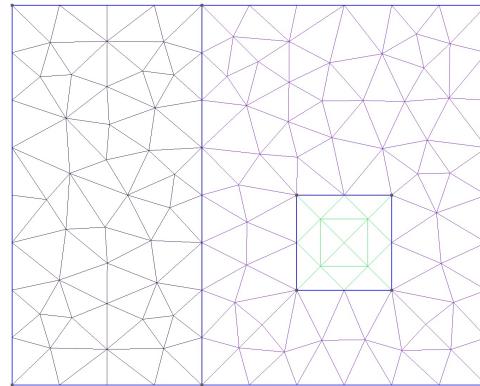


Figure 4.9: Rectangle mesh with 214 triangles. Left part emulates the nucleus, right part the cytoplasm containing the bleaching area (the square with green triangles).

h_{\max}	triangles	est. error	EOC
1.4665	214	1.197796	-
0.9485	466	0.877973	0.71
0.8482	676	0.819151	0.62
0.7137	854	0.773946	0.32
0.5747	1294	0.673180	0.64
0.4425	2516	0.526906	0.94
0.3794	3408	0.472851	0.70
0.3010	5162	0.399167	0.84
0.2262	9752	0.265909	1.29
0.1983	12822	0.237004	0.87
0.1797	14986	0.219800	0.77

Table 4.1: Convergence test results.

a reference solution is computed up to time $T = 13$.

This reference solution is used to compute the L_2 error in space at final time on a sequence of coarser meshes and to estimate the order of convergence. We observe approximately linear convergence with respect to the maximal cell diameter h_{\max} as presented in Table 4.1. A least square fit of the rates based on the five finest meshes gives an estimated order of convergence of $EOC = 1.02$. Given the fact that the approximate solution is discontinuous along the internal interface, the observed rates can be regarded as satisfying. Convergence of the method is clearly verified.

S.3 Error analysis with respect to number of FLIP images

One of the natural questions that arise when doing FLIP imaging is, how long does the sequence need to be for reliable FLIP data modeling. Especially for the simulation and calibration cases the length of the FLIP sequence is important as each iterative time step can be very time-consuming. In this section, it is therefore tested how big impact the length of the FLIP sequence has on the precision of the calibration.

The calibration test is based on synthetic data. Thus the goal function is emulated by computing the PDE FLIP model on the 2D mesh displayed in Figure 3 and with the parameters

$$\alpha = 20 , \quad \beta = 15 , \quad \gamma = 0.4 , \quad p = 0.1 .$$

When started at the initial guess

$$\alpha_0 = 25 , \quad \beta_0 = 20 , \quad \gamma_0 = 0.5 , \quad p_0 = 0.05 .$$

Figure 4.10 presents 250 calibrations within 5 different groups. Each group represents the error measured on 50 independent calibrations with 25, 50, 75,

100, 125 comparisons to the goal function per iteration, respectively. The error calculated in each of the calibrations is found with respect to the same synthetic FLIP sequence with known parameters, for each calibration, there is added random Gaussian noise to the synthetic goal sequence. An error minimization with respect to four parameters can be very time consuming, especially for the case with 125 comparisons, as each iteration then consists of 1625 Euler steps, see Section 5. Thus, to reduce the complexity, the parameters β and γ are fixed, for the calibrations in Figure 4.10.

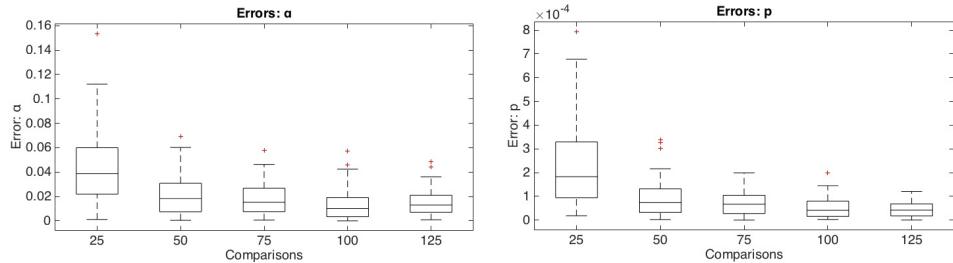


Figure 4.10: Boxplot of the errors for α and p , when β and γ is fixed, with different number of comparisons to the goal function.

Figure 4.10 shows that even though only 25 comparisons to the first 25 goal FLIP images are used, it is possible to achieve a quite accurate estimate of the parameters with respect to their size. Also, note that from 25 to 50 comparisons, the median of the error is reduced by half. However, increasing the number of comparisons to more than 50 would not give a significantly better result, especially not if compared to the increased computation time. Same experiment for calibration of β and γ , with α and p fixed is shown on Figure 4.11.

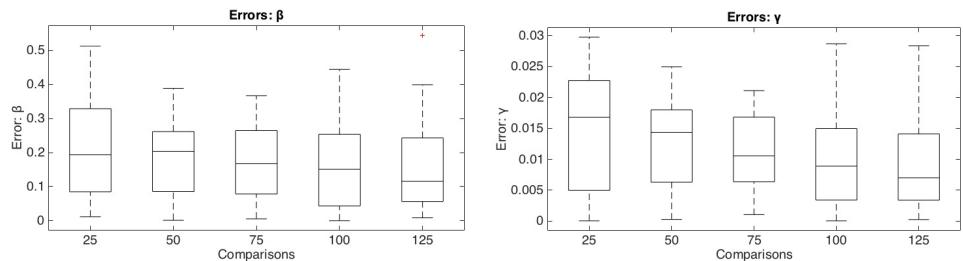


Figure 4.11: Boxplot of the errors for β and γ , when α and p is fixed, with different number of comparisons to the goal function.

The results in Figure 4.11 shows that the error might be reduced by increasing the number of comparisons, however, it doesn't show any significant improvement.

S.4 Error analysis with respect to the norm

The errors calculated in the previous section are given by the L_2 -norm of the difference between the simulation and the goal function derived from the FLIP images. In Figure 4.12 test with the L_1 , L_2 and H_1 norm are shown. As in Section S.3 each norm test consists of 50 calibrations with the synthetic FLIP sequence as goal function. The parameters β and γ are fixed, thus calibration is done with respect to α and p .

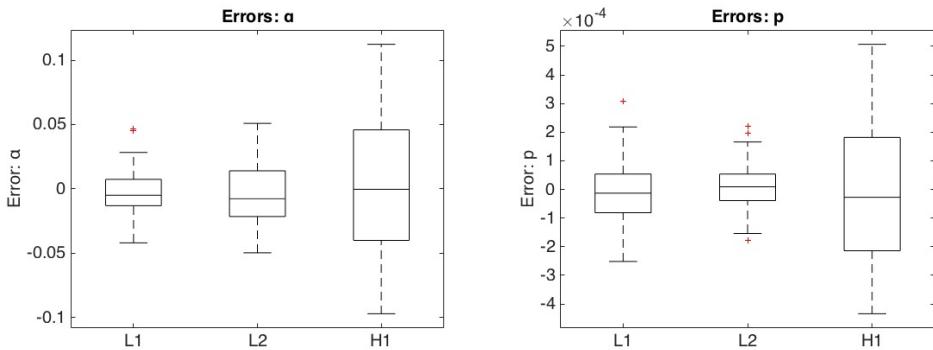


Figure 4.12: Boxplot of the errors for α and p , when β and γ is fixed, with different norms.

Figure 4.12 shows that the error obtained with the use of H_1 -norm can be significantly larger than for the L_1 and L_2 -norm. However, it is not possible to tell whether L_1 or L_2 is the better. The number of iterations for the calibration to converge is approximately the same for all the three norms. The L_2 and H_1 -norm are build into the errornorm function in FEniCS and thus easier to use.

S.5 Optimization algorithms

Calibrating parameters by minimizing the error function is done by the use of the Nelder-Mead method. However, the choice of the best optimization algorithm is not trivial and strongly depends on the problem. In this section three different optimization algorithms are tested on our problem, the Broyden–Fletcher–Goldfarb–Shanno (BFGS) algorithm (see [54]), the Limited-memory BFGS (L-BFGS) (see [8, 40, 64]) and Nelder-mead (see [43, 52]). which are all a part of the [54] library. First test shown in Figure 4.13 test the precision of the calibration of the parameters α and p , when β and γ are fixed. Each algorithm is tested 50 times, analog to the setup in Section S.3.

The different algorithms do of course have different stop criteria. For BFGS the iteration stops if a local minimizer is found and the gradient is small $\|g_k\|_\infty \leq 10^{-4}$. L-BFGS stops if either $\|g_k\|_\infty \leq 10^{-4}$ or the relative

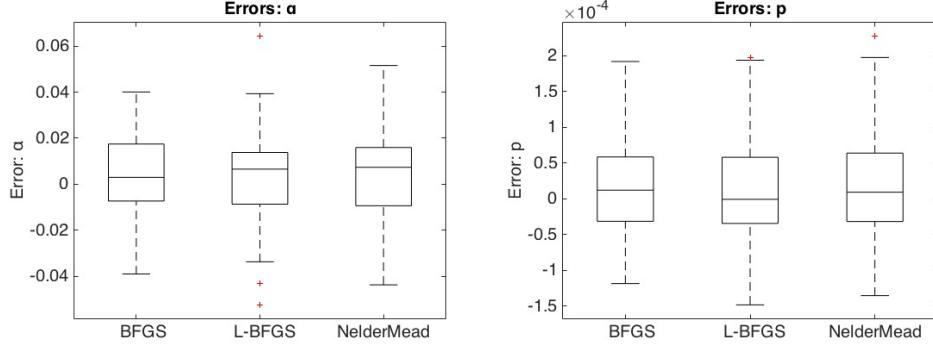


Figure 4.13: Boxplot of the errors for α and p , when β and γ is fixed, with different optimization algorithms

progress is small

$$\frac{(E^k - E^{k+1})}{\max(|E^k|, |E^{k+1}|, 1)} \leq 10^{-7} .$$

For Nelder-mead the iteration stops if either the absolute value of the largest stepsize between the found parameters is less than 10^{-3} or the absolute change in the error function between iterations is lower than 10^{-4} . These stop criteria are chosen for moderate accuracy (see [54, 64]). But despite the different stop criteria Figure 4.13 and Figure 4.14 shows that the three different optimization algorithms find the parameters with approximately same precision and same minimum for the calibration error.

Thus given the approximately same result, it is ideal to look at the computation time for each algorithm. The computation times for the calibration process can be found in Figure 4.15, which shows that the median value for the calibration process using the L-BFGS algorithm is approximately three times as slow as the fastest algorithm Nelder-Mead.

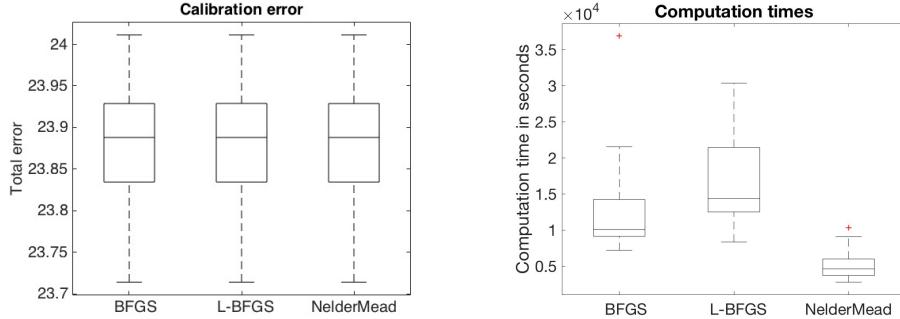


Figure 4.14: Calibration error for calibration of α and p with different algorithms.

Figure 4.15: Computation times for calibration of α and p with different algorithms.

S.6 Bleaching sensitivity

For the FLIP experiment shown in Figure 6A-D, the diameter of the bleaching area is 30 pixels. To investigate the effect and sensitivity of the bleaching coefficient β and permeability p when varying the size of the bleaching area on the calibration mesh, we setup an experiment where α and γ are fixed and set to $\alpha = 16.1$ and $\gamma = 0.319$ from the estimates presented in (15).

Diameter	$\bar{\beta}$	\bar{p}
20	106	0.111
30	35.6	0.111
40	15.3	0.111

Table 4.2: Sensitivity test with respect to the size of the bleaching area on the mesh used for calibration.

The results in Table 4.2 shows that the size of the bleaching area is inversely proportional to the bleaching parameter β , meaning that if one increases the size of the bleaching area in the mesh used for calibration, the calibrated bleaching parameter β will be lower. The change in the diameter of the bleaching area does not affect the permeability constant p . This is what one would expect, as the goal function for the calibration in all three experiments are the same, i.e., if one increased the bleaching area and did not change β , one would obtain a larger bleaching effect than the one seen in the goal function.

S.7 Molecular size

Figure 4.16 (C-N) shows simulation results for different sized molecules. All three simulations have the same initial values, k^+ , k^- and we set $\beta = 35.6$ and $\gamma = 0.319$. The first four images (C-F) are from a simulation with permeability and diffusion like Fluorescein-Cys i.e. $p = 1.97$ and $\alpha = 84.52$. The left most cFLIP image (C) is from before bleaching, the next image (D) is after it has been bleached 10 times i.e. time $t = 26$ s. The third cFLIP image (E) is the 20'th cFLIP image in the sequence (time $t = 52$ s) and the last (F) is at time $t = 104$ s which correspond to cFLIP frame 40. The second row (G-J) shows a simulation with permeability and diffusion like Ubiquitin i.e. $p = 0.07450$ and $\alpha = 33.51$. The third row (K-N) shows a simulation of MBP with $p = 0.00046$ and $\alpha = 19.87$. See the Discussion section in the paper for further details.

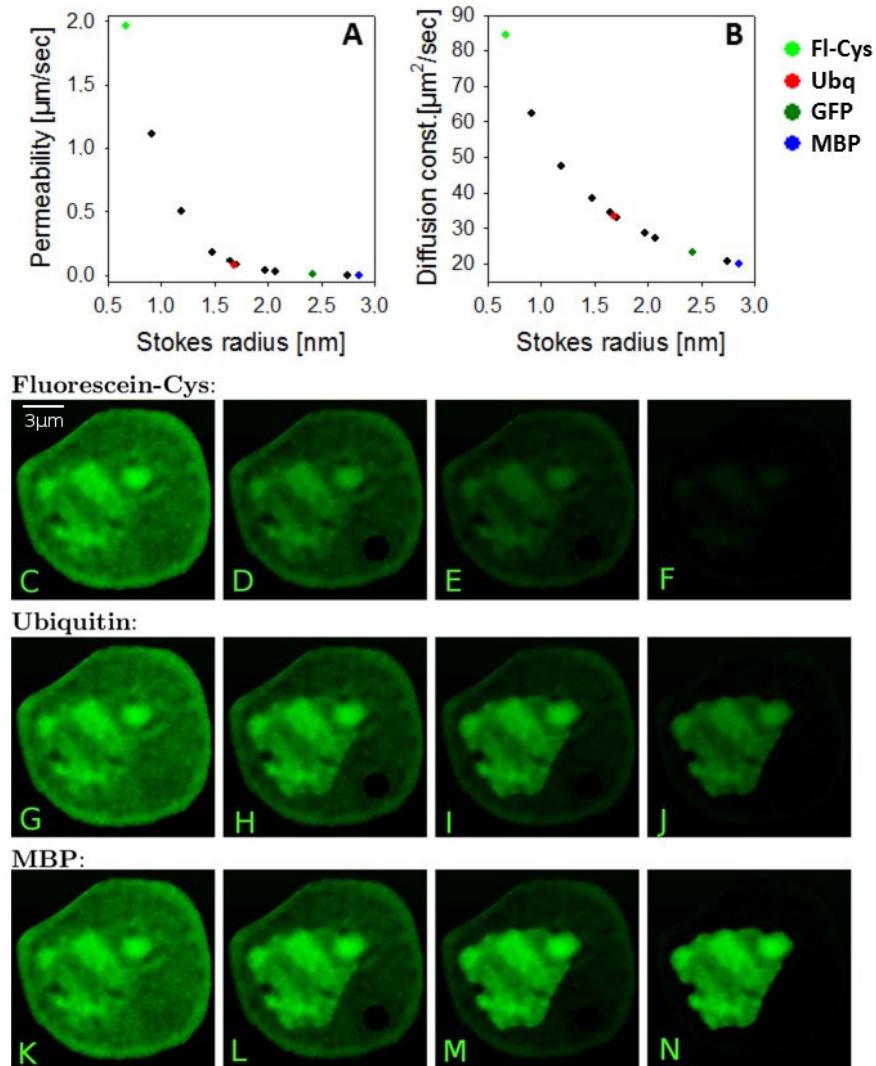


Figure 4.16: A, B. Simulation of permeability (A) and 2D-diffusion of various inert probe molecules across the nuclear membrane according to the data of [39]. Probe molecules selected for FLIP simulation are fluorescein-tagged cysteine (Fl-Cys, light green symbols), ubiquitin (Ubq, red symbols), GFP (dark green symbols) and maltose binding protein (MBP, blue symbols). C-N, selected frames from a forward FLIP simulation using the permeability values and diffusion constants shown in A, B for Fl-Cys (C-F), Ubq (G-J) and MBP (K-N).

Bibliography

- [1] Arnold D. N.: An Interior Penalty Finite Element Method with Discontinuous Elements. *SIAM J. Numer. Anal.*, 19, 742-760. (1982)
- [2] Arnold D. N., Brezzi F., Cockburn B. and Marini L. D.: Unified Analysis of Discontinuous Galerkin Methods for Elliptic Problems. *SIAM J. Numer. Anal.*, 39(5), 1749-1779. (2002)
- [3] Babuska I.: The Finite Element Method with Penalty. *Mathematics of Computation*, 27, 221-228. (1973)
- [4] Banani S.F., Lee H.O., Hyman A.A., Rosen M.K.: Biomolecular condensates: organizers of cellular biochemistry *Nature Reviews Molecular Cell Biology*, Vol. 18, No. 5, pp. 285-298 (2017)
- [5] Bancaud A., Huet S., Daigle N., Mozziconacci J., Beaudouin J. and Ellenberg J.: Molecular crowding affects diffusion and binding of nuclear proteins in heterochromatin and reveals the fractal organization of chromatin. *EMBO J.* 28, 3785-3798. (2009)
- [6] Beaudouin J., Mora-Bermúdez F., Klee T., Daigle N. and Ellenberg J.: Dissecting the contribution of diffusion and interactions to the mobility of nuclear proteins. *Biophys. J.* 90: 1878-1894. (2006)
- [7] Berberan-Santos M. N., Bodunov E. N. and Valeur B.: Mathematical functions for the analysis of luminescence decays with underlying distributions 1. Kohlrausch decay function (stretched exponential). *Chemical Physics* 315: 171-182. (2005)
- [8] Byrd R. H., Lu P. and Nocedal J.: A Limited Memory Algorithm for Bound Constrained Optimization. *SIAM Journal on Scientific and Statistical Computing*, 16, 5, pp. 1190-1208. (1995)
- [9] Cangiani A. and Natalini R.: A spatial model of cellular molecular trafficking including active transport along microtubules. *Journal of Theoretical Biology* 267, 614-625 (2010)

- [10] Cardarelli F. and Gratton E.: In vivo imaging of single-molecule translocation through nuclear pore complexes by pair correlation functions. *PLoS One.* 5: e10475. (2010)
- [11] Cherstvy A. G. and Metzler R.: Population splitting, trapping, and non-ergodicity in heterogeneous diffusion processes. *Physical chemistry chemical physics. PCCP* 15: 20220-20235. (2013)
- [12] Chan T. and Vese L.: Active contours without edges. *IEEE Trans. Image Process.* 10(2), 266-277 (2001)
- [13] Chen Y. and Müller J. D.: Probing nucleocytoplasmic transport by two-photon activation of PA-GFP. *Microsc. Res. Tech.* 69: 220-226. (2006)
- [14] Cockburn B., Karniadakis G. E. and Shu C.-W.: The Development of Discontinuous Galerkin Methods. *Cockburn B., Karniadakis G. E., Shu C.-W. (Eds.): Discontinuous Galerkin Methods. Theory, Computation and Applications. Lecture Notes in Computational Science and Engineering, 11. Springer Verlag, pp. 3-50.* (2000)
- [15] Davis T. A.: Algorithm 832. *ACM Trans. Math. Softw.* 30(2), 196-199 (2004)
- [16] Digman M. A., Sengupta P., Wiseman P. W., Brown C. M., Horwitz A. R. and Gratton E.: Fluctuation correlation spectroscopy with a laser-scanning microscope: exploiting the hidden time structure. *Biophys. J.* 88: L33-L36. (1995)
- [17] Digman M. A., and Gratton E.: Lessons in fluctuation correlation spectroscopy. *Annu. Rev. Phys. Chem.* 62: 645-668. (2011)
- [18] Dross N., Spriet C., Zwerger M., Muller G., Waldeck W. and Langowski J.: Mapping eGFP oligomer mobility in living cell nuclei. *PloS one* 4: e5041. (2009)
- [19] Elias J., Dimitro L., Clairambault J. and Natalini R.: Dynamics of p53 in single cells: physiologically based ODE and reaction-diffusion PDE models. *Phys Biol.* 11(4). (2014)
- [20] Erdel F. and Rippe K.: Quantifying transient binding of ISWI chromatin remodelers in living cells by pixel-wise photobleaching profile evolution analysis. *Proceedings of the National Academy of Sciences of the United States of America* 109: E3221-3230. (2012)
- [21] Friedman Morton H.: Principles and Models of Biological Transport, 2nd ed. *Springer-Verlag New York.* (2008)
- [22] Geuzaine Ch. and Remacle J.F.: Gmsh. <http://geuz.org/gmsh>

- [23] Guo M., H. Gelman, and M. Gruebele.: Coupled protein diffusion and folding in the cell. *PloS one* 9: e113040. (2014)
- [24] Hansen C.V.: Segmentation of Fluorescent Microscopy Images of Living Cells. *Bachelor Project Report, IMADA, SDU* (2012)
- [25] Hansen C.V., Schroll H.J. and Wüstner D.: Computational Modeling of Fluorescence Loss in Photobleaching. *Computing and Visualization in Science*, 17, 151-166. (2015)
- [26] Hairer E. and Wanner G.: Solving Ordinary Differential Equations II. *Springer, Heidelberg*. (1996)
- [27] Hinde E., Cardarelli F., Digman M.A. and Gratton E.: In vivo pair correlation analysis of EGFP intranuclear diffusion reveals DNA-dependent molecular flow. *Proc. Natl. Acad. Sci. U. S. A.* 107, 16560-16565. (2010)
- [28] Jonsson P., M. P. Jonsson, J. O. Tegenfeldt, and F. Hook.: A method improving the accuracy of fluorescence recovery after photobleaching analysis. *Biophysical journal* 95: 5334-5348. (2008)
- [29] Kalaidzidis Y.: Multiple objects tracking in fluorescence microscopy. *J. Math. Biol.* 58: 57-80. (2009)
- [30] Keppler A., Gendreizig S., Gronemeyer T., Pick H., Vogel H., and Johnson K.: A general method for the covalent labeling of fusion proteins with small molecules in vivo. *Nature biotechnology* 21: 86-89. (2003)
- [31] Kohlrausch R.: Theorie des elektrischen Rückstandes in der Leidner Flasche. *Pogg. Ann. Phys. Chem.* 91: 179-213. (1854)
- [32] Kolin D. L., and Wiseman P. W.: Advances in image correlation spectroscopy: measuring number densities, aggregation states, and dynamics of fluorescently labeled macromolecules in cells. *Cell Biochem. Biophys.* 49: 141-164. (2007)
- [33] Kühn T., Ihlainen T. O., Hyväläuma J., Dross N., Willman S. F., Langowski J., Vihinen-Ranta M. and Timonen J.: Protein diffusion in mammalian cell cytoplasm. *PLoS One*. 6: e22962. (2011)
- [34] Lippincott-Schwartz J., Altan-Bonnet N. and Patterson G.H.: Photo-bleaching and photoactivation: following protein dynamics in living cells., *Nat. Cell Biol., Suppl.* 7 (2003), S7-14. (2003)
- [35] Logg A., Mardal K.-A., Wells G.N. et al.: Automated Solution of Differential Equations by the Finite Element Method. *Springer, Heidelberg*. (2012)

- [36] Los G. V., Encell L. P., McDougall M. G., Hartzell D. D., Karassina N., Zimprich C., Wood M. G., Learish R., Ohana R. F., Urh M., Simpson D., Mendez J., Zimmerman K., Otto P., Vidugiris G., Zhu J., Darzins A., Klaubert D. H., Bulleit R. F., and Wood K. V.: HaloTag: a novel protein labeling technology for cell imaging and protein analysis. *ACS chemical biology* 3: 373-382. (2008)
- [37] Luedeke C., Frei S. B., Sbalzarini I., Schwarz H., Spang A., and Barral Y.: Septin- dependent compartmentalization of the endoplasmic reticulum during yeast polarized growth. *The Journal of cell biology* 169: 897-908. (2005)
- [38] Lund F.W. and Wüstner D.: A comparison of single particle tracking and temporal image correlation spectroscopy for quantitative analysis of endosome motility, *Journal of Microscopy*, Vol. 252, pp. 169-188. (2013)
- [39] Mohr D., Frey S., Fischer T., Guttler T. and Gorlich D.: Characterisation of the passive permeability barrier of nuclear pore complexes. *The EMBO journal* 28: 2541-2553. (2009)
- [40] Morales J.L. and Nocedal J.: Remark on "Algorithm 778: L-BFGS-B, FORTRAN routines for large scale bound constrained optimization". *ACM Trans. Math. Softw.*, 38, pp. 7:1-7:4 (2011)
- [41] Mueller F., D. Mazza, T. J. Stasevich, and J. G. McNally.: FRAP and kinetic modeling in the analysis of nuclear protein dynamics: what do we really know? *Curr. Opin. Cell Biol.* 22: 403-411. (2010)
- [42] Mueller F., Wach P., and McNally J. G.: Evidence for a common mode of transcription factor interaction with chromatin as revealed by improved quantitative fluorescence recovery after photobleaching. *Biophysical journal* 94: 3323-3339. (2008)
- [43] Nelder J.A. and Mead R.: A simplex method for function minimization. *The Computer Journal*, 7, 308-313. (1965)
- [44] Nitsche J.: Über ein Variationsprinzip zur Lösung von Dirichlet-Problemen bei Verwendung von Teilräumen, die keinen Randbedingungen unterworfen sind. *Abh. Math. Sem. Hamburg*, 36, 9-15. (1971)
- [45] ParaView by Sandia Corporation and Kitware Inc.
<http://www.paraview.org>
- [46] Peters R.: Nuclear Envelope Permeability Measured by Fluorescence Micropolymerolysis of Single Liver Cell Nuclei. *J. Biol. Chem.*, 258 (19), 11427-11429. (1983)

- [47] Portable, Extensible Toolkit for Scientific Computation (PETSc):
<http://www.mcs.anl.gov/petsc>
- [48] ImageJ by Rasband W. <https://imagej.nih.gov/ij/>
- [49] Reed W.H. and Hill T.R.: Triangular mesh methods for the neutron transport equation. *Technical Report LA-UR-73-479, Los Alamos Scientific Laboratory.* (1973)
- [50] Ribbeck K. and Gorlich D.: The permeability barrier of nuclear pore complexes appears to operate via hydrophobic exclusion. *The EMBO journal* 21: 2664-2671. (2002)
- [51] Sage D. and Wüstner D.: PixBleach: Pixelwise analysis of bleach rate in time-lapse images. *A plugin to ImageJ. In.* (2010)
- [52] Sauer T.: Numerical Analysis. *Pearson Education* (2006)
- [53] Saxton M. J. and Jacobson K.: Single-particle tracking: applications to membrane dynamics. *Annu. Rev. Biophys. Biomol. Struct.* 26: 373-399. (1997)
- [54] Scientific Computing Tools for Python: SciPy <http://scipy.org>
- [55] Stasevich T. J., Mueller F., Brown D. T. and McNally J. G.: Dissecting the binding mechanism of the linker histone in live cells: an integrated FRAP analysis. *The EMBO journal* 29: 1225-1234. (2010)
- [56] Strom A.R., Emelyanov A.V., Mir M., Fyodorov C.V., Darzacq X., Karpen G.H.: Phase separation drives heterochromatin domain formation. *Nature* 547, 241-245. (2017)
- [57] UFL Specification and User Manual:
<http://fenics.readthedocs.io/projects/ufl/en/latest/index.html>
- [58] Varga R.S.: Matrix iterative analysis. *Prentice-Hall, Englewood Cliffs, N.J.* (1962)
- [59] Wachsmuth M., Weidemann T., Müller G., Hoffmann-Rohrer U. W., Knoch T. A., Waldeck W. and Langowski J.: Analyzing Intracellular Binding and Diffusion with Continuous Fluorescence Photobleaching. *Biophys. J.* 84: 3353-3363. (2003)
- [60] Wheeler M. F.: An Elliptic Collocation-Finite Element Method with Interior Penalties. *SIAM J. Numer. Anal.*, 15, 152-161. (1978)
- [61] Wüstner D., Larsen A. L., Færgeman N. J., Brewer J. R. and Sage D.: Selective visualization of fluorescent sterols in *Caenorhabditis elegans* by bleach-rate based image segmentation. *Traffic* 11: 440-454. (2010)

- [62] Wüstner D., Solanko L.M., Lund F.W., Sage D., Schroll H.J. and Lomholt M.A.: Quantitative Fluorescence Loss in Photobleaching for Analysis of Protein Transport and Aggregation. *BMC Bioinformatics*, 13:296. (2012)
- [63] Wüstner D., Christensen T., Solanko L. M. and Sage D.: Photobleaching kinetics and time-integrated emission of fluorescent probes in cellular membranes. *Molecules* 19: 11096-11130. (2014)
- [64] Zhu C., Byrd R. H. and Nocedal J.: Algorithm 778: L-BFGS-B, FORTRAN routines for large scale bound constrained optimization. *ACM Trans. Math. Softw.*, 23, 4, pp. 550-560. (1997)

Author contributions statement

D.W. carried out the FLIP experiments. H.J.S., C.V.H. and D.W. devised the FLIP model, its validation and the assessment of the parameter space. H.J.S. and C.V.H. made the numerical implementation of the model, performed simulations, calibrations and verifications of these. H.J.S., C.V.H. and D.W. wrote the paper. All authors reviewed and approved the manuscript.

Additional information

H.J.S. and C.V.H. acknowledge financial support from the dept. of Mathematics and Computer Science and D.W. acknowledges support from the Villum Center for Bioanalytical Sciences.

The authors declare no financial conflict of interest.

A discontinuous Galerkin model for fluorescence loss in photobleaching of intracellular polyglutamine protein aggregates

Christian V. Hansen, Hans J. Schroll and Daniel Wüstner

Abstract

Background: Intracellular phase separation and aggregation of proteins with extended poly-glutamine (polyQ) stretches are hallmarks of various age-associated neurodegenerative diseases. Progress in our understanding of such processes heavily relies on quantitative fluorescence imaging of suitably tagged proteins. Fluorescence loss in photobleaching (FLIP) is particularly well-suited to study the dynamics of protein aggregation in cellular models of Chorea Huntington and other polyQ diseases, as FLIP gives access to the full spatio-temporal profile of intensity changes in the cell geometry. In contrast to other methods, also dim aggregates become visible during time evolution of fluorescence loss in cellular compartments. However, methods for computational analysis of FLIP data are sparse, and transport models for estimation of transport and diffusion parameters from experimental FLIP sequences are missing.

Results: In this paper, we present a computational method for analysis of FLIP imaging experiments of intracellular polyglutamine protein aggregates also called inclusion bodies (IBs). By this method, we are able to determine the diffusion constant and nuclear membrane permeability coefficients of polyQ proteins as well as the exchange rates between aggregates and the cytoplasm. Our method is based on a reaction-diffusion multi-compartment model defined on a mesh obtained by segmentation of the cell images from the FLIP sequence. The discontinuous Galerkin (DG) method is used for numerical implementation of our model in FEniCS, which greatly reduces the computing time. The method is applied to representative experimental FLIP sequences, and consistent estimates of all transport parameters are obtained.

Conclusions: By directly estimating the transport parameters from live-cell image sequences using our new computational FLIP approach surprisingly fast exchange dynamics of mutant Huntingtin between cytoplasm and dim IBs could be revealed. This is likely relevant also for other polyQ diseases. Thus, our method allows for quantifying protein dynamics at different stages of the protein aggregation process in cellular models of neurodegeneration.

5.1 Background

Our understanding of protein transport and aggregation has been revolutionized by the development of genetically encoded fluorescent protein tags combined with technical innovations in high-resolution live cell fluorescence imaging. In particular, various advanced imaging methods have been used to study aggregation and phase partitioning of proteins in the nucleus and cytosol. Such protein segregation and aggregation is a hallmark of various age-associated neurodegenerative diseases, such as Alzheimer’s disease, Chorea Huntington, Ataxia or Parkinson disease. In several inherited neurodegenerative diseases, like ataxia and Huntington disease, certain proteins bearing a CAG triplet expansion coding for an extended poly-glutamine (polyQ) stretch causes the affected proteins to show the tendency to self-associate and form small and large aggregates, the latter also called inclusion bodies (IBs). Formation of IBs has been associated with disease progression, but it remains unclear, whether such large aggregates are cytoprotective or cytotoxic [17, 22, 39]. In Huntington disease, the polyQ protein is mutated huntingtin (mtHtt) containing more than 30 glutamine repeats typically, while in ataxia, one finds one out of various ataxin proteins mutated containing a polyQ stretch.

The aggregation process in Huntington disease and related polyQ diseases

has been studied extensively. Typically, suitable model cells are transfected with fluorescent protein tagged derivatives of the studied polyQ protein, and the aggregation process is studied by a variety of methods including photo-bleaching techniques like fluorescence recovery after photobleaching (FRAP) and fluoprescence loss in photobleaching (FLIP) [6, 19, 27, 44], number and brightness (N & B) analysis of intensity fluctuations [31], fluorescence complementation assays with split GFP [21], Förster resonance energy transfer (FRET) [27, 6, 38], fluorescence correlation spectroscopy [38], fluorescence lifetime microscopy [6, 12], fluorescence anisotropy imaging [3], stimulated emission depletion (STED) microscopy [33] or single molecule tracking (SMT) [23, 32, 33]. Using such techniques, different aspects of the aggregation process have been revealed. In particular, it has been suggested that diffusive oligomers and small fibrillary aggregates co-exist with IBs, which accumulate after some delay as clearly discernable micron-sized structures [10, 30, 31, 33, 34]. The oligomers or protein fibrils are sometimes difficult to detect, first due to their small size compared to IBs and second due to their low brightness which makes that they are often overshadowed by the much brighter IBs [31, 32, 33]. However, also the micron-sized IBs formed of green fluorescent protein-tagged mtHtt (GFP-mtHtt) come in strongly varying brightness levels and are eventually preceded by similarly sized but much more dynamic and eventually less bright intermediate structures in the aggregation process [32, 33]. Indeed, protein aggregates detected in cellular models of polyQ diseases are dynamic entities, often recruiting other proteins and thereby sequestering enzymes and signaling proteins which strongly affect the functionality of cells [19, 21, 27, 44]. In detailed FRAP and FLIP studies, both fast- and slow exchanging components have been described for various ataxins and mtHtt with half-times for exchange of tagged protein between cytoplasm and IBs in the range of less than 10-20 sec for various ataxins [8, 37] over 1-2 min for larger IBs of mtHtt6 [6, 37]. This strongly suggests that different populations of inclusions with different physico-chemical properties coexist in affected cells. Supporting that notion, both fibrillary and globular IBs have been detected upon expression of fluorescent protein-tagged mtHtt in the same cells, and this structural heterogeneity was reflected in differing exchange dynamics [6]. An additional level of complexity comes from the complex architecture of the cytoplasm, which generates sub-compartments of varying composition not only via membrane-bound organelles but also in the form of membrane-less liquid phases into which proteins can partition differently [36]. It has been suggested that such variety of physico-chemical phases in the cyto- and nucleoplasm can be a driving force for protein segregation, and in case of mutated polyQ proteins, trigger protein aggregation [4].

Aggregates of polyQ proteins can form in both, the cytoplasm and nucleus, and some polyQ proteins, such as mtHtt or ataxins have been shown

to bear nuclear localization and export signals, suggesting active transport across the nuclear membrane [18, 41, 43, 45]. On the other hand for mtHtt, a Ran-GTPase independent transport across the nuclear membrane has been described [9]. How the nucleo-cytoplasmic transport of polyQ proteins is kinetically coupled to their intracellular diffusion and binding to IBs is not known. FLIP is in principle an ideal method to answer this question, as fluorescence loss in different cellular areas can be quantified for repeated localized bleaching far from IBs. However, most studies applying FLIP in this context do not attempt to develop a physical model underlying the observed fluorescence loss kinetics [8, 19, 27]. In a previous study, we presented the first attempt at developing a quantitative FLIP model to estimate exchange rate constants for GFP-mtHtt from FLIP image sequences [44]. We tracked individual IBs and determined exchange rate constants relative to the overall fluorescence loss kinetics based on a multi-compartment model. However, this method lacked a proper description of intracellular diffusion and nucleo-cytoplasmic exchange of GFP-mtHtt not associated with the IBs [44].

Here, we present what we believe is a new computational method to directly infer the diffusion constants and nuclear membrane permeability coefficients of polyQ proteins as well as their binding dynamics to IBs in concert with bleaching coefficients for the intended laser bleach in the FLIP experiment directly from experimental confocal FLIP images. For that, we made use of a reaction-diffusion multi-compartment model implemented into FEniCS and solved that on a meshed surface geometry directly obtained from the cell images in the FLIP sequence. We used a discontinuous Galerkin (DG) model for improved boundary description and numerical integration of the underlying partial differential equation (PDE) system after transforming that into the weak form.

5.2 Methods

5.2.1 A reaction–diffusion model on real cell geometry.

In [16] we present a reaction–diffusion model with semipermeable nuclear membrane and hindrance for spatial heterogeneity. As described in [16] there is currently put a lot of research effort on understanding the architectures and molecular crowding in living cells. Therefor the computational FLIP model also allows this by a space dependent first order reaction kinetic given by:

$$u \xrightarrow{\frac{k_{on}}{k_{off}}} u_b, \quad (5.1)$$

where u and u_b is the intensities of the free and hindered molecules, respectively.

Letting the observed fluorescence intensity from the FLIP images be described by:

$$c = u + u_b. \quad (5.2)$$

For areas with high intensity we would find a higher population of the hindered u_b proteins. Then given the first order reaction kinetic (5.1), the space dependent reaction rate k_{on} will be high in high-intensity areas and zero in the areas with lowest intensities. Thus letting c^0 be the observed intensity from the first FLIP image, u^0 be the intensity of the free molecules and u_b^0 be the intensity of the hindered molecules such that (5.2) is fulfilled. Letting γ be the proportionality constant then by [16] the reaction rates are set to:

$$k_{\text{on}}(\mathbf{x}) = \gamma u_b^0(\mathbf{x}) = \gamma(c^0(\mathbf{x}) - u^0) , \quad (5.3)$$

where γ is a proportionality constant. Consequently, k_{off} is constant

$$k_{\text{off}} = \frac{k_{\text{on}}(\mathbf{x})}{u_b^0(\mathbf{x})} u^0 = \gamma u^0 . \quad (5.4)$$

Letting eGFP diffusion be expressed in the terms of Fick's law and α being the diffusion constant for the free eGFP molecules, our time-dependent PDE model reads:

$$\begin{aligned} u_t &= \nabla \cdot (\alpha \nabla u) + k_{\text{off}} u_b - k_{\text{on}} u - \theta b \frac{q}{1+q} u \Big|_{\Omega_B} , \\ (u_b)_t &= k_{\text{on}} u - k_{\text{off}} u_b - \theta b \frac{q}{1+q} u_b \Big|_{\Omega_B} , \quad \mathbf{x} \in \Omega , \quad t > 0 , \end{aligned} \quad (5.5)$$

where θ is the time dependent indicator function simulating the high intensity laser bleaches, b is the intrinsic bleaching rate constant and q is the equilibrium constant for the reaction between the ground and excited state for a fluorophore [25]. For mass conservation the Neumann boundary condition along $\partial\Omega$ is used,

$$\mathbf{n} \cdot \nabla u = \mathbf{n} \cdot \nabla u_b = 0 , \quad \mathbf{x} \in \partial\Omega , \quad (5.6)$$

where \mathbf{n} is the outward unit normal. With initial conditions:

$$u(0, \mathbf{x}) = u_0(\mathbf{x}) , \quad u_b(0, \mathbf{x}) = (u_b)_0(\mathbf{x}) , \quad \mathbf{x} \in \Omega . \quad (5.7)$$

The cytoplasm and nucleus are separated by the nuclear membrane Γ_M with diffusive transport for eGFP through the nuclear pore complex leading to the compartment model presented in [16], where the diffusive flux is expressed as interface condition

$$\mathbf{J} \cdot \mathbf{n}^- = -\alpha \frac{\partial u^-}{\partial \mathbf{n}^-} = p[u]\mathbf{n}^- \quad \mathbf{x} \in \Gamma_M . \quad (5.8)$$

5.2.2 Multi-compartment modeling of eGFP-mtHtt exchange

As the aggregates cannot always be seen on the first FLIP image, the structure creative reaction mechanism from (5.1) do not form the aggregates. In [44] a multi-compartment model of eGFP-mtHtt exchange between cytoplasm and aggregates where presented. The multi-compartment approach is here transferred into a multi-compartment model with a transport process as an internal interface conditions, with the first order transport kinetics described as:

$$u_C \xrightleftharpoons[k_2]{k_1} u_A , \quad (5.9)$$

where u_C is the intensity in the cytoplasm and u_A is the intensity in the respective aggregate. Expressed as a differential equation the mass preserving transport process becomes:

$$\begin{aligned} (u_C)_t &= k_2 u_A - k_1 u_C , \\ (u_A)_t &= k_1 u_C - k_2 u_A . \end{aligned} \quad (5.10)$$

Applied as an interface condition, u_C and u_A becomes the intensities in each of the illustrated neighboring triangles in Figure 5.1, which are located at the cytoplasmic and aggregate side of the aggregates boundary Γ_A , respectively.

Therefore this reaction only happens between two adjacent triangles where their common edge is a part of the line that separates the cytoplasm and aggregates.

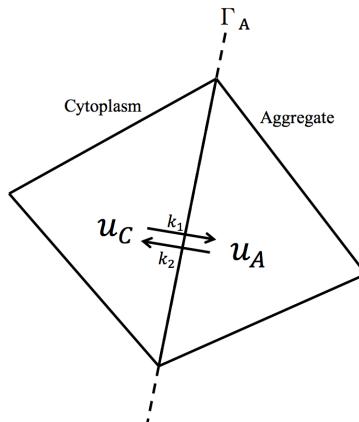


Figure 5.1: Transport kinetics between the aggregates and cytoplasm.

5.2.3 Real cell geometry

The cell geometry (see Figure 5.2) is conveyed from the FLIP images by use of an extended implementation of [14] which uses the "Active Contours Without Edges" method by Chan and Vese[7]. The Chan-Vese model does not depend on the image gradients, and is therefore able to accomplish a segmentation on more blurred images. This Chan-Vese model uses the level set function to iteratively minimize the Chan-Vese energy function that considers the length of the contour and the divergence in the pixel values inside and outside the contour, respectively. As bleaching of the FLIP images occurs in the nucleus, it is hard to segment it automatically from the FLIP sequence. Thus the geometry of the nucleus is here set by hand. However, the cell geometry is segmented from the first image and the aggregates are all segmented from the last FLIP image. The mesh is generated on the geometry in Figure 5.2 with Gmsh and then converted to XML-file.

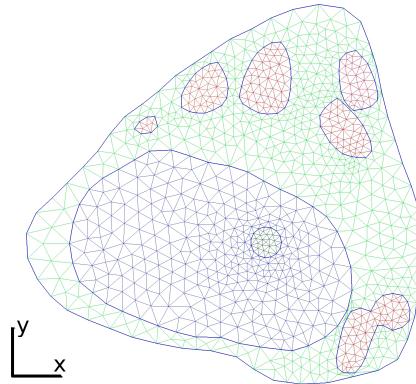


Figure 5.2: Mesh with 1825 triangles on the real cell geometry. The green triangles constitute the cytoplasm, in red is the aggregates, the dark blue triangles form the nucleus and inside nucleus the round bleaching area with a diameter of $25 \mu\text{m}$ can be found.

5.2.4 A discontinuous Galerkin method with internal interface condition

In [16], the interface condition along the nuclear membrane (5.8) was implemented into the interior penalty discontinuous Galerkin method based on [1, 2]. Additionally, in this paper, the internal interface condition along the aggregates boundaries are implemented. For the implementation, the weak form for the aggregate interface conditions is here considered.

First let the discretization of Ω be denoted by \mathcal{T}_h consisting of disjoint open elements $\mathcal{K} \in \mathcal{T}_h$. While integrating along Γ_A , u^- and u^+ are considered as

the values of two different but adjacent elements \mathcal{K}^+ and \mathcal{K}^- with a common edge on Γ_A . To rewrite (5.10) into integral form with the u^- and u^+ notation, (5.10) is split up in two cases, one if u^- is in the cytoplasm and one if u^- is in the aggregate. An indicator function I_C is therefore introduced as:

$$I_C(u) = \begin{cases} 1 & \text{if } \mathcal{K}_u \in \Omega_C \\ 0 & \text{else.} \end{cases} \quad (5.11)$$

Thus the weak form reads:

$$\int_{\Omega} u_t \, dx = M(u, v) , \quad (5.12)$$

where

$$\begin{aligned} M(u, v) := & \int_{\Gamma_A} I_C(u^-) \left((k_2 u^- - k_1 u^+) v^+ + (k_1 u^+ - k_2 u^-) v^- \right) \, dS \\ & + \int_{\Gamma_A} I_C(u^+) \left((k_1 u^- - k_2 u^+) v^+ + (k_2 u^+ - k_1 u^-) v^- \right) \, dS \end{aligned} \quad (5.13)$$

and v as the usual test function.

For notation, now let Γ denote the union of the boundaries of all the disjoint open elements \mathcal{K} . Furthermore, let Γ consist of four disjoint subsets, such that $\Gamma = \partial\Omega \cup \Gamma_{\text{int}} \cup \Gamma_M \cup \Gamma_A$. Thus Γ_{int} holds all internal edges. Then usual average and jump term for DG-methods are defined as $\{u\} = (u^+ + u^-)/2$, $\llbracket u \rrbracket = u^+ \mathbf{n}^+ + u^- \mathbf{n}^-$. For vector valued functions \mathbf{q} the average and jump term are defined as: $\{\mathbf{q}\} = (\mathbf{q}^+ + \mathbf{q}^-)/2$, $\llbracket \mathbf{q} \rrbracket = \mathbf{q}^+ \cdot \mathbf{n}^+ + \mathbf{q}^- \cdot \mathbf{n}^-$. where \mathbf{n}^\pm is the outward unit vectors on $\partial\mathcal{K}^\pm$.

Reusing the notation from [16] we let

$$\begin{aligned} D(u, v, \alpha) := & \int_{\Omega} \alpha \nabla u \cdot \nabla v \, dx - \int_{\Gamma_{\text{int}}} \{\alpha \nabla v\} \cdot \llbracket u \rrbracket \, ds \\ & - \int_{\Gamma_{\text{int}}} \{\alpha \nabla u\} \cdot \llbracket v \rrbracket \, ds + \int_{\Gamma_{\text{int}}} \frac{\sigma}{h} \llbracket u \rrbracket \cdot \llbracket v \rrbracket \, ds , \end{aligned} \quad (5.14)$$

$$R(u, u_b, v) := \int_{\Omega} (k_{\text{off}} u_b - k_{\text{on}} u) v \, dx , \quad (5.15)$$

$$B(u, v) := \int_{\Omega_B} \theta b \frac{q}{1+q} u v \, dx . \quad (5.16)$$

Thus our weak formulation reads:

$$\begin{aligned} \int_{\Omega} u_t v \, dx + D(u, v, \alpha) = & R(u, u_b, v) - B(u, v) + M(u, v) - p \int_{\Gamma_M} \llbracket u \rrbracket \cdot \llbracket v \rrbracket \, ds , \\ \int_{\Omega} (u_b)_t w \, dx = & -R(u, u_b, w) - B(u_b, w) , \end{aligned} \quad (5.17)$$

where v and w are the usual test functions.

Any L-stable method can be used for discretizing the time derivative. Here the backward Euler is used for the implementation using the automated Finite Element package FEniCS [24]. Pre-assemble the system matrix will improve the computational time in FEniCS. However, as the bleaching term is time dependent the system is here pre-assembled into two system matrices. One with and one without the bleaching term. Inside the python script, the weak formulation is therefore expressed twice in the UFL form language.

For simplicity the bleaching term $b \frac{q}{1+q}$ from (5.16) is replaced by β in the implementation and calibration. An example of the weak formulation with the bleaching term is presented here:

Python code

```
# F1 and F2 with bleaching
F1b = (1/dt)*(u-u0)*v *dx \
+ alpha*dot(grad(v), grad(u))*dx \
+ p*dot(jump(u,n), jump(v,n))*dSm \
- dot(avg(alpha*grad(v)), jump(u, n))*dSS \
- dot(jump(v, n), avg(alpha*grad(u)))*dSS \
+ sigma/h_avg*dot(jump(v, n), jump(u, n))*dSS \
+ k_on*u*v*dx - k_off*ub*v*dx \
+ beta*u*v*dxb \
- (Ic('')*((k2*u('')-k1*u(''))*v('') + \
(k1*u('')-k2*u(''))*v('')))*dSa \
- (Ic('')*((k1*u('')-k2*u(''))*v('') + \
(k2*u('')-k1*u(''))*v('')))*dSa

F2b = (1/dt)*(ub-ub0)*w *dx \
- dot(avg(grad(w)), jump(ub, n))*dSS \
- dot(jump(w, n), avg(grad(ub)))*dSS \
+ sigma/h_avg*dot(jump(w, n), jump(ub, n))*dSS \
+ k_off*ub*w*dx - k_on*u*w*dx \
+ beta*ub*w*dxb

# preassembly
Fb = F1b + F2b
ab = lhs(Fb); Lb = rhs(Fb)
Ab = assemble(ab)
```

Where dSm represent the integral along the membrane, dSa is the integral along the aggregates boundaries, dSS is the integral on the remaining edges with smooth solutions and dxb represents the bleaching area. A similar system matrix is implemented without the bleaching term and the left-hand side is pre-assembled as the matrix A with the right-hand side L . The time dependent system is solved in FEniCS by:

Python code

```
while t < t_end:
    if t%t_frame <= t_bleach:
        b = assemble(Lb, tensor=b)
```

```

    solve(Ab, c1.vector(), b)
else:
    b = assemble(L, tensor=b)
    solve(A, c1.vector(), b)

c0.assign(c1)
(u0, ub0) = c0.split(True)
t += dt

```

5.3 Results

5.3.1 Calibration and simulation of intracellular transport

To calibrate the unknown parameters $\alpha, \beta, \gamma, p, k_1, k_2$ we make a comparison between the simulation result and the FLIP images. The frame time for the FLIP experiment in Figure 5.3(A-D) where $\Delta t_{frame} = 2.8\text{s}$, within that time the bleaching area with a diameter of $25\mu\text{m}$ where bleached with 100% laser intensity for 2s. Thus the imaging process with a laser power of 0.5% took 0.8s.

To easily compare the simulation results and the FLIP sequence, the goal function seen in Figure 5.3(E-H) is created. The goal function is a piecewise linear discontinuous Galerkin function defined on the mesh, which represents the values from the denoised FLIP images. To denoise the FLIP sequence, Gaussian blur with a radius of 1 pixel is used. At the discrete times $t_i = \Delta t_{frame}(i-1) + t_{compare}$ seconds $i = 1, 2, 3, \dots, n$ the L_2 norm of the difference between the goal function and the simulation is calculated to represent the misfit functional as:

$$E = \frac{1}{n} \sum_{i=1}^n \int_{\Omega} |u(t_i, \mathbf{x}) + u_b(t_i, \mathbf{x}) - c_g(t_i, \mathbf{x})|^2 dx , \quad (5.18)$$

where c_g is the goal function. For the sequence in Figure 5.3 the number of FLIP images is $n = 40$ and the time where the simulation and FLIP data are compared is $t_{compare} = 2.6\text{s}$. To calibrate the unknown parameters, the Nelder–Mead downhill simplex algorithm [29] from the SciPy library [35] is used. The stop criterium is set such that either the difference in the parameter or the difference in the misfit functional between each iteration should be lower than 10^{-4} . Looking at the reactions rates k_1 and k_2 it is known from (5.10) that in equilibrium the equilibrium constant can be described as:

$$K = \frac{k_1}{k_2} = \frac{u_A}{u_C} . \quad (5.19)$$

Assuming that the first FLIP image before bleaching (see Figure 5.3A) is in equilibrium, K can be determined by the use of the average intensities

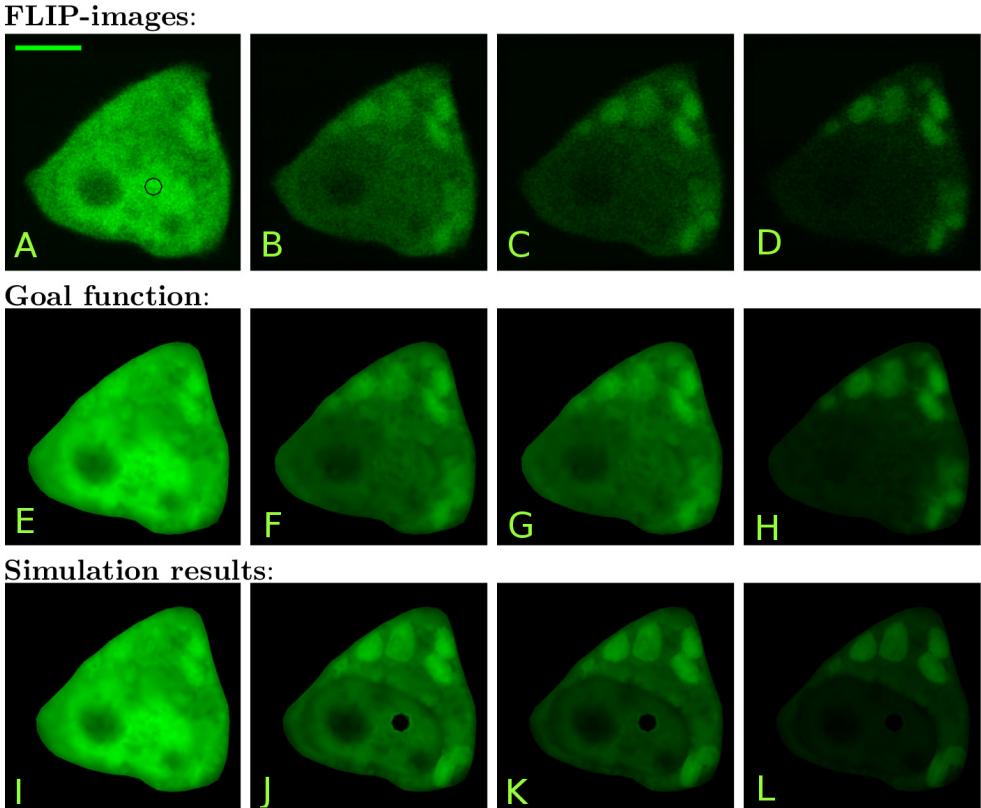


Figure 5.3: The first four images (A-D) are the original FLIP images of the CHO cells expressing GFP-Q73 in the cytoplasm and nucleus. It is produced in a temperature controlled ($35 \pm 1^\circ\text{C}$) environment on a Zeiss LSM 510 confocal microscope using the 488nm line of an Argon laser. The black circle on the image (A) shows the 25-pixel wide bleaching area and a scalebar which is 5 μm . The leftmost FLIP image (A) is taken before bleaching, the next image (B) is taken after it has been bleached 10 times, i.e., time $t = 28$ s. The third FLIP image (C) is the 20'th FLIP image in the sequence (time $t = 56$ s) and the last (D) is at time $t = 109.2$ s which correspond to FLIP frame 39. The second row (E-H) shows the corresponding goal function. The third row (I-L) shows the simulation results, all at times corresponding to the displayed FLIP images.

from inside the aggregates and cytoplasm, respectively. From the FLIP image in Figure 5.3A the equilibrium constant turns out to be $K = 1.16$. Thus by expressing k_2 in terms of k_1 , the parameters that need to be calibrated are reduced to $\alpha, \beta, \gamma, p, k_1$. The initial guesses for the calibration are set to $\alpha_0 = 25, \beta_0 = 20, \gamma_0 = 0.5, p_0 = 0.05$ and $(k_1)_0 = 0.001$. After 405 iterations and 679 evaluations, the resulting calibrated parameters are

$$\begin{aligned}\tilde{\alpha} &= 17.6, \quad \tilde{\beta} = 36.0, \quad \tilde{\gamma} = 0.198, \quad \tilde{p} = 0.318, \\ \widetilde{k_1} &= 0.0718, \quad \text{and} \quad \widetilde{k_2} = \frac{1}{1.16} \widetilde{k_1} = 0.0619.\end{aligned}\quad (5.20)$$

The misfit functional with the initial parameters $E_0 = 7,141$ were lowered to $E = 2,807$ for the calibrated parameters in (5.20). The calibration process took around 9 hours on an Intel Core i5 processor at 3.2 GHz with 8 GB memory running Ubuntu 16.04 LTS. The results of the calibration process are presented in Figure 5.3(I-L).

In Figure 5.4(A-D) a similar FLIP sequence with $\Delta t_{frame} = 2.6\text{s}$, $t_{compare} = 2.4\text{s}$ and $n = 55$ can be seen. The simulations have been made on a mesh consistent of 1998 triangles, and the initial guesses for the calibration are set to $\alpha_0 = 15$, $\beta_0 = 10$, $\gamma_0 = 0.05$, $p_0 = 0.5$ and $(k_1)_0 = 0.01$. After 179 iterations and 293 evaluations within five and a half hour the resulting calibrated parameters are

$$\begin{aligned}\tilde{\alpha} &= 15.9, \quad \tilde{\beta} = 34.6, \quad \tilde{\gamma} = 0.0614, \quad \tilde{p} = 0.447, \\ \widetilde{k_1} &= 0.0111, \quad \text{and} \quad \widetilde{k_2} = \frac{1}{1.02} \widetilde{k_1} = 0.0109.\end{aligned}\quad (5.21)$$

The simulation result with the calibrated parameters can be seen in Figure 5.4(I-L).

5.3.2 Calibration test

To test the calibration approach a forward simulation with known parameters is made to represent and replace the FLIP images, which we calibrated against. The forward simulation is made with the same initial and boundary conditions as used in Figure 5.3, on the mesh from Figure 5.2. The chosen parameters are:

$$\begin{aligned}\alpha &= 17, \quad \beta = 36, \quad \gamma = 0.2, \quad p = 0.3, \\ k_1 &= 0.0718, \quad \text{and} \quad k_2 = \frac{1}{1.16} k_1 = 0.0619.\end{aligned}\quad (5.22)$$

Gaussian noise with the mean set to zero and a variance whose size is approximately 10% of the maximum intensity is added to the results of the forward simulation. The forward simulation result now replace the goal function that is usually extracted from the experimental FLIP images in the calibration process. The rest of the setup, including the initial guesses on the parameters for the calibration, is identical to the one used for the calibration in Figure 5.3. Through the calibration process the misfit function E was lowered from 639.4

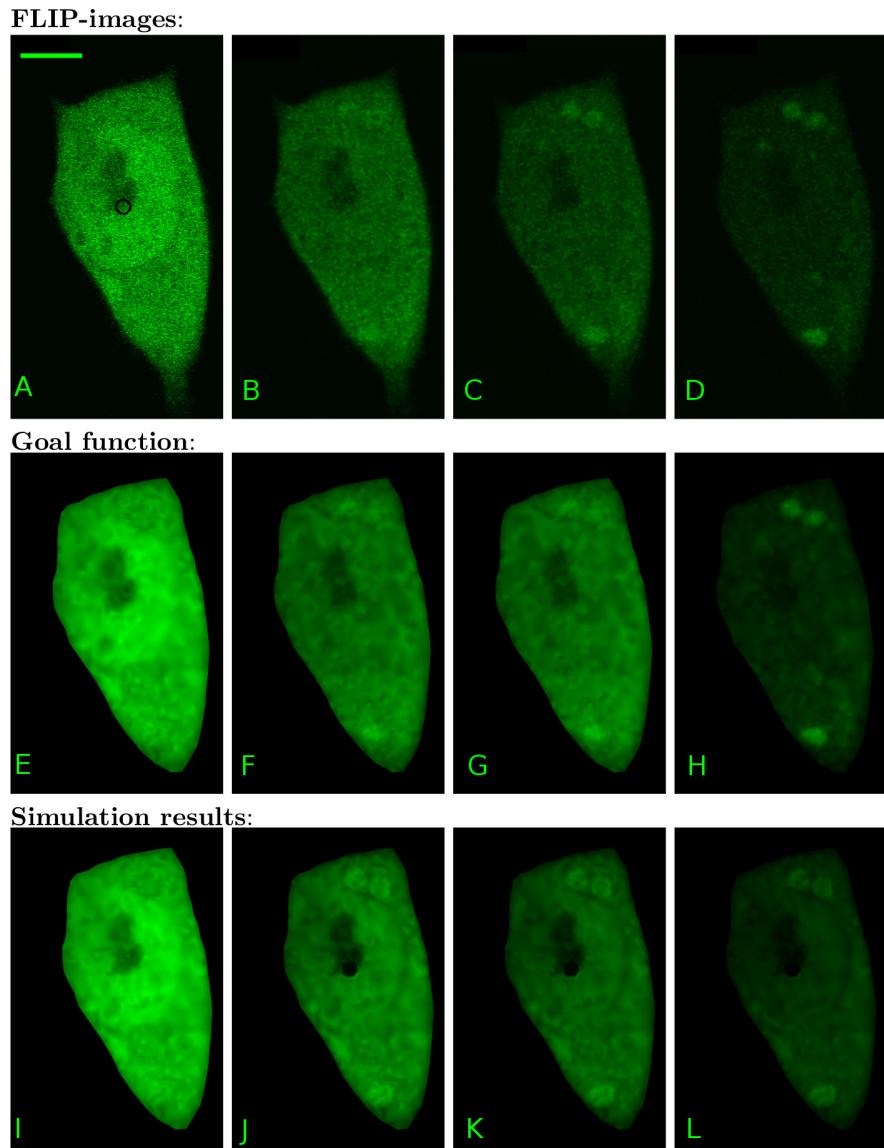


Figure 5.4: (A-D) are the original FLIP images of the CHO cells expressing GFP-Q73 in the cytoplasm and nucleus. The black circle on the image (A) shows the 18-pixel wide bleaching area and a scalebar which is $5 \mu\text{m}$. (A) is taken before bleaching, (B) is after 10 time bleaches, i.e., time $t = 26 \text{ s}$. (C) is the 20'th FLIP image in the sequence (time $t = 52 \text{ s}$) and (D) is produced at time $t = 104 \text{ s}$ which correspond to FLIP fame 40. The second row (E-H) shows the corresponding goal function. The third row (I-L) shows the simulation results, all at times corresponding to the displayed FLIP images.

to 169.7 in 388 iterations with 612 function evaluations which took around 10 hours. The calibrated parameters are:

$$\begin{aligned}\tilde{\alpha} &= 16.96 , \quad \tilde{\beta} = 35.99 , \quad \tilde{\gamma} = 0.2002 , \quad \tilde{p} = 0.3003 , \\ \tilde{k}_1 &= 0.07182 , \quad \text{and} \quad \tilde{k}_2 = \frac{1}{1.16} \tilde{k}_1 = 0.06191 .\end{aligned}\quad (5.23)$$

A small error is seen on the fourth digit, which is due to both the Gaussian noise and the size of the stop criterion for the Nelder–Mead algorithm.

5.4 Discussion

Phase separation and aggregation of polyQ proteins are prominent signs of certain neurodegenerative diseases. Often, protein inclusions of GFP–tagged polyQ proteins are first visible in cells after several days in culture allowing only for studying relatively inert, bright and stable aggregate structures [32, 33]. Thus a key requirement in traditional approaches is that the IBs and similar fluorescent protein aggregates differ in their intensity significantly from the fluorescent protein pool in the surrounding cyto- or nucleoplasm. This, however, limits the analysis to certain inclusion types. Here, we present a new computational approach for inferring diffusion, membrane permeability, and exchange rate constants of GFP–mtHtt between cytoplasm and aggregates of differing brightness directly from experimental FLIP image sequences. Our method allows for detection and dynamic characterization of protein aggregates even in cases, where they are not visible in single image acquisitions. Using the calibrated reaction–diffusion model, we found that rate constants for exchange of GFP–mtHtt between such large but dim inclusions and the cytoplasm are fast (binding rate constant $k_1 = 0.0718 \text{ s}^{-1}$ (Figure 5.3) and $k_1 = 0.0111 \text{ s}^{-1}$ (Figure 5.4) and release rate constant of $k_2 = 0.0619 \text{ s}^{-1}$ (Figure 5.3) and $k_2 = 0.0109 \text{ s}^{-1}$ (Figure 5.4)). We found similar values previously for the same protein and cell system using a simple multi–compartment model which ignored diffusion and nucleo–cytoplasmic exchange of GFP–mtHtt (i.e. binding rate constant $k_1 = 0.016 \pm 0.006 \text{ s}^{-1}$ and release rate constant of $k_2 = 0.0127 \pm 0.004 \text{ s}^{-1}$, mean \pm SEM of 6 cells) [44]. From that, we can conclude, that the typical residence time of GFP–mtHtt once bound to cytoplasmic aggregates is on order 16–83 s before being again released and available for free cytoplasmic transport and nucleo–cytoplasmic exchange. Our estimates of intracellular diffusion constants for GFP–mtHtt of $\alpha = \frac{1}{2}(15.9 + 17.6) = 16.75 \mu\text{m}^2/\text{s}$ are in good agreement with what would be expected for a protein the size of GFP–Q73 (i.e. Stokes radius of $R \approx 3.4 \text{ nm}$ [21]) in the cytoplasm (i.e. viscosity of $\eta = 3.79 \cdot 10^{-9} \frac{\text{kg}}{\text{s}\cdot\text{m}}$ predicts $\alpha = 16.6$ according to data from [28]). Supporting that notion is a previous report, which found $\alpha = 18.4 \pm 3.3 \mu\text{m}^2/\text{s}$ for diffusion of GFP–mtHtt of the same size

(i.e., Q73) in the cytoplasm of N2a cells using FRAP [21]. Using an average cytoplasmic diffusion constant of $\alpha = 16.75 \mu\text{m}^2/\text{s}$ and the upper estimate of the time constant for binding of $1/(k_1 = 0.0718 \text{ s}^{-1}) = 14 \text{ s}$ from our analysis, we conclude that GFP–mtHtt can diffuse on average $30 \mu\text{m}$ away from an aggregate after release before the next binding event takes place. Thus, diffusion is not limiting the aggregation kinetics, which explains, why we found very similar estimates for the binding and dissociation constants as reported here with our previous model which ignored cytoplasmic diffusion altogether [44]. We believe that rapid diffusion and exchange of soluble mtHtt with cytoplasmic inclusions could contribute to the efficient recruitment of other proteins to IBs which further accelerates cellular dysfunction as observed in various studies [20, 23, 27].

In [16] we presented a method using a semi-permeable membrane model to describe the transport of eGFP. The same semi-permeable membrane model is used in this paper. However, the relatively high permeabilities of the GFP–mtHtt protein may indicate that the traffic across the nuclear membrane could be caused by selective and directed transport [42]. In fact, we found that nuclear membrane permeabilities for GFP–mtHtt in the cells studied in Figure 5.3 and 5.4 were higher than what we previously observed for GFP using the same FLIP modeling approach (ref. Scientific Reports-MS). On the other hand, two to three days after transient transfection, we often observed slowed nuclear–cytoplasmic exchange of GFP–mtHtt compared GFP, likely due to the pronounced formation of sub-resolution aggregates which interfere with normal nucleo–cytoplasmic transport (not shown but see Figure 6 in [44]). Such varying results have been reported previously [5, 9, 13, 26, 40] and they could be well attributed to the eventual occurrence of soluble oligomers, whose transport across the nuclear membrane is delayed, while transport of monomeric mtHtt profits from interaction with FG-rich repeats in the nuclear pore, which can accelerate transport compared to passive cargo [28]. It is possible to replace the semi-permeable membrane model with a model similar to the compartment model for the aggregates, and thus obtain a reactive membrane transition.

The proportionality constant γ depends on the structures and intensity observed on the first FLIP image. Those these cannot be expected to be fully equal, as they are structure dependent.

In this paper, only one reaction rate is fitted for all aggregates in the same cell. Each extra reaction rate per aggregate would increase the complexity of the calibration process, such that one should have independent evidence for such heterogeneity before extending the model into that direction. For the readers that may want individual reaction mechanics for each aggregate, we suggest to calibrate the parameters α, β, γ and p first, and then fix these parameters while finding the ones for the aggregates. This can be done under

the assumption that the traffic from the aggregates is so small that it would not affect the other parameters.

5.5 Conclusion

Our new computational method allows one to determine diffusion constants, nucleo-cytoplasmic permeability and exchange kinetics of polyQ proteins, such as mtHtt, from live-cell FLIP image data. This is the first time, to our knowledge, that all such transport parameters can be inferred in parallel from the full spatiotemporal FLIP intensity profile directly within the cell geometry. Using this new method, we find that polyQ proteins can exchange rapidly between cytoplasm and aggregates and that diffusion of protein monomers is not limiting this exchange process. Furthermore, we show that computational FLIP is an efficient method to detect dim protein aggregates due to their delayed fluorescence loss. Binding and dissociation constants of mtHtt to and from such aggregates are comparable such that the inclusions are hardly visible in single images. Finally, our method sets the stage for a systematic exploration of how the aggregation process affects the nucleo-cytoplasmic permeability of polyQ proteins. Our new approach is widely applicable to quantify protein dynamics in cellular inclusions of various disease models.

Availability of data and material

The datasets used and/or analysed during the current study are available from the corresponding author on reasonable request.

Funding

DW acknowledges support from the Villum Center for Bioanalytical Sciences.

Authors' contributions

The FLIP experiments were provided by DW. The FLIP model was devised by HJS, CVH, and DW. The numerical implementation, simulations, and calibrations were performed by HJS and CVH. HJS, CVH, and DW wrote the paper. All authors reviewed and approved the manuscript.

Acknowledgements

Not applicable.

Ethics approval and consent to participate

Not applicable.

Consent for publication

Not applicable.

Competing interests

The authors declare that they have no competing interests.

Bibliography

- [1] Arnold D. N.: An Interior Penalty Finite Element Method with Discontinuous Elements. *SIAM J. Numer. Anal.*, 19, 742-760. (1982)
- [2] Arnold D. N., Brezzi F., Cockburn B. and Marini L. D.: Unified Analysis of Discontinuous Galerkin Methods for Elliptic Problems. *SIAM J. Numer. Anal.*, 39(5), 1749-1779. (2002)
- [3] Bhardwaj V., Panicker M. M., and Udgaonkar J. B.: Fluorescence anisotropy uncovers changes in protein packing with inclusion growth in a cellular model of polyglutamine aggregation. *Biochemistry* 53, 3621-3636. (2014)
- [4] Brangwynne C. P., Tompa P., and Pappu R. V.: Polymer physics of intracellular phase transitions. *Nat Phys* 11, 899-904. (2015)
- [5] Cao F., Levine J. J., Li S. H., Li X. J.: Nuclear aggregation of huntingtin is not prevented by deletion of chaperone Hsp104. *Biochimica et biophysica acta*, 1537(2), 158-166. (2001)
- [6] Caron N. S., Hung C. L., Atwal R. S., and Truant R.: Live cell imaging and biophotonic methods reveal two types of mutant huntingtin inclusions. *Human molecular genetics* 23, 2324-2338. (2014)
- [7] Chan T. and Vese L.: Active contours without edges. *IEEE Trans. Image Process.* 10(2), 266-277 (2001)
- [8] Chai Y., Shao J., Miller,V. M., Williams A., and Paulson H. L.: Live-cell imaging reveals divergent intracellular dynamics of polyglutamine disease proteins and supports a sequestration model of pathogenesis. *Proc. Natl. Acad. Sci. U. S. A.* 99, 9310-9315. (2002)
- [9] Cornett J., Cao F., Wang C. E., Ross C. A., Bates G. P., Li S. H., and Li X. J.: Polyglutamine expansion of huntingtin impairs its nuclear export. *Nature genetics* 37, 198-204. (2005)

- [10] Duim W. C., Jiang Y., Shen K., Frydman J., and Moerner W. E.: Super-resolution fluorescence of huntingtin reveals growth of globular species into short fibers and coexistence of distinct aggregates. *ACS chemical biology* 9, 2767-2778. (2014)
- [11] Geuzaine Ch. and Remacle J.F.: Gmsh. <http://geuz.org/gmsh>
- [12] Ghukasyan V., Hsu C. C., Liu C. R., Kao F. J., and Cheng T. H.: Fluorescence lifetime dynamics of enhanced green fluorescent protein in protein aggregates with expanded polyglutamine. *Journal of biomedical optics* 15, 016008. (2010)
- [13] Hackam A. S., Singaraja R, Wellington C. L., Metzler M., McCutcheon K., Zhang T., Kalchman M., Hayden M. R.: The influence of huntingtin protein size on nuclear localization and cellular toxicity. *The Journal of cell biology*, 141(5), 1097-1105. (1998)
- [14] Hansen C.V.: Segmentation of Fluorescent Microscopy Images of Living Cells. *Bachelor Project Report, IMADA, SDU* (2012)
- [15] Hansen C.V., Schroll H.J. and Wüstner D.: Computational Modeling of Fluorescence Loss in Photobleaching. *Computing and Visualization in Science*, 17, 151-166. (2015)
- [16] Hansen C.V., Schroll H.J. and Wüstner D.: A Discontinuous Galerkin Model for Fluorescence Loss in Photobleaching. *Submitted to Scientific Reports*. (2017)
- [17] Hatters D. M.: Protein Misfolding Inside Cells: The Case of Huntingtin and Huntington's Disease. *IUBMB Life.* 60, 724-728. (2008)
- [18] Irwin S., Vandelft M., Pinchev D., Howell J. L., Graczyk J., Orr H. T., and Truant R.: RNA association and nucleocytoplasmic shuttling by ataxin-1. *Journal of cell science* 118, 233-242. (2005)
- [19] Kim S., Nollen E. A., Kitagawa K., Bindokas,V. P., and Morimoto R. I.: Polyglutamine protein aggregates are dynamic. *Nat. Cell Biol.* 4, 826-831. (2002)
- [20] Kim Y. E., Hosp F., Frottin F., Ge H., Mann M., Hayer-Hartl M., Hartl F. U.: Soluble Oligomers of PolyQ-Expanded Huntingtin Target a Multiplicity of Key Cellular Factors. *Molecular cell*, 63(6), 951-964. (2016)
- [21] Lajoie P., and Snapp E. L.: Formation and toxicity of soluble polyglutamine oligomers in living cells. *PloS One.* 5, e15245. (2010)
- [22] Li S. H. and Li X. J.: Huntingtin-protein interactions and the pathogenesis of Huntington's disease. *Trends Genetics.* 20, 146-154. (2004)

- [23] Li L., Liu H., Dong P., Li D., Legant W. R., Grimm J. B., Lavis L. D., Betzig E., Tjian R., and Liu Z.: Real-time imaging of Huntington aggregates diverting target search and gene transcription. *eLife* 5. (2016)
- [24] Logg A., Mardal K.-A., Wells G.N. et al.: Automated Solution of Differential Equations by the Finite Element Method. *Springer, Heidelberg.* (2012)
- [25] Lund F.W. and Wüstner D.: A comparison of single particle tracking and temporal image correlation spectroscopy for quantitative analysis of endosome motility, *Journal of Microscopy*, Vol. 252, pp. 169-188. (2013)
- [26] Martin-Aparicio E., Avila J., Lucas J. J.: Nuclear localization of N-terminal mutant huntingtin is cell cycle dependent. *The European journal of neuroscience*, 16(2), 355-359. (2002)
- [27] Matsumoto G., Kim S., and Morimoto R. I.: Huntingtin and mutant SOD1 form aggregate structures with distinct molecular properties in human cells. *J. Biol. Chem.* 281, 4477-4485. (2006)
- [28] Mohr D., Frey S., Fischer T., Guttler T., and Gorlich D.: Characterisation of the passive permeability barrier of nuclear pore complexes. *The EMBO journal*, 28(17), 2541-2553. (2009)
- [29] Nelder J.A. and Mead R.: A simplex method for function minimization. *The Computer Journal*, 7, 308-313. (1965)
- [30] Olshina M. A., Angley L. M., Ramdzan Y. M., Tang J., Bailey M. F., Hill A. F., and Hatters D. M.: Tracking mutant huntingtin aggregation kinetics in cells reveals three major populations that include an invariant oligomer pool. *The Journal of biological chemistry* 285, 21807-21816. (2010)
- [31] Ossato G., Digman M. A., Aiken C., Lukacsovich T., Marsh J. L., and Gratton E.: A two-step path to inclusion formation of huntingtin peptides revealed by number and brightness analysis. *Biophys. J.* 98, 3078-3085. (2010)
- [32] Sahl S. J., Weiss L. E., Duim W. C., Frydman J., and Moerner W. E.: Cellular inclusion bodies of mutant huntingtin exon 1 obscure small fibrillar aggregate species. *Scientific reports* 2, 895. (2012)
- [33] Sahl S. J., Lau L., Vonk W. I., Weiss L. E., Frydman J., and Moerner W. E.: Delayed emergence of subdiffraction-sized mutant huntingtin fibrils following inclusion body formation. *Quarterly reviews of biophysics* 49, e2. (2016)

- [34] Sahoo B., Arduini I., Drombosky K. W., Kodali R., Sanders L. H., Greenamyre J. T., and Wetzel R.: Folding Landscape of Mutant Huntingtin Exon1: Diffusible Multimers, Oligomers and Fibrils, and No Detectable Monomer. *PloS one* 11, e0155747. (2016)
- [35] Scientific Computing Tools for Python: SciPy <http://scipy.org>
- [36] Shin Y., and Brangwynne C. P.: Liquid phase condensation in cell physiology and disease. *Science* 357. (2017)
- [37] Stenoien D. L., Mielke M., and Mancini M. A.: Intranuclear ataxin1 inclusions contain both fast- and slow-exchanging components. *Nature cell biology* 4, 806-810. (2002)
- [38] Takahashi Y., Okamoto Y., Popiel H. A., Fujikake N., Toda T., Kinjo M., and Nagai Y.: Detection of polyglutamine protein oligomers in cells by fluorescence correlation spectroscopy. *The Journal of biological chemistry* 282, 24039-24048. (2007)
- [39] Takahashi T., Katada S., and Onodera O.: Polyglutamine diseases: where does toxicity come from? what is toxicity? where are we going?, *Journal of molecular cell biology* 2, 180-191. (2010)
- [40] Tao T. and Tartakoff A. M.: Nuclear relocation of normal huntingtin. *Traffic*, 2(6), 385-394. (2001)
- [41] Taylor J., Grote S. K., Xia J., Vandelft M., Graczyk J., Ellerby L. M., La Spada A. R., and Truant R.: Ataxin-7 can export from the nucleus via a conserved exportin-dependent signal. *The Journal of biological chemistry* 281, 2730-2739. (2006)
- [42] Timney B.L., Raveh B., Mironksa R., Trivedi J.M., Kim S.J., Russel D., Wente S.R., Sali A., Rout M.P.: Simple rules for passive diffusion through the nuclear pore complex, *The Journal of Cell Biology*. (2016)
- [43] Truant R., Atwal R. S., and Burtnik A.: Nucleocytoplasmic trafficking and transcription effects of huntingtin in Huntington's disease. *Progress in neurobiology* 83, 211-227. (2007)
- [44] Wüstner D., Solanko L.M., Lund F.W., Sage D., Schroll H.J. and Lomholt M.A.: Quantitative Fluorescence Loss in Photobleaching for Analysis of Protein Transport and Aggregation. *BMC Bioinformatics*, 13, 296. (2012)
- [45] Xia J., Lee D. H., Taylor J., Vandelft M., and Truant R.: Huntingtin contains a highly conserved nuclear export signal. *Human molecular genetics* 12, 1393-1403. (2003)

Part III

An adaptive E-scheme

An Adaptive Viscosity E-scheme for Balance Laws

Ebise Adugna Abdi, Christian Valdemar Hansen and Hans Joachim Schroll

Abstract

An adaptive E-scheme for degenerate, viscous balance laws is presented. The scheme makes use of given, natural viscosity to adaptively reduce artificial diffusion. Numerical experiments demonstrate the improved accuracy of the adaptive scheme. It is proven that explicit and implicit E-schemes are monotone, TVD and nonlinearly stable.

6.1 Introduction

For scalar conservation laws $u_t + f(u)_x = 0$ the entropy solution may be constructed as the vanishing viscosity weak solution to the viscous conservation law $u_t + f(u)_x = du_{xx}$. Numerical methods for hyperbolic conservation laws appeal to that principle by using artificial, vanishing viscosity. For example, the classical Lax–Friedrichs scheme applies the numerical viscosity $d = \epsilon\Delta x$ where $\epsilon = \frac{1}{2}\frac{\Delta x}{\Delta t}$. By the CFL-condition ϵ is bounded from below $2\epsilon \geq \|f'\|_\infty$. In agreement with the concept: sufficient diffusion grants stability, Tadmor [11] showed that any scheme containing more numerical viscosity than Godunov's scheme is entropy–stable. Moreover, it is exactly the class of E–schemes [9] that have no less numerical viscosity than that of Godunov.

In this paper we present an adaptive viscosity E–scheme for degenerate, viscous conservation laws

$$u_t + f(u)_x = (d(x)u_x)_x , \quad d(x) \geq 0 \quad (6.1)$$

making use of the given "natural" diffusion $d(x) \geq 0$ and adding only that much numerical viscosity as needed for stability. Like for Rusanov's [10] local Lax–Friedrichs scheme the numerical viscosity will depend on the local speed of propagation $f'(u)$ turning the effective viscosity $d(x) + \epsilon(u)\Delta x$ nonlinear. The resulting adaptive viscosity scheme is an E–scheme. In section 6.4 we prove that explicit E–schemes are monotone. Applying the calculus of inverse–monotone matrices, it is shown in section 6.5 that also implicit E–schemes are monotone. Using Kröner's version [8] of Harten's theorem [5], it follows that ϑ –time stepping with E–fluxes is a TVD operation, see section 6.6. Numerical experiments in section 6.7 demonstrate the effect of reduced numerical viscosity in the presence of natural diffusion. Finally, in section 6.9 a nonlinear reaction term is included in the analysis and stability of the adaptive E–scheme when applied to balance laws is proven. Conclusions follow in section 6.10.

6.2 A lower bound on the effective diffusion

To implement the idea outlined in the introduction, a quantitative lower bound on the effective diffusion is needed. To this end consider eq. (6.1) with $f \in C^1(\mathbb{R})$, $\|f'\|_\infty < \infty$ and possibly degenerate "natural" diffusion $d(x) \geq 0$. Let $\epsilon\Delta x \geq 0$ denote artificial diffusion and $D = d + \epsilon\Delta x$ the effective, total diffusion. On an uniform mesh $x_j = j\Delta x$, $\Delta x > 0$ the second order central difference operator

$$-\frac{1}{\Delta x^2}\Gamma_D \approx \partial_x(D\partial_x)$$

is given by a symmetric matrix. At inner grid points it has the local structure

$$\Gamma_D \sim \begin{pmatrix} D_{j-3/2} + D_{j-1/2} & -D_{j-1/2} & & \\ -D_{j-1/2} & D_{j-1/2} + D_{j+1/2} & -D_{j+1/2} & \\ & -D_{j+1/2} & D_{j+1/2} + D_{j+3/2} & \end{pmatrix} .$$

The convection term is discretized as

$$f(u)_x \approx \frac{1}{2\Delta x} \Lambda \phi(u) ,$$

where ϕ denotes the diagonal field $\phi(u) = (\dots, f(u_{j-1}), f(u_j), f(u_{j+1}), \dots)^T$ and Λ is the anti-symmetric, central difference operator

$$\Lambda \sim \begin{pmatrix} 0 & 1 & & \\ -1 & 0 & 1 & \\ & -1 & 0 & \end{pmatrix} .$$

The convection diffusion operator $-f(u)_x + (d(x)u_x)_x$ is discretized by central differences

$$F_{\Delta x}(u) = -\frac{1}{\Delta x^2} \Gamma_D u - \frac{1}{2\Delta x} \Lambda \phi(u) .$$

The classical, central difference scheme has no artificial diffusion, while Lax–Friedrichs uses $\epsilon = \frac{1}{2} \frac{\Delta x}{\Delta t}$. Rusanov's local Lax–Friedrichs scheme adapts the numerical viscosity locally

$$\epsilon_{j+1/2} = \frac{1}{2} \max_u |f'(u)| , \quad \forall u \in \overline{u_j, u_{j+1}} . \quad (6.2)$$

The forward marching scheme

$$u^{n+1} = u^n + \Delta t F_{\Delta x}(u^n) = \mathcal{H}(u^n)$$

is monotone in the sense of Crandall and Majda [2] if $\mathcal{H} = I + \Delta t F_{\Delta x}$ is a non-decreasing function in all unknowns. In particular, the Jacobian $D\mathcal{H}$ is off-diagonal non-negative, or quasi-positive. Whenever diffusion $D = d + \epsilon \Delta x$ does not depend on u , the Jacobian reads

$$D\mathcal{H}(u) = -\frac{1}{\Delta x^2} \Gamma_D - \frac{1}{2\Delta x} \text{Adiag}(f'(u)) .$$

Quasi-positivity is ensured by enough diffusion

$$2D_{j\pm 1/2} \geq \Delta x \|f'\|_\infty . \quad (6.3)$$

Apparently, classical, central differences require enough "natural" diffusion in relation to the mesh size. The CFL condition $\frac{\Delta t}{\Delta x} \|f'\|_\infty \leq 1$ ensures that the Lax–Friedrichs scheme has enough diffusion $\epsilon \Delta x = \frac{1}{2} \frac{\Delta x^2}{\Delta t}$, even in the absence of natural diffusion. Monotonicity of the local Lax–Friedrichs scheme, where $\epsilon = \epsilon(u)$ does depend on u , is not discussed as easily by taking derivatives. However, as E-scheme, local Lax–Friedrichs is monotone as well, see section 6.4.

6.3 An adaptive viscosity E-scheme

To be on the safe side, Lax–Friedrichs applies artificial diffusion enough to approximate inviscid conservation laws disregarding any given viscosity; in fact, the scheme is designed for hyperbolic problems. Classical central differences, on the other hand, need positive natural diffusion and fine meshes. They cannot deal with locally degenerate viscosity and become unstable in the limit $d \rightarrow 0$. When dealing with diffusive conservation laws, it is therefore tempting to adopt the artificial viscosity to the characteristic speed of the hyperbolic operator $f'(u)$ and the available natural diffusion $d(x) \geq 0$. To satisfy the "enough diffusion" condition eq. (6.3) it is sufficient to set

$$\epsilon_{j+1/2} = \max \left(0, \frac{1}{2} |f'|_{j+1/2} - \frac{1}{\Delta x} d_{j+1/2} \right) , \quad (6.4)$$

where $d_{j+1/2}$ means $d(x_{j+1/2})$, $x_{j+1/2} = (j + 1/2)\Delta x$ while $|f'|_{j+1/2}$ denotes the local maximum

$$|f'|_{j+1/2} = \max_u |f'(u)| , \quad \forall u \in \overline{u_j, u_{j+1}} .$$

Note that this choice avoids artificial anti-diffusion $\epsilon \geq 0$ and guarantees the local condition

$$2D_{j+1/2} \geq \Delta x |f'|_{j+1/2} . \quad (6.5)$$

The adaptive Lax–Friedrichs flux

$$F(u_j, u_{j+1}) = \frac{1}{2} (f(u_j) + f(u_{j+1})) - \frac{D_{j+1/2}}{\Delta x} (u_{j+1} - u_j) \quad (6.6)$$

with

$$D_{j+1/2} = d_{j+1/2} + \epsilon_{j+1/2} \Delta x \quad (6.7)$$

and ϵ according to eq. (6.4) is consistent with the viscous conservation law

$$u_t + f(u)_x = (d(x)u_x)_x , \quad d(x) \geq 0$$

in the sense that $F(u, u) = f(u)$. While the non-linear artificial diffusion $\epsilon_{j+1/2} = \epsilon(u_j, u_{j+1})$ turns the stability analysis in terms of derivatives $DF_{\Delta x}(u)$ rather complicated, it is not difficult to verify that the adaptive viscosity scheme is an E-scheme. According to Osher [9], a three point scheme in conservation form

$$u_j^{n+1} = u_j^n - \frac{\Delta t}{\Delta x} \left(F(u_j^n, u_{j+1}^n) - F(u_{j-1}^n, u_j^n) \right) \quad (6.8)$$

is called E-scheme if its flux satisfies the E-property

$$\text{sign}(u_{j+1} - u_j) (F(u_j, u_{j+1}) - f(u)) \leq 0 , \quad \forall u \in \overline{u_j, u_{j+1}} . \quad (6.9)$$

A flux satisfying eq. (6.9) is also called E-flux.

Lemma 6.3.1. *The numerical flux eq. (6.6) with sufficient diffusion eq. (6.5) is an E-flux.*

Proof. Assume $u_j \leq u \leq u_{j+1}$ and find

$$\begin{aligned} f(u_{j+1}) &= f(u) + f'(\xi)(u_{j+1} - u) , \quad \xi \in [u, u_{j+1}] \\ f(u_j) &= f(u) - f'(\eta)(u - u_j) , \quad \eta \in [u_j, u] . \end{aligned}$$

Using eq. (6.5)

$$\begin{aligned} \frac{1}{2}(f(u_j) + f(u_{j+1})) &= f(u) + \frac{1}{2}f'(\xi)(u_{j+1} - u) - \frac{1}{2}f'(\eta)(u - u_j) \\ &\leq f(u) + \frac{D_{j+1/2}}{\Delta x}(u_{j+1} - u) + \frac{D_{j+1/2}}{\Delta x}(u - u_j) \\ &= f(u) + \frac{D_{j+1/2}}{\Delta x}(u_{j+1} - u_j) . \end{aligned}$$

By definition of the flux eq. (6.6) the E-property eq. (6.9) follows. The remaining other case $u_j \geq u \geq u_{j+1}$ is completely analogous. \square

As eq. (6.4) implies eq. (6.5) it is clear that

Corollary 6.3.2. *The central scheme eqs. (6.6) to (6.8) with numerical viscosity adapted by eq. (6.4) is E-scheme.*

6.4 E-schemes are monotone

A three-point scheme in conservation form eq. (6.8) is monotone if

$$H(u_{j-1}, u_j, u_{j+1}) = u_j - \frac{\Delta t}{\Delta x} (F(u_j, u_{j+1}) - F(u_{j-1}, u_j))$$

is non-decreasing in all its arguments [2]. By setting $f(u) = F(u, u)$ in the E-condition eq. (6.9) it is obvious that a monotone and consistent scheme eq. (6.8) is E-scheme. To show that consistent E-schemes are monotone, consider first

Lemma 6.4.1. *A consistent, three point E-scheme eqs. (6.8) and (6.9) is quasi-monotone; that is $\partial_v F(v, w) \geq 0$ and $\partial_w F(v, w) \leq 0$.*

Proof. Assume $u_j < u_{j+1}$. By the E-property eq. (6.9)

$$F(u_j, u_{j+1}) \leq f(u) , \quad \forall u \in [u_j, u_{j+1}] .$$

Selecting $u = u_{j+1}$:

$$F(u_j, u_{j+1}) \leq f(u_{j+1}) = F(u_{j+1}, u_{j+1}) .$$

We find that $F(v, w)$ is non-decreasing in v . Choosing $u = u_j$ instead:

$$F(u_j, u_{j+1}) \leq f(u_j) = F(u_j, u_j) .$$

Apparently $F(v, w)$ is non-increasing in w . Similar arguments in the case $u_j > u_{j+1}$ complete the proof. \square

Lemma 6.4.2. *If $\partial_v F(v, w) \geq 0$, $\partial_w F(v, w) \leq 0$ and the CFL-condition holds*

$$\frac{\Delta t}{\Delta x} \|\partial_v F - \partial_w F\|_\infty \leq 1 , \quad (6.10)$$

then the scheme eq. (6.8) is monotone.

Proof.

$$\begin{aligned} \partial_{u_{j-1}} H(u_{j-1}, u_j, u_{j+1}) &= \frac{\Delta t}{\Delta x} \partial_{u_{j-1}} F(u_{j-1}, u_j) \geq 0 . \\ \partial_{u_{j+1}} H(u_{j-1}, u_j, u_{j+1}) &= -\frac{\Delta t}{\Delta x} \partial_{u_{j+1}} F(u_j, u_{j+1}) \geq 0 . \end{aligned}$$

The CFL-condition eq. (6.10) ensures that

$$\partial_{u_j} H(u_{j-1}, u_j, u_{j+1}) = 1 - \frac{\Delta t}{\Delta x} (\partial_{u_j} F(u_j, u_{j+1}) - \partial_{u_j} F(u_{j-1}, u_j)) \geq 0 .$$

This completes the proof. \square

The main result of this section is an obvious consequence of lemmas 6.4.1 and 6.4.2:

Corollary 6.4.3. *A consistent, three-point E-scheme eqs. (6.8) and (6.9) is monotone if the CFL-condition eq. (6.10) holds.*

Finally, the adaptive viscosity scheme as E-scheme is also monotone:

Corollary 6.4.4. *Under the CFL-condition eq. (6.10), the adaptive viscosity scheme eqs. (6.4) and (6.6) to (6.8) is monotone.*

6.5 Implicit time stepping

As diffusive problems tend to be stiff, it is appropriate to consider implicit time stepping applied to the semi-discrete system $u_t = F_{\Delta x}(u)$. The discretization in space is conservative

$$F_{\Delta x}(u)_j = -\frac{1}{\Delta x} (F(u_j, u_{j+1}) - F(u_{j-1}, u_j)) \quad (6.11)$$

with a consistent E-flux eq. (6.9). For such systems, consider ϑ -time stepping

$$\frac{1}{\Delta t} (u^{n+1} - u^n) = \vartheta F_{\Delta x}(u^{n+1}) + (1 - \vartheta) F_{\Delta x}(u^n) , \quad 0 \leq \vartheta \leq 1 . \quad (6.12)$$

The parameter ϑ determines the method: $\vartheta = 0/0.5/1$ corresponds to explicit Euler / Crank–Nicholson / implicit Euler, respectively. Sorting out next and previous time levels

$$\frac{1}{\Delta t} u^{n+1} - \vartheta F_{\Delta x}(u^{n+1}) = \frac{1}{\Delta t} u^n + (1 - \vartheta) F_{\Delta x}(u^n)$$

we define the left– and right–hand side operators

$$L_h(u) = \frac{1}{\Delta t} u - \vartheta F_{\Delta x}(u) , \quad R_h(u) = \frac{1}{\Delta t} u + (1 - \vartheta) F_{\Delta x}(u) .$$

To investigate the monotonicity of the implicit ϑ –scheme, consider two solutions

$$L_h(u^{n+1}) - L_h(v^{n+1}) = R_h(u^n) - R_h(v^n) .$$

Let

$$C_{\Delta x}^n = \int_0^1 DF_{\Delta x}(v^n + s(u^n - v^n)) ,$$

$$A_h^n = \frac{1}{\Delta t} I - \vartheta C_{\Delta x}^n , \quad B_h^n = \frac{1}{\Delta t} I + (1 - \vartheta) C_{\Delta x}^n .$$

It follows

$$A_h^n(u^{n+1} - v^{n+1}) = B_h^n(u^n - v^n) .$$

By lemma 6.4.1 the Jacobian $DF_{\Delta x}$ thus $C_{\Delta x}^n$ and B_h^n are quasi–positive while A_h^n is Z–matrix. For Δt small enough

$$(1 - \vartheta) \frac{\Delta t}{\Delta x} \|\partial_v F - \partial_w F\|_\infty \leq 1 , \quad (6.13)$$

then B_h^n is non–negative. From the local structure of the Jacobian $DF_{\Delta x}$ ~

$$-\frac{1}{\Delta x} \begin{pmatrix} * & F_w(u_{j-1}, u_j) \\ -F_v(u_{j-1}, u_j) & F_v(u_j, u_{j+1}) - F_w(u_{j-1}, u_j) & F_w(u_j, u_{j+1}) \\ & -F_v(u_j, u_{j+1}) & * \end{pmatrix} \quad (6.14)$$

it is obvious that inner columns –but not rows– of $DF_{\Delta x}$ sum to zero i.e. $DF_{\Delta x}^T \delta \leq 0$ where $\delta = (1, 1, \dots, 1)^T$ denotes the vector of ones. It follows $(C_{\Delta x}^n)^T \delta \leq 0$ and

$$(A_h^n)^T \delta \geq \frac{1}{\Delta t} > 0 .$$

By the M–criterion (Bohl [1], Theorem I.4.3 or Fiedler and Pták [3] Theorem (4,3)) A_h^n is M–matrix i.e. $(A_h^n)^{-1} \geq 0$. Finally, also implicit E–schemes are monotone in the sense:

$$u^{n+1} - v^{n+1} \geq (A_h^{n+1})^{-1} B_h^n(u^n - v^n) , \quad (A_h^{n+1})^{-1} B_h^n \geq 0 .$$

If initially $u^0 \geq v^0$ (componentwise for all j), then the same is true at later time.

Lemma 6.5.1. *Under the time step restriction eq. (6.13) ϑ -time stepping eq. (6.12) for a consistent and conservative eq. (6.8) E-flux eq. (6.9) is a monotone operation.*

It is well-known that monotone methods are total variation diminishing (TVD). In the next section, we present a direct approach to TVD.

6.6 E-schemes are TVD

Applying Harten's theorem [5] we shall see that the CFL-condition guarantees R_h to be a TVD operator. Moreover, Kröner's extension of Harten's theorem turns L_h into a total variation increasing (TVI) operator. Consequently the ϑ -scheme $L_h(u^{n+1}) = R_h(u^n)$ is TVD for any value of $\vartheta \in [0, 1]$.

Theorem 6.6.1 (Harten [5]). *The method*

$$u_j^{n+1} = u_j^n + \alpha_{j+1/2} (u_{j+1}^n - u_j^n) - \beta_{j-1/2} (u_j^n - u_{j-1}^n)$$

is TVD if $\alpha_{j+1/2} \geq 0$, $\beta_{j-1/2} \geq 0$ and $\alpha_{j+1/2} + \beta_{j+1/2} \leq 1$.

Corollary 6.6.2. *Let $F(v, w)$ be a consistent, Lipschitz continuous*

$$\left. \begin{array}{l} F(v, w) - f(w) \\ F(v, w) - f(v) \end{array} \right\} \geq -L(w - v) \quad (6.15)$$

E-flux eq. (6.9). Under the CFL-condition

$$(1 - \vartheta) \frac{\Delta t}{\Delta x} 2L \leq 1 \quad (6.16)$$

R_h is a TVD operator: $TV(R_h(u)) \leq TV(u)$.

Proof. The E-condition eq. (6.9) implies the flux to be quasi-monotone, see lemma 6.4.1 i.e. $\partial_u F(u, v) \geq 0$ and $\partial_v F(u, v) \leq 0$. Hence,

$$\begin{aligned} C_{j+1/2} &= \frac{F(u_j, u_{j+1}) - f(u_j)}{u_{j+1} - u_j} \leq 0 , \\ E_{j+1/2} &= \frac{F(u_j, u_{j+1}) - f(u_{j+1})}{u_{j+1} - u_j} \leq 0 , \end{aligned}$$

and $C_{j+1/2} + E_{j+1/2} \geq -2L$. By definition

$$[\Delta t R_h(u)]_j = u_j + \alpha_{j+1/2} (u_{j+1} - u_j) - \beta_{j-1/2} (u_j - u_{j-1})$$

with

$$\begin{aligned} \alpha_{j+1/2} &= -(1 - \vartheta) \frac{\Delta t}{\Delta x} C_{j+1/2} \geq 0 , \\ \beta_{j+1/2} &= -(1 - \vartheta) \frac{\Delta t}{\Delta x} E_{j+1/2} \geq 0 , \end{aligned}$$

and $\alpha_{j+1/2} + \beta_{j+1/2} \leq (1 - \vartheta) \frac{\Delta t}{\Delta x} 2L \leq 1$. The claim follows by theorem 6.6.1. \square

Using the same coefficients $C_{j+1/2}$ and $E_{j+1/2}$ the left-hand side reads

$$[\Delta t L_h(u)]_j = u_j + \tilde{\alpha}_{j+1/2} (u_{j+1} - u_j) - \tilde{\beta}_{j-1/2} (u_j - u_{j-1})$$

with non-positive coefficients

$$\tilde{\alpha}_{j+1/2} = \vartheta \frac{\Delta t}{\Delta x} C_{j+1/2} \leq 0 ,$$

$$\tilde{\beta}_{j+1/2} = \vartheta \frac{\Delta t}{\Delta x} E_{j+1/2} \leq 0 .$$

A direct application of Kröner's lemma shows that L_h is a TVI operator.

Lemma 6.6.3 ([8] Lemma 2.3.20). *Consider*

$$u_j^{n+1} = u_j^n + \alpha_{j+1/2} (u_{j+1}^n - u_j^n) - \beta_{j-1/2} (u_j^n - u_{j-1}^n) .$$

If $\alpha_{j+1/2} \leq 0$ and $\beta_{j-1/2} \leq 0$, then $TV(u^{n+1}) \geq TV(u^n)$.

It follows that the ϑ -scheme is TVD:

Corollary 6.6.4. *Assume F is a consistent, Lipschitz continuous eq. (6.15) E-flux eq. (6.9). Given the CFL-condition eq. (6.16) the ϑ -scheme $L_h(u^{n+1}) = R_h(u^n)$ is TVD.*

Proof. $TV(u^{n+1}) \leq TV(L_h(u^{n+1})) = TV(R_h(u^n)) \leq TV(u^n)$. \square

6.7 Numerical experiments

The performance of the adaptive viscosity scheme will be demonstrated for scalar conservations laws in one and two space dimensions as well as for a degenerate Navier–Stokes system.

6.7.1 A single conservation law

This first example is constructed to demonstrate the effect of reduced artificial diffusion. Consider the viscous advection equation

$$u_t + u_x = (d(x)u_x)_x \quad (6.17)$$

with degenerate diffusion

$$d(x) = \max \left(0, \frac{1}{40} \cos \left(\frac{\pi x}{6} \right) \right) .$$

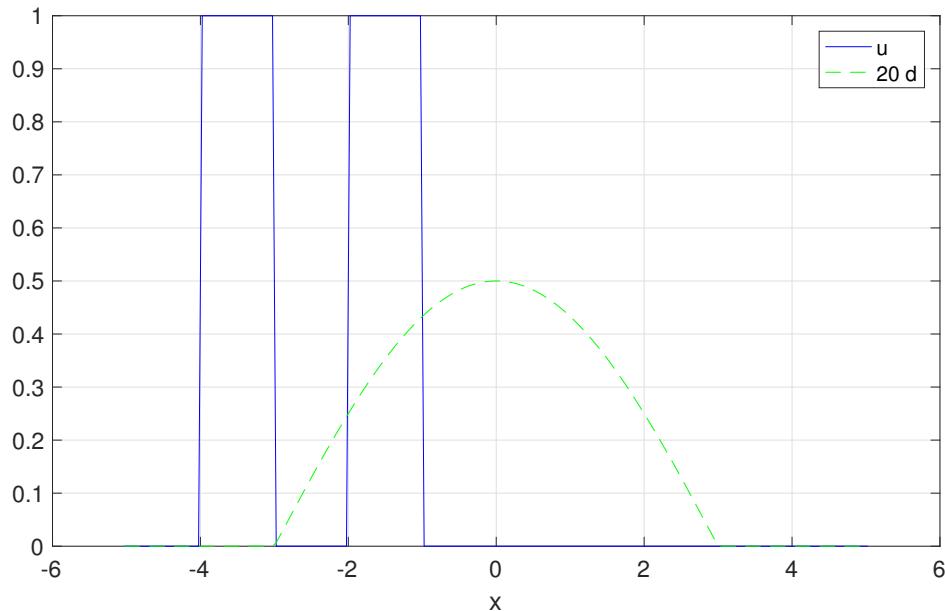
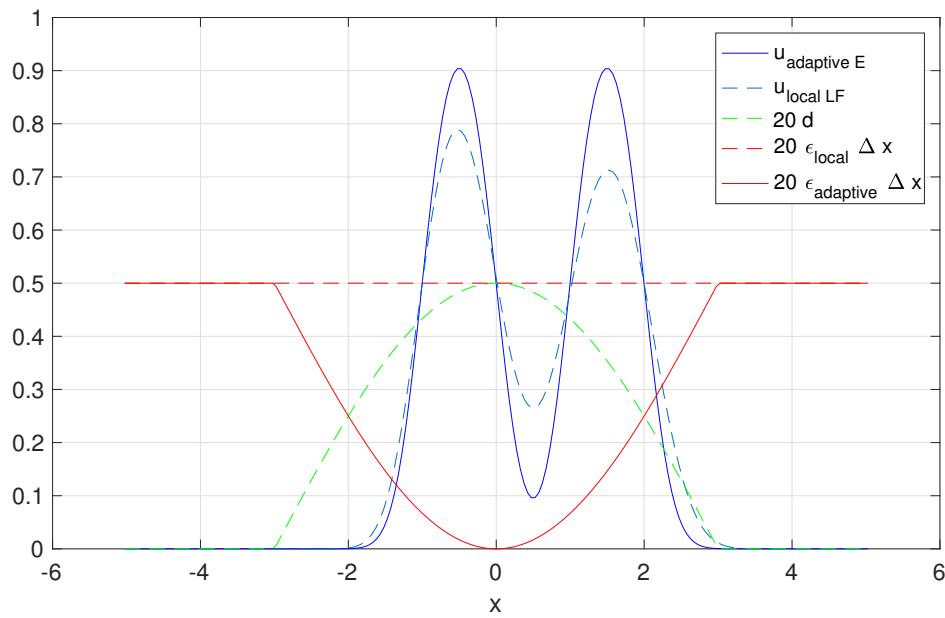


Figure 6.1: Initial data and upscaled diffusion coefficient.

Figure 6.2: Approximations to eq. (6.17) at $t = 3$ with given-, local and adaptive viscosities, $\Delta x = 1/20$.

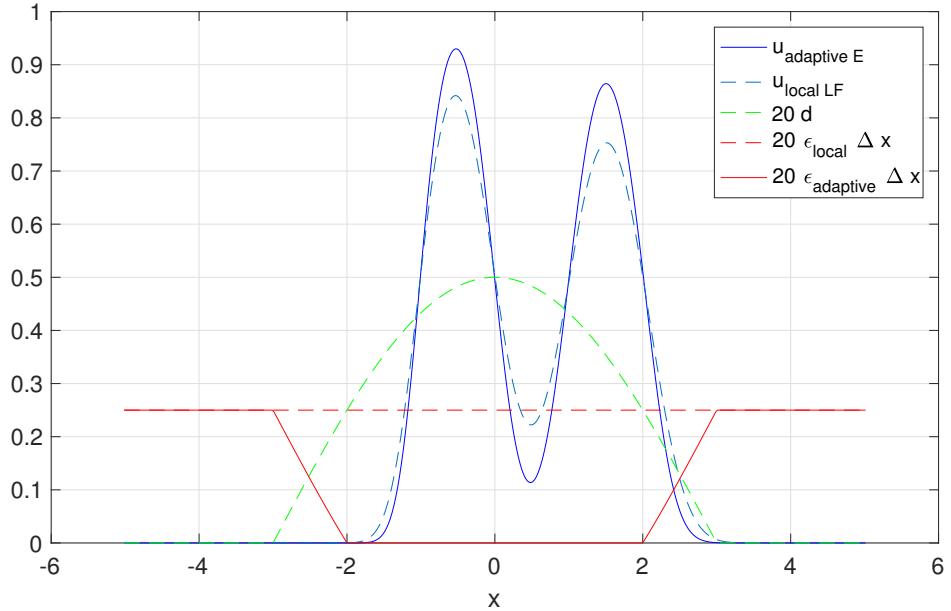


Figure 6.3: Approximations to eq. (6.17) at $t = 3$ on refined mesh $\Delta x = 1/40$.

The initial data $u|_{t=0} = \chi_{[-4,-3] \cup [-2,-1]}$ and the diffusion coefficient are displayed in fig. 6.1. Approximate solutions are computed on the interval $[-5, 5]$ using a mesh size of $\Delta x = 1/20$ and applying periodic boundary conditions.

Approximations generated by the local Lax–Friedrichs scheme in comparison to the adaptive viscosity method are displayed in fig. 6.2. Explicit Euler time stepping is applied. By corollary 6.4.4 the time step should be limited by eq. (6.10). With local and adaptive viscosity coefficients however, the partial derivatives $\partial_v F$ and $\partial_w F$ may be complicated. A good practical compromise is to limit

$$\frac{\Delta t}{\Delta x^2} \max_j (D_{j+1/2} + D_{j-1/2}) = \text{CFL} , \quad (6.18)$$

with a CFL number less than one. In this section for scalar equations in one space variable the CFL limit was set to 0.8. With constant characteristic speed $f'(u) = 1$ local Lax–Friedrichs applies constant artificial viscosity $2\epsilon_l = 1$ according to eq. (6.2) and disregarding any given viscosity. The adaptive scheme however, reduces artificial diffusion where natural diffusion is available. This leads to less effective diffusion $D = d + \epsilon\Delta x$ and reduced damping, see fig. 6.2.

Artificial viscosities scale with the mesh size. On a refined mesh $\Delta x = 1/40$ the adaptive scheme does not require any artificial viscosity for $x \in [-2, 2]$, see fig. 6.3. In fact the difference between both approximations measured in L^1 -norm scales linearly with the mesh size. table 6.1 confirms that both schemes converge to the same solution.

Δx	L^1 difference	rate
1/20	0.40451	-
1/40	0.27122	0.58
1/80	0.14990	0.86
1/160	0.07991	0.91
1/320	0.041513	0.94

Table 6.1: L^1 deviation between the local Lax–Friedrichs and the adaptive E-scheme applied to eq. (6.17).

Similar experiments with the nonlinear, viscous Burgers' equation

$$u_t + \frac{1}{2}(u^2)_x = (d(x)u_x)_x \quad (6.19)$$

confirm the observations. Both the artificial viscosities and the difference between both schemes scale linearly with the mesh size, see figs. 6.4 and 6.5 and table 6.2.

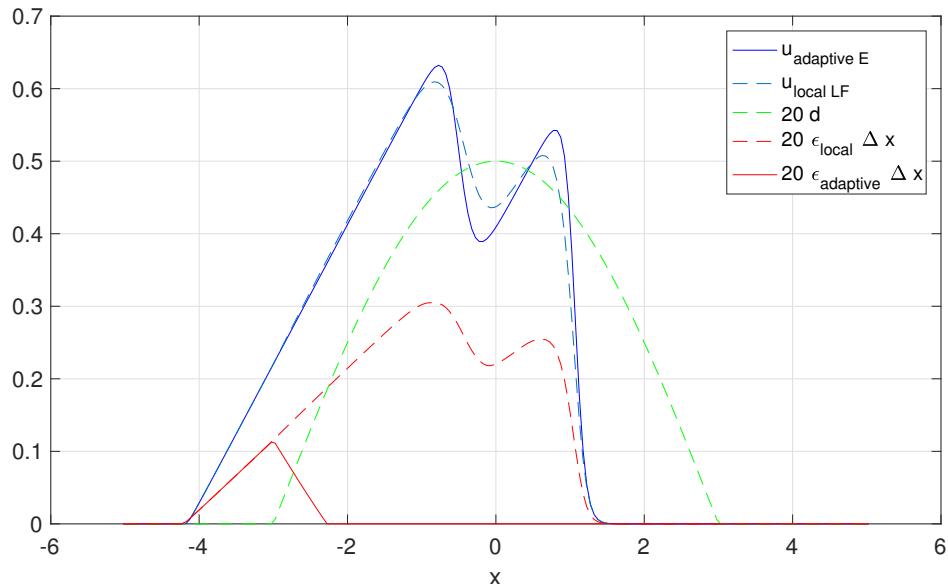


Figure 6.4: Approximations to Burgers' equation eq. (6.19) at time $t = 5$, $\Delta x = 1/20$.

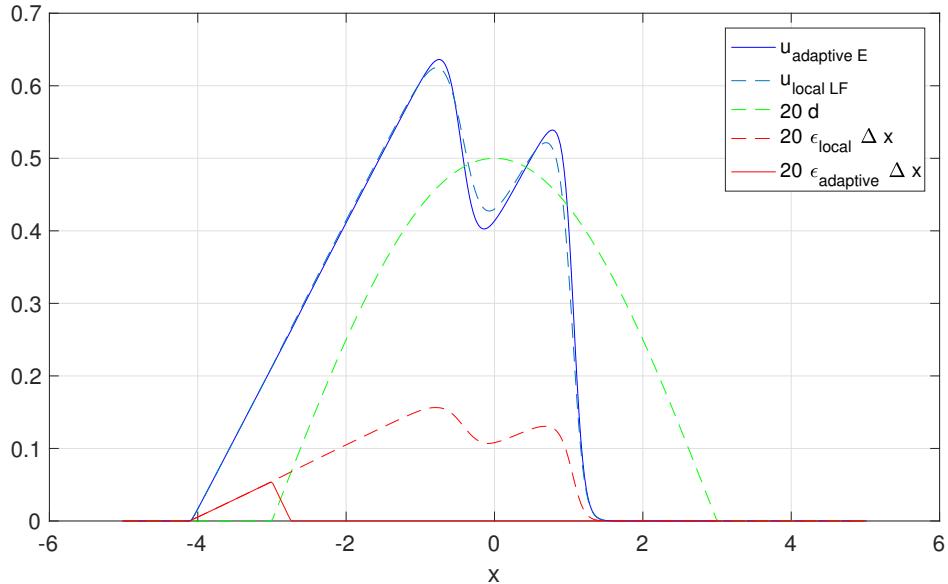


Figure 6.5: Approximations to eq. (6.19) at time $t = 5$ on refined mesh $\Delta x = 1/40$.

Δx	L^1 difference	rate
1/20	0.086074	-
1/40	0.047870	0.85
1/80	0.024012	1.00
1/160	0.012074	0.99
1/320	0.006063	0.99

Table 6.2: L^1 deviation between the local Lax–Friedrichs and the adaptive E–scheme applied to eq. (6.19).

In the next experiment we approximate the solution to the conservation law

$$u_t + \nabla \cdot f(u) = \nabla \cdot (d(x)\nabla u) , \quad f(u) = \left(u^2/2, u^4/4 \right)^T \quad (6.20)$$

in two variables $|x| \leq 10$ and $|y| \leq 6$. The viscosity coefficient depends on x alone. It is largest around $x = 0$ and fades out towards the left and right boundaries.

$$d(x) = \max \left(0, \frac{1}{40} \left(\frac{1}{2} + \sin \left(\frac{\pi(x+5)}{10} \right) \right) \right) .$$

Initial data is piecewise constant. In square blocks of size 4×4 units, the data shown in table 6.3 is given. In the background, the u –value is zero. Finally, boundary conditions are periodic in both variables.

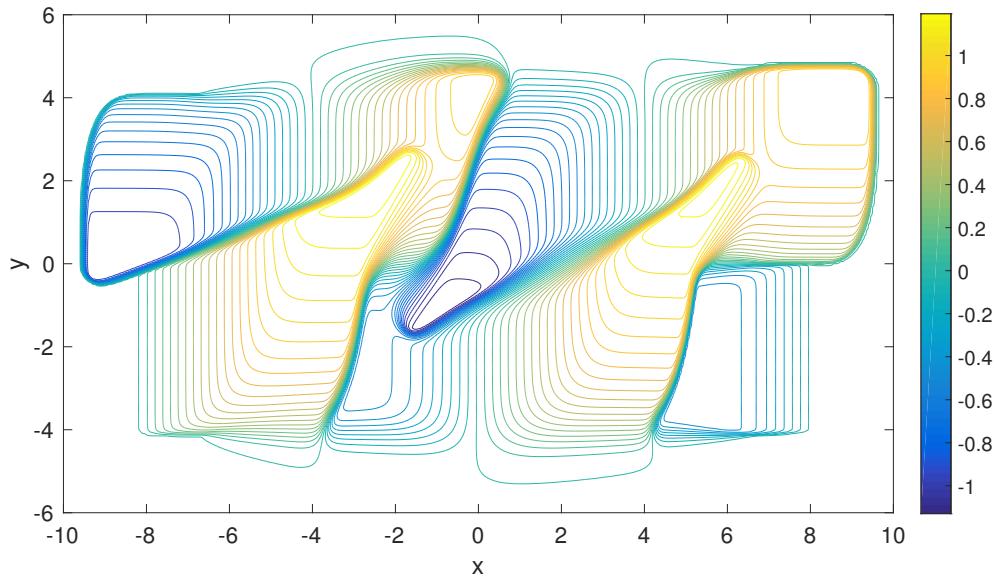


Figure 6.6: Approximation to eq. (6.20) by the adaptive E-scheme at $t = 3$, $\Delta x = \Delta y = 1/20$.

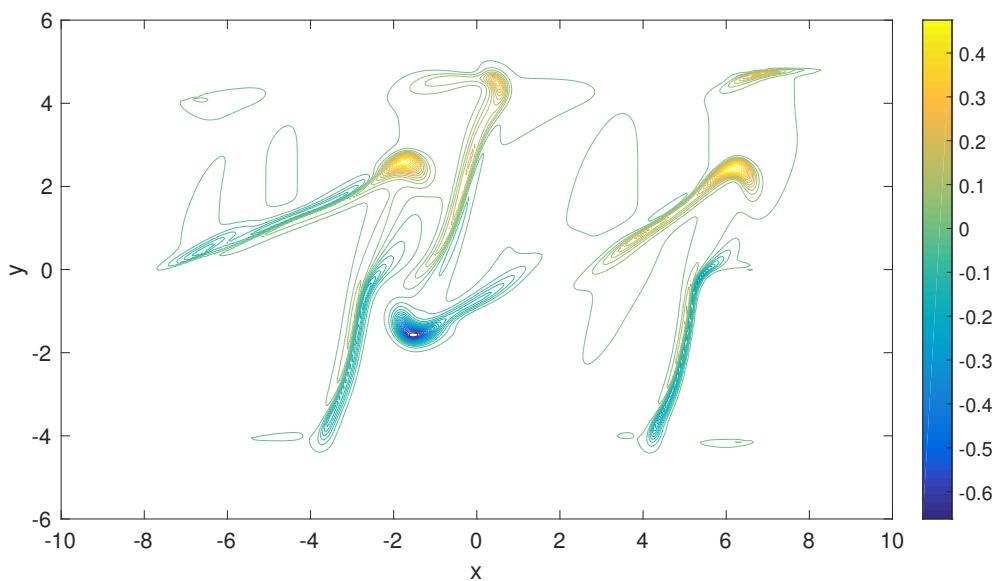


Figure 6.7: Deviation of adaptive E-scheme from local Lax-Friedrichs, $t = 3$.

-1.0	+1.0	-1.5	+1.0
+1.5	-0.5	+1.5	-0.5

Table 6.3: Piecewise constant initial data.

The one dimensional schemes are applied in both coordinate directions. Approximations are advanced in time by an explicit Euler step. The time-step is limited by eq. (6.18) applied to both directions and with a CFL limit of 0.5.

The approximation by the adaptive E-scheme on a 400×240 points mesh is displayed in fig. 6.6 with the difference of both schemes seen in fig. 6.7.

A cross-section (fig. 6.8) along the line $y = -1.3$ reveals large differences around $x = -1$. While the adaptive scheme resolves the minimum already on the coarse grid, local Lax–Friedrichs can only hardly sense it. Mesh refinement confirms the location of the minimum. Apparently both schemes converge to the same solution.

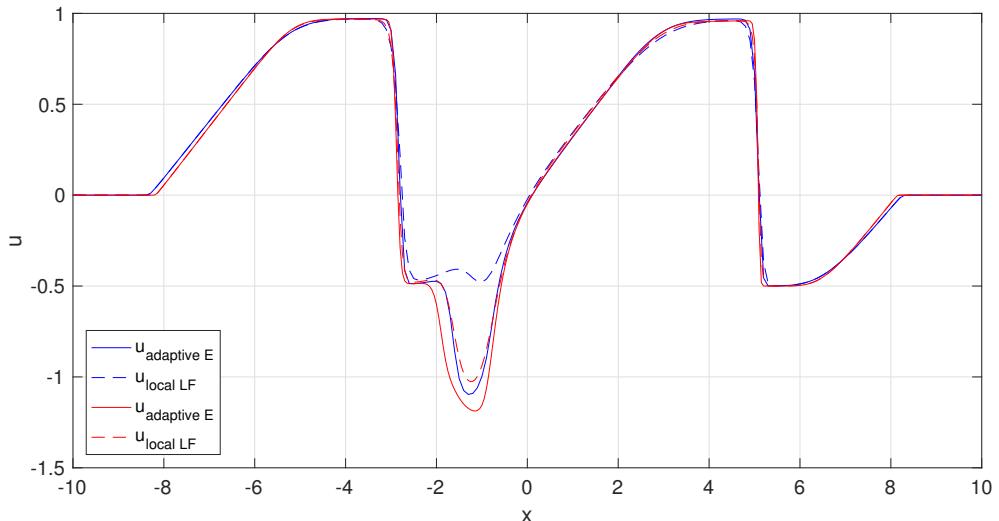


Figure 6.8: Cross-section at $y = -1.3$, $t = 3$, $\Delta x = \Delta y = 1/10$ (blue) and $\Delta x = \Delta y = 1/20$ (red).

6.7.2 Conservation laws in gas dynamics

The Navier–Stokes equations [7] describe conservation of mass

$$\rho_t + \nabla \cdot (\rho \mathbf{u}) = 0 ,$$

and momentum

$$\rho D\mathbf{u}/Dt + \nabla p = S .$$

Here ρ is the mass density, \mathbf{u} stands for the flow velocity and $D/Dt = \partial/\partial t + \mathbf{u} \cdot \nabla$ denotes the material derivative. Assuming constant entropy, the system is closed by an equation of state $p = p(\rho)$, for example the gamma-law $p = \rho^\gamma$ with $\gamma = 1.4$ for air under normal conditions. The viscous stress tensor is given by

$$S = \nabla((\mu + \lambda)\nabla \cdot \mathbf{u}) + \nabla \cdot (\mu \nabla \mathbf{u}) .$$

In conservative form and in one space variable, the system reads

$$\begin{pmatrix} \rho \\ \rho u \end{pmatrix}_t + \begin{pmatrix} \rho u \\ \rho u^2 + p \end{pmatrix}_x = \begin{pmatrix} 0 \\ (2\mu + \lambda)u_x \end{pmatrix}_x . \quad (6.21)$$

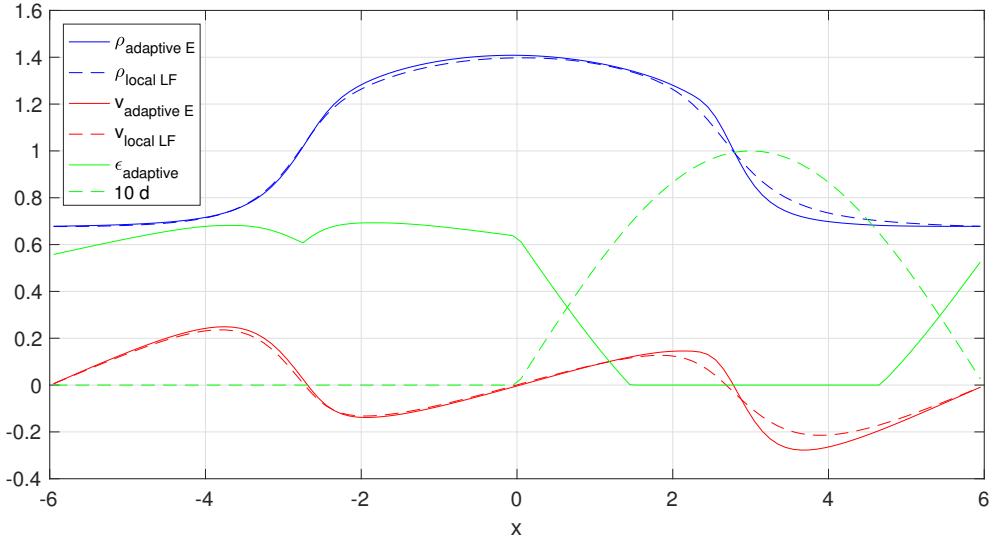


Figure 6.9: Approximations to Navier–Stokes system (6.21), $\Delta x = 1/10$ at $t = 5$.

fig. 6.9 shows the adaptive E-scheme in comparison to local Lax–Friedrichs when applied to the one-dimensional system eq. (6.21) with given diffusion

$$2\mu + \lambda = \max \left(0, \frac{1}{10} \sin \left(\frac{x\pi}{6} \right) \right) .$$

Note that for negative $x < 0$ the system degenerates to the inviscid, isentropic Euler equations. The initial density is $\rho(0, x) = 1.0 - 0.5 \cos(x\pi/6)$ with zero initial momentum and periodic boundary conditions. The mesh size is $\Delta x = 0.1$ with CFL number 0.9. Due to zero artificial diffusion around $x = 3$, the approximation by the adaptive E-scheme appears much sharper

than the local Lax–Friedrichs scheme that suffers from unnecessary numerical diffusion and overdamping.

Our last experiment simulates "dancing" waves in the two-dimensional Navier–Stokes system

$$\begin{aligned} \left(\begin{array}{c} \rho \\ \rho u \\ \rho v \end{array} \right)_t + \left(\begin{array}{c} \rho u \\ \rho u^2 + p \\ \rho u v \end{array} \right)_x + \left(\begin{array}{c} \rho v \\ \rho u v \\ \rho v^2 + p \end{array} \right)_y = \\ \left(\begin{array}{c} 0 \\ (2\mu + \lambda)u_x + (\mu + \lambda)v_y \\ \mu v_x \end{array} \right)_x + \left(\begin{array}{c} 0 \\ \mu u_y \\ (\mu + \lambda)u_x + (2\mu + \lambda)v_y \end{array} \right)_y . \quad (6.22) \end{aligned}$$

with degenerate diffusion

$$\mu = \frac{2}{10} \max \left(0, \frac{2}{10} + \sin \left(\frac{x\pi}{6} \right) \sin \left(\frac{y\pi}{6} \right) \right) , \quad \lambda = -2\mu/3 .$$

The initial density is a perturbed sin-wave

$$\rho(0, x, y) = 1 + \frac{1}{10} \sin \left(\frac{(x-6)\pi}{3} \right) \sin \left(\frac{(y+3)\pi}{3} \right)$$

with $\rho(0, x, y) = 0.5$ within the rectangle $[-3, 3] \times [-2, 2]$. At time $t = 0$ the flow is at rest, i.e. $(u, v) = (0, 0)$. The mesh is uniform and square with $\Delta x = \Delta y = 1/20$.

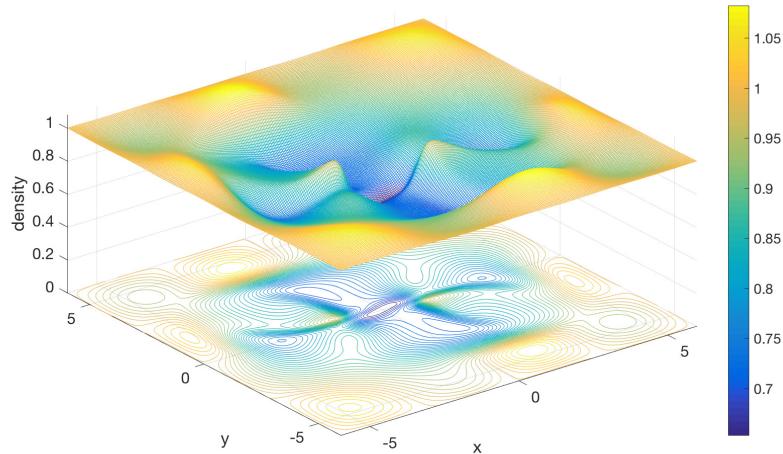


Figure 6.10: Density in Navier–Stokes (6.22) at $t = 1.5$.

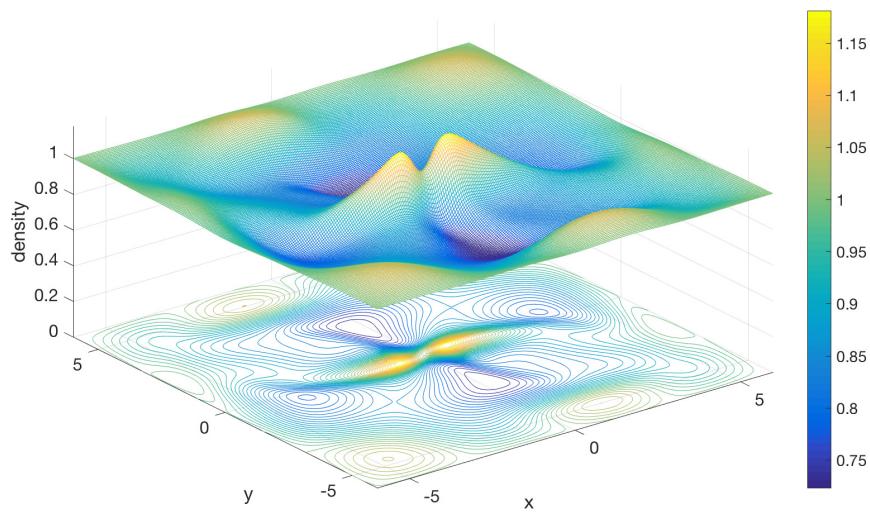


Figure 6.11: Density in Navier–Stokes (6.22) at $t = 2.0$.

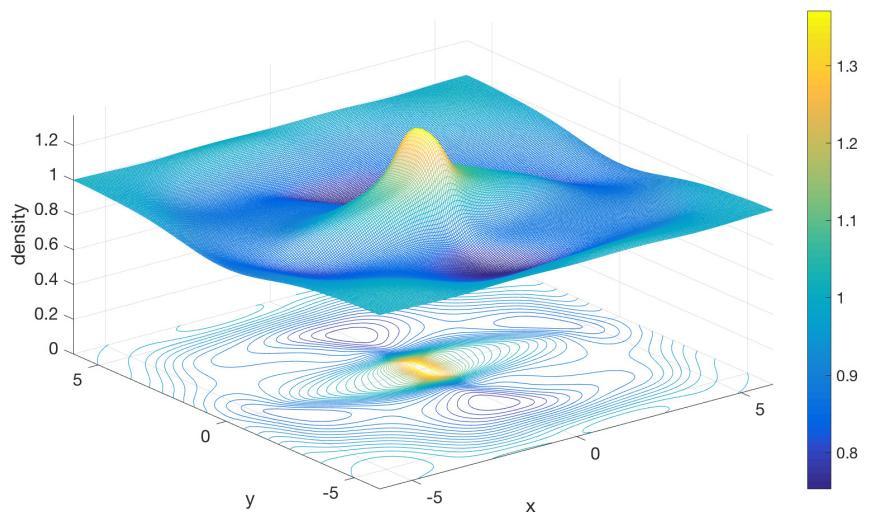


Figure 6.12: Density in Navier–Stokes (6.22) at $t = 2.5$.

Natural viscosities given in (6.22) are $d^x = (0, 2\mu + \lambda, \mu)^T$ and $d^y = (0, \mu, 2\mu + \lambda)^T$. Artificial viscosities are adapted component- and coordinate-wise such that the total, effective viscosity $D = d + \epsilon\Delta x$ is nowhere less than the local characteristic speed in each direction $|u| + \sqrt{p'(\rho)}$ and $|v| + \sqrt{p'(\rho)}$ respectively. As there is no natural diffusion in the continuity equation, the numerical viscosity for ρ is strictly positive. This is in contrast to the momentum equations where numerical viscosities degenerate locally. Despite low and locally degenerate artificial viscosities, the scheme is stable and produces approximations free of spurious oscillations.

The time step is again limited by eq. (6.18) applied coordinate- and componentwise, i.e. the maximum taken over all three components and in both directions to determine a global time step. The CFL number for the simulations shown in fig. 6.10–6.12 is 0.3.

6.8 Boundary conditions

What type of boundary condition to pose with the multiphysics PDE eq. (6.1) depends on the local type of the PDE. In case of periodic data it is appropriate to ask for periodic solutions thus avoiding the identification of specific boundary conditions. For example, assuming 1-periodic initial data $u(0, x) = u(0, 1 + x)$ and diffusion coefficients $d(x) = d(1 + x)$, the 1-periodic solution is to be determined on the mesh

$$x_j = j\Delta x , \quad \Delta x = \frac{1}{M} , \quad j = 1, 2, \dots, M .$$

Identifying u_{M+j} with u_j the conservative space discretization eq. (6.11) reads

$$F_{\Delta x}(u) = -\frac{1}{\Delta x} \begin{pmatrix} F(u_1, u_2) - F(u_M, u_1) \\ F(u_j, u_{j+1}) - F(u_{j-1}, u_j) , \quad j = 2, 3 \dots M-1 \\ F(u_M, u_1) - F(u_{M-1}, u_M) \end{pmatrix} .$$

The Jacobian $DF_{\Delta x}(u)$ has internal structure eq. (6.14) with first and last columns

$$-\frac{1}{\Delta x} \begin{pmatrix} F_v(u_1, u_2) - F_w(u_M, u_1) \\ -F_v(u_1, u_2) \\ 0 \in \mathbb{R}^{M-3} \\ F_w(u_M, u_1) \end{pmatrix} , \quad -\frac{1}{\Delta x} \begin{pmatrix} -F_v(u_M, u_1) \\ 0 \in \mathbb{R}^{M-3} \\ F_w(u_{M-1}, u_M) \\ F_v(u_M, u_1) - F_w(u_{M-1}, u_M) \end{pmatrix}$$

By lemma 6.4.1 for consistent E-schemes the Jacobian is quasi-positive and columns sum to zero $DF(u)_{\Delta x}^T \delta = 0$. Monotonicity follows along the lines in section 6.5.

For parabolic boundaries, that is $d(0) > 0$ at the left boundary for example, also Dirichlet conditions are common. In that case $u(t, 0) = u_0 = \gamma$ is given and the space discretization reduces to

$$F_{\Delta x}(u) = -\frac{1}{\Delta x} \begin{pmatrix} F(u_1, u_2) - F(\gamma, u_1) \\ F(u_2, u_3) - F(u_1, u_2) \\ \vdots \end{pmatrix} .$$

Apparently the Jacobian is quasi-positive

$$-\frac{1}{\Delta x} \begin{pmatrix} F_v(u_1, u_2) - F_w(\gamma, u_1) & F_w(u_1, u_2) \\ -F_v(u_1, u_2) & F_v(u_2, u_3) - F_w(u_1, u_2) & F_w(u_2, u_3) \\ \ddots & \ddots & \ddots \end{pmatrix} .$$

Columns sum to zero except the first column where the sum is short of one term

$$(DF_{\Delta x}^T \delta)_1 = \frac{1}{\Delta x} F_w(\gamma, u_1) \leq 0 .$$

Again $DF_{\Delta x}^T \delta \leq 0$ and monotonicity follows.

Without diffusion $d(0) = 0$ or for a homogenous Neumann condition $u(t, 0)_x = 0$ the boundary is hyperbolic and well posedness of the initial boundary value problem is nontrivial, see [4, 6, 7]. In this situation the in-going characteristics need to be specified.

6.9 Balance laws

Consider

$$u_t + f(u)_x = (d(x)u_x)_x + g(u) \quad (6.23)$$

with

$$f, g \in C^1(\mathbb{R}) , \quad \|f'\|_\infty < \infty , \quad \nu \leq g' \leq \mu , \quad d(x) \geq 0 .$$

E-schemes applied to balance laws are monotone. The following lemma generalizes corollary 6.4.3.

Lemma 6.9.1. *Let F be a consistent E-flux eq. (6.9). Under the CFL-condition*

$$\frac{\Delta t}{\Delta x} \|F_v - F_w\|_\infty \leq 1 + \Delta t \nu \quad (6.24)$$

the augmented discrete operator

$$H(u_{j-1}, u_j, u_{j+1}) = u_j - \frac{\Delta t}{\Delta x} (F(u_j, u_{j+1}) - F(u_{j-1}, u_j)) + \Delta t g(u_j) \quad (6.25)$$

is monotone.

Proof. By lemma 6.4.1 the flux is quasi-monotone. Thus

$$\partial_{u_{j-1}} H = \frac{\Delta t}{\Delta x} \partial_{u_{j-1}} F(u_{j-1}, u_j) \geq 0$$

and

$$\partial_{u_{j+1}} H = -\frac{\Delta t}{\Delta x} \partial_{u_{j+1}} F(u_j, u_{j+1}) \geq 0 .$$

due to the assumption eq. (6.24)

$$\partial_{u_j} H = 1 - \frac{\Delta t}{\Delta x} \left(\partial_{u_j} F(u_j, u_{j+1}) - \partial_{u_j} F(u_{j-1}, u_j) \right) + \Delta t g'(u_j) \geq 0 .$$

□

Moreover, ϑ -time stepping for the operator eq. (6.25)

$$\frac{1}{\Delta t} (u^{n+1} - u^n) = \vartheta (F_{\Delta x} + G) (u^{n+1}) + (1 - \vartheta) (F_{\Delta x} + G) (u^n) , \quad 0 \leq \vartheta \leq 1 \quad (6.26)$$

with the conservative space discretization eq. (6.11) and the diagonal field $G(u)_j = g(u_j)$ is stable in the sense of Hadamard. To this end consider the scheme expressed as an operator root equation

$$T_h = \begin{pmatrix} u^0 - u(0, \cdot) \\ L_h(u^n) - R_h(u^{n-1}) , \quad n = 1, 2, \dots, N \end{pmatrix} = 0 \quad (6.27)$$

with the augmented left- and right-hand sides

$$L_h(v) = \frac{1}{\Delta t} v - \vartheta (F_{\Delta x} + G) (v) , \quad R_h(v) = \frac{1}{\Delta t} v + (1 - \vartheta) (F_{\Delta x} + G) (v) .$$

The discrete solution operator, that is the numerical scheme, is given by the inverse operator $\hat{u} = T_h^{-1}(0)$. Stability means that T_h^{-1} is uniformly Lipschitz continuous

$$\|u - v\| \leq S \|T_h(u) - T_h(v)\| , \quad (6.28)$$

with a stability constant S independent of $h = (\Delta t, \Delta x)$. Taking \tilde{u} as the solution to the balance law eq. (6.23), convergence follows, assuming the method is consistent in the sense

$$\|\tilde{u}|_h - \hat{u}\| \leq S \|T_h(\tilde{u}|_h)\| \rightarrow 0 , \quad h \rightarrow 0 .$$

Lemma 6.9.2. *Under the time step restrictions*

$$\vartheta \mu \Delta t \leq \kappa < 1 , \quad (1 - \vartheta) \left(\frac{\Delta t}{\Delta x} \|\partial_v F - \partial_w F\|_{\infty} - \Delta t \frac{\nu + \mu}{2} \right) \leq 1 ,$$

the ϑ -scheme eq. (6.27) with conservative $F_{\Delta x}$ eq. (6.11) and consistent E-flux eq. (6.9) is stable i.e. there is a stability constant such that eq. (6.28) holds.

Proof. To verify the stability inequality eq. (6.28) it is natural to consider

$$T_h^n(u) - T_h^n(v) = L_h(u^n) - L_h(v^n) - (R_h(u^{n-1}) - R_h(v^{n-1})), \quad n = 1, 2, \dots, N.$$

By the mean value theorem

$$A_h^n(u^n - v^n) = T_h^n(u) - T_h^n(v) + B_h^{n-1}(u^{n-1} - v^{n-1}) \quad (6.29)$$

with

$$\begin{aligned} A_h^n &= \frac{1}{\Delta t} I - \vartheta \int_0^1 D(F_{\Delta x} + G)(\xi(s)) \, ds , \\ B_h^n &= \frac{1}{\Delta t} I + (1 - \vartheta) \int_0^1 D(F_{\Delta x} + G)(\xi(s)) \, ds , \quad \xi(s) = v^n + s(u^n - v^n). \end{aligned}$$

By lemma 6.4.1, the consistent E-flux is quasi-monotone and $DF_{\Delta x}$ is quasi-positive. The additional diagonal field G does not effect quasi-positivity of the augmented Jacobian $D(F_{\Delta x} + G) = DF_{\Delta x} + \text{diag}(g'(u_j))$. Due to the conservation form eq. (6.11) $DF_{\Delta x}^T \delta \leq 0$. By the Lipschitz-bound for g' it holds $D(F_{\Delta x} + G)^T \delta \leq \mu \delta$ and thus

$$(A_h^n)^T \delta \geq \left(\frac{1}{\Delta t} - \vartheta \mu \right) \delta .$$

Under the restriction $\Delta t \vartheta \mu \leq \kappa < 1$, A_h^n is M-matrix and

$$\|(A_h^n)^{-1}\|_1 = \|(A_h^n)^{-1} \delta\|_\infty \leq \frac{1}{\frac{1}{\Delta t} - \vartheta \mu} \leq \frac{\Delta t}{1 - \kappa} = \mathcal{O}(\Delta t). \quad (6.30)$$

With $DF_{\Delta x}$ also B_h^n is quasi-positive. Unless the time step is restricted by eq. (6.24) B_h^n cannot be expected to be monotone. Under the milder condition

$$(1 - \vartheta) \left((DF_{\Delta x})_{jj} - \frac{\nu + \mu}{2} \right) \Delta t \leq 1 .$$

B_h^n is bounded

$$-P_h^n \leq B_h^n \leq P_h^n = \frac{1}{\Delta t} I + (1 - \vartheta) \left(\int_0^1 DF_{\Delta x}(\xi(s)) + \mu I \right)$$

and $P_h^n \geq |B_h^n| \geq 0$ is monotone. Moreover, $(P_h^n)^T \delta \leq \frac{1}{\Delta t} \delta + (1 - \vartheta) \mu \delta$. Thus,

$$\|P^n\|_1 = \|(P^n)^T\|_\infty = \|(P^n)^T \delta\|_\infty \leq \frac{1}{\Delta t} + (1 - \vartheta) \mu .$$

In combination with eq. (6.30)

$$\|(A^n)^{-1} P^{n-1}\|_1 \leq \frac{\Delta t}{1 - \vartheta \mu \Delta t} \left(\frac{1}{\Delta t} + (1 - \vartheta) \mu \right) \leq 1 + \frac{\max(\mu, 0)}{1 - \kappa} \Delta t . \quad (6.31)$$

From eq. (6.29) with eq. (6.30) and eq. (6.31) it follows

$$\|u^n - v^n\|_1 = \mathcal{O}(\Delta t) \|T_h^n(u) - T_h^n(v)\|_1 + (1 + \mathcal{O}(\Delta t)) \|u^{n-1} - v^{n-1}\|_1 .$$

With a suitable constant and by iteration, find

$$\begin{aligned} \|u^n - v^n\|_1 &\leq (1 + c\Delta t)^n \|T_h^0(u) - T_h^0(v)\|_1 \\ &\quad + \sum_{l=1}^n (1 + c\Delta t)^{n-l} c\Delta t \|T_h^l(u) - T_h^l(v)\|_1 \\ &\leq e^{cn\Delta t} (1 + cn\Delta t) \max_{l=0,\dots,n} \|T_h^l(u) - T_h^l(v)\|_1 , \quad \forall n\Delta t \leq T . \end{aligned}$$

Finally, stability follows

$$\|u - v\|_{1,\infty} \leq (1 + cT) e^{cT} \|T_h(u) - T_h(v)\|_{1,\infty} .$$

□

6.10 Conclusions

It is well-known that monotone and consistent schemes are E-schemes. In turn, consistent E-schemes are monotone provided that the time step is small enough (corollary 6.4.3 and lemma 6.5.1). While it can be difficult to verify monotonicity in the case of non-linear diffusion, the E-property is a useful design principle. For example, numerical schemes featuring enough effective diffusion are E-schemes (lemma 6.3.1). In the presence of natural diffusion, artificial diffusion may be reduced leading to less diffusive and more accurate approximations (corollary 6.3.2). On finite meshes $\Delta x > 0$ adaptive E-schemes are less diffusive and more accurate than standard methods. On fine meshes and in the limit $\Delta x \rightarrow 0$ they give the same results. In general, E-schemes when applied to balance laws are both monotone, TVD and stable (lemmas 6.9.1 and 6.9.2 and corollary 6.6.4). Explicit E-schemes are stable and, under the more restrictive condition eq. (6.24), also monotone.

Acknowledgments

The authors acknowledge funding by the Ethiopian Department of Education, and the Norwegian Agency for Development Cooperation.

Bibliography

- [1] E. Bohl, Finite Modelle gewöhnlicher Randwertaufgaben, Teubner, Stuttgart, 1981.
- [2] [M. G. Crandall and A. Majda, Monotone difference approximations for scalar conservation laws, *Math. Comput.*, 34 (1980), pp. 1–21.]
- [3] M. Fiedler and V. Pták, On matrices with non-positive off-diagonal elements and positive principal minors, *Czech. Math. J.*, 43 (1962), pp. 382–400.
- [4] B. Gustafsson, H.-O. Kreiss, and A. Sundström, Stability theory of difference approximations for mixed initial boundary value problems II, *Math. Comput.*, 26 (1972), pp. 649–689.
- [5] A. Harten, High resolution schemes for hyperbolic conservation laws, *J. Comp. Math. Phys.*, 49 (1983), pp. 357–393.
- [6] H.-O. Kreiss, Initial boundary value problems for hyperbolic partial differential equations., Proc. ICM Vancouver, (1974), pp. 127–137.
- [7] H. O. Kreiss and J. Lorenz, Initial-Boundary Value Problems and the Navier–Stokes Equations, Academic Press, Boston, 1989.
- [8] D. Kröner, Numerical Schemes for Conservation Laws, Wiley, Teubner, Stuttgart, 1997.
- [9] S. Osher, Riemann solvers, the entropy condition, and difference approximations, *SIAM J. Numer. Anal.*, 21 (1984), pp. 217–235.
- [10] V. V. Rusanov, Calculation of interaction of nonsteady shock waves with obstacles, *J. Comp. Math. Phys. USSR*, 1 (1961), pp. 267–279.
- [11] E. Tadmor, Numerical viscosity and the entropy condition for conservative difference schemes, *Math. Comput.*, 43 (1984), pp. 369–381.

An adaptive E-scheme for conservation laws

Ebise Adugna Abdi, Christian Valdemar Hansen and Hans Joachim Schroll

Abstract

An adaptive E-scheme for possibly degenerate, viscous conservation laws is presented. The scheme makes use of both given and numerical diffusion to establish the E-property. In the degenerate case it reduces to local Lax-Friedrichs. Both explicit and time-implicit E-schemes are monotone. Numerical experiments demonstrate the robustness and improved accuracy of the adaptive scheme.

7.1 Introduction

Numerical methods for hyperbolic conservation laws $u_t + f(u)_x = 0$ require artificial viscosity to be stable. Tadmor [6] showed that E-schemes [4], that are schemes featuring no less numerical viscosity than Godunov's scheme, are entropy-stable. In this paper, we outline the construction of an adaptive scheme for possibly degenerate viscous conservation laws

$$u_t + f(u)_x = (du_x)_x , \quad d = d(x) \geq 0$$

based on the observation that enough effective (natural and artificial) diffusion

$$2D = 2(d + \epsilon\Delta x) \geq |f'| \Delta x$$

leads to E-schemes. While it is well known that monotone schemes [2] are E-schemes, we prove in [1] that consistent E-schemes are monotone. Thus our adaptive viscosity scheme is entropy stable and monotone. The latter would be difficult to verify directly as the local diffusion coefficient depends on the solution via the characteristic speed $f'(u)$.

Numerical experiments with scalar conservation laws and a degenerate Navier-Stokes system confirm that the adaptive E-scheme is stable but more accurate than traditional schemes designed for hyperbolic problems and augmented by the given, natural diffusion.

7.2 Diffusion leads to E-schemes

As a starting point let us consider the classical Lax-Friedrichs scheme in conservation form

$$u_j^{n+1} = u_j^n - \frac{\Delta t}{\Delta x} \left(F(u_j^n, u_{j+1}^n) - F(u_{j-1}^n, u_j^n) \right) \quad (7.1)$$

with numerical flux

$$F(u_j, u_{j+1}) = \frac{1}{2} (f(u_j) + f(u_{j+1})) - D_{j+1/2} \frac{u_{j+1} - u_j}{\Delta x} \quad (7.2)$$

and numerical diffusion $2\epsilon = \Delta x / \Delta t \geq \|f'\|_\infty$ augmented by the natural, given diffusion d

$$D_{j+1/2} = d_{j+1/2} + \epsilon \Delta x .$$

Expanding both fluxes $f(v)$ and $f(w)$ around u between $v \leq u \leq w$, we find

$$\frac{1}{2} (f(w) + f(v)) = f(u) + \frac{1}{2} f'(\xi)(w - u) - \frac{1}{2} f'(\eta)(u - v) ,$$

with some $\xi \in [u, w]$ and $\eta \in [v, u]$. When the diffusion in relation to the step size is large enough

$$2D \geq |f'| \Delta x , \quad (7.3)$$

the numerical flux (7.2) cannot exceed the given, natural flux

$$F(v, w) = \frac{1}{2} (f(w) + f(v)) - \frac{D}{\Delta x} (w - v) \leq f(u) .$$

Presumably, this observation led Osher to define E-schemes as schemes in conservation form (7.1) with so-called E-fluxes

$$\text{sign}(w - v) (F(v, w) - f(u)) \leq 0 , \quad \forall u \in \overline{v, w} . \quad (7.4)$$

Lemma 3.1 in [1] states that locally enough diffusion

$$2D_{j+1/2} \geq |f'|_{j+1/2} \Delta x , \quad |f'|_{j+1/2} = \max_u |f'(u)| , \quad \forall u \in \overline{u_j, u_{j+1}}$$

guarantees E-fluxes and E-schemes. The local Lax–Friedrichs or Rusanov–scheme [5] realizes the condition through numerical diffusion alone $2\epsilon_{j+1/2} = |f'|_{j+1/2}$. In this contribution we suggest to exploit any available natural diffusion

$$2\epsilon_{j+1/2} \Delta x = (\Delta x |f'|_{j+1/2} - 2d_{j+1/2})^+ \quad (7.5)$$

to avoid unnecessary overdamaging. Throughout this article the superscript + denotes the positive part $(x)^+ = \max(x, 0)$. Note that the adaptive viscosity scheme (7.1) (7.2) with effective diffusion $D_{j+1/2} = d_{j+1/2} + \epsilon_{j+1/2} \Delta x$ and adaptive numerical diffusion according to (7.5) is by construction an E-scheme and as such entropy stable.

7.3 Less diffusion gives better resolution

To compare the adaptive viscosity scheme vs local Lax–Friedrichs consider Burgers' equation

$$u_t + \frac{1}{2} (u^2)_x = (d(x)u_x)_x , \quad x \in (-6, 6]$$

augmented with degenerate diffusion $d(x) = (0.025 \sin(\pi x/6))^+$. Boundary conditions are periodic, and the initial data is the piecewise constant characteristic function $\chi_{[-5, -3] \cup [1, 3]}$. On an equidistant mesh $\Delta x = 0.05$ and at time $t = 3.0$, approximations obtained by local Lax–Friedrichs and the adaptive E-scheme are shown in Fig. 7.1.

It is obvious that for negative x and without natural diffusion both schemes coincide. On the right half line, however, the local Lax–Friedrich's numerical diffusion (dashed red) grows with $f'(u) = u$ and, on top, is almost as big as the given diffusion (dashed green). The adaptive E-scheme in contrast requires only very little artificial diffusion. Accordingly, the approximation of the latter scheme (solid blue) appears sharper especially at its peak near $x = 4$.

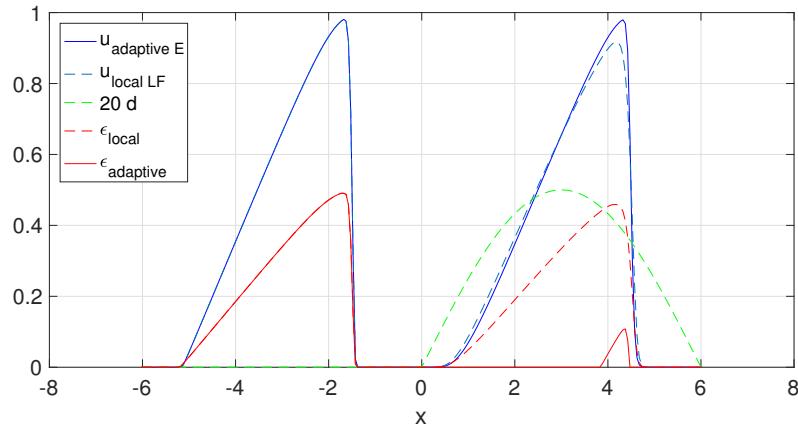


Figure 7.1: Approximations at time $t = 3$ and upscaled diffusion coefficients for Burgers' equation. Note that $D = d + \epsilon\Delta x$ and $\Delta x = 1/20$.

Let's consider next the scalar conservation law in two variables

$$u_t + \nabla \cdot f(u) = \nabla \cdot (d(x)\nabla u) , \quad f(u) = \frac{1}{2} \left(u^2, u^4 \right)^T$$

with degenerate diffusion depending on x alone

$$d(x) = \frac{1}{32} \left(\frac{1}{2} + \sin \left(\frac{\pi(x-6)}{12} \right) \right)^+$$

and periodic boundary- and initial conditions

$$u(0, x, y) = \sin \left(\frac{(x-6)\pi}{6} \right) \sin \left(\frac{(y+3)\pi}{6} \right) .$$

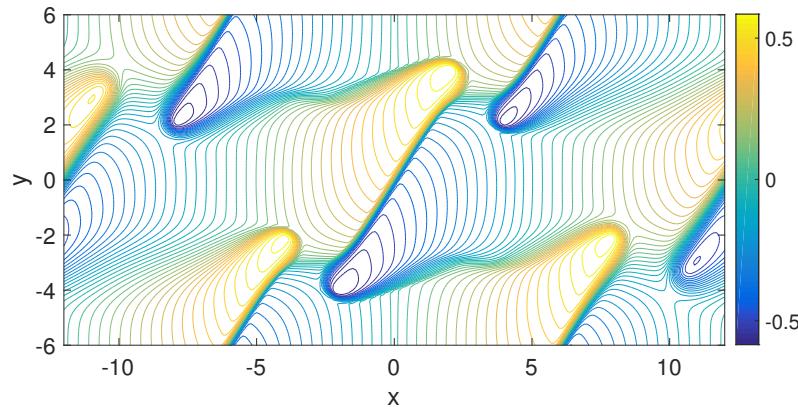


Figure 7.2: Approximation by the adaptive E-scheme at $t = 10$.

The mesh size in both directions is $\Delta x = \Delta y = 0.1$ and the one dimensional schemes are applied coordinate-by-coordinate; one step in x -direction followed by one step in y -direction. By this approach the CFL limit is 0.5. The approximation by the adaptive E-scheme is depicted in Fig. 7.2. Due to excess diffusion and overdamping in the local Lax–Friedrichs scheme, especially near corners of propagating fronts, one can observe large local deviations compared to the adaptive E-scheme, see Fig. 7.3.

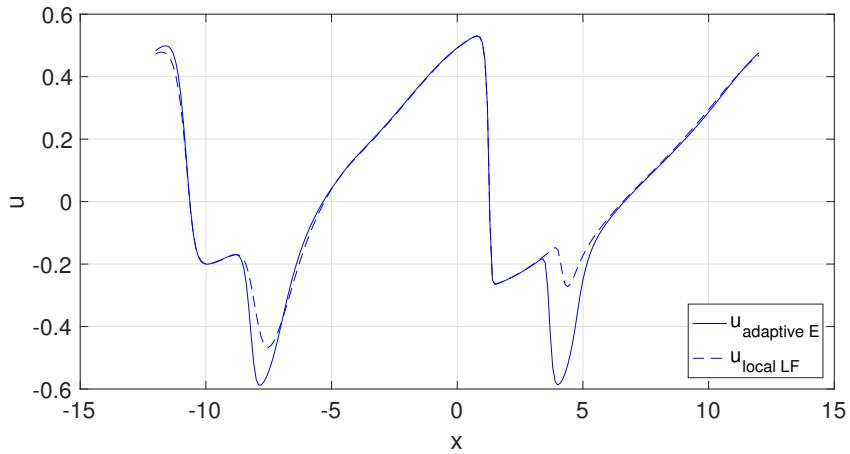


Figure 7.3: Local Lax–Friedrichs vs adaptive E approximations at $y = 2$ and $t = 10$.

7.4 E-schemes are monotone

The explicit scheme (7.1) is monotone in the sense of Crandall and Majda [2]

$$u_j^{n+1} = u_j^n - \frac{\Delta t}{\Delta x} (F(u_j^n, u_{j+1}^n) - F(u_{j-1}^n, u_j^n)) = \mathcal{H}(u^n)_j$$

if \mathcal{H} is a non-decreasing function in all its arguments. As already pointed out by Osher [4] monotone schemes are E-schemes. In turn, consistent E-schemes with CFL condition $\frac{\Delta t}{\Delta x} \|\partial_v F - \partial_w F\|_\infty \leq 1$ are monotone, see [1] Lemma 4.1 and 4.2.

Consider for example the case $u_j < u_{j+1}$. By the E-property (7.4)

$$F(u_j, u_{j+1}) \leq \begin{cases} f(u_j) = F(u_j, u_j) \\ f(u_{j+1}) = F(u_{j+1}, u_{j+1}) \end{cases} .$$

Hence, $F(v, w)$ is non-decreasing in v and non-increasing in w . The diagonal element in $D\mathcal{H}(u)$ is $1 - \Delta t/\Delta x(\partial_v F - \partial_w F)$ and hence the CFL condition controls monotonicity of $\mathcal{H}(u)_j$ with respect to u_j .

This shows that E-schemes cannot extend the class of monotone schemes and thus are limited to first order of accuracy.

Via the calculus of inverse-monotone matrices, it follows that implicit time stepping

$$\frac{1}{\Delta t} (u^{n+1} - u^n) = \vartheta F_{\Delta x}(u^{n+1}) + (1 - \vartheta) F_{\Delta x}(u^n) , \quad 0 \leq \vartheta \leq 1$$

for the conservative space discretization

$$F_{\Delta x}(u)_j = -\frac{\Delta t}{\Delta x} (F(u_j, u_{j+1}) - F(u_{j-1}, u_j))$$

with a consistent E-flux (7.4) is monotone under the CFL condition

$$(1 - \vartheta) \frac{\Delta t}{\Delta x} \|\partial_v F - \partial_w F\|_\infty \leq 1 .$$

See [1], Lemma 5.1 for details.

7.5 Application to compressible Navier–Stokes equations

Most relevant are adaptive viscosity E-schemes for incomplete and degenerate parabolic systems like the compressible Navier–Stokes equations [3]. In the isentropic case without thermodynamic effects, the equations consist of conservation of mass $\rho_t + \nabla \cdot (\rho \mathbf{u}) = 0$, conservation of momentum $\rho \mathbf{D}\mathbf{u}/Dt + \nabla p = \mathbf{S}$ and an equation of state, for example the gamma-law $p = \rho^\gamma$, $\gamma = 1.4$. Here \mathbf{D}/Dt denotes the material derivative. The viscous stress tensor is given by $\mathbf{S} = \nabla((\mu + \lambda)\nabla \cdot \mathbf{u}) + \nabla \cdot (\mu \nabla \mathbf{u})$.

In conservative form and one space variable, the system reads

$$\begin{pmatrix} \rho \\ \rho u \\ \rho v \end{pmatrix}_t + \begin{pmatrix} \rho u \\ \rho u^2 + p \\ \rho u v \end{pmatrix}_x = \begin{pmatrix} 0 \\ (2\mu + \lambda)u_x \\ \rho v \end{pmatrix}_x . \quad (7.6)$$

In two space dimensions, we have

$$\begin{aligned} & \begin{pmatrix} \rho \\ \rho u \\ \rho v \end{pmatrix}_t + \begin{pmatrix} \rho u \\ \rho u^2 + p \\ \rho u v \end{pmatrix}_x + \begin{pmatrix} \rho v \\ \rho u v \\ \rho v^2 + p \end{pmatrix}_y = \\ & \begin{pmatrix} 0 \\ (2\mu + \lambda)u_x + (\mu + \lambda)v_y \\ \mu v_x \end{pmatrix}_x + \begin{pmatrix} 0 \\ \mu u_y \\ (\mu + \lambda)u_x + (2\mu + \lambda)v_y \end{pmatrix}_y . \end{aligned} \quad (7.7)$$

Here ρ , ρu and ρv denote mass- and momentum densities respectively.

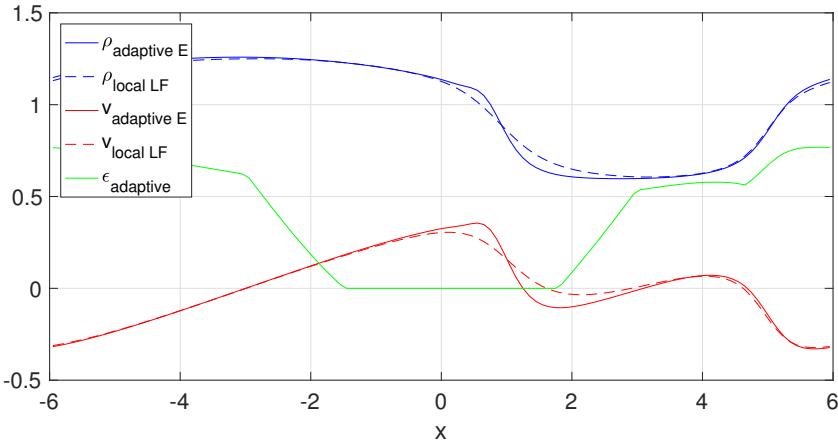


Figure 7.4: Approximations to Navier–Stokes system (7.6) at $t = 6$.

Fig. 7.4 shows the adaptive E–scheme when applied to the one–dimensional system (7.6) with given diffusion $2\mu + \lambda = (0.1 \sin((x+3)\pi/6))^+$. The initial density is $\rho(0, x) = 1.0 + 0.5 \sin(x\pi/6)$ with zero initial momentum and periodic boundary conditions. The mesh size Δx is 0.1 with CFL number 0.9. Due to zero artificial diffusion around the origin, the approximation by the adaptive E–scheme appears much sharper than the local Lax–Friedrichs scheme that suffers from unnecessary numerical diffusion and overdamping.

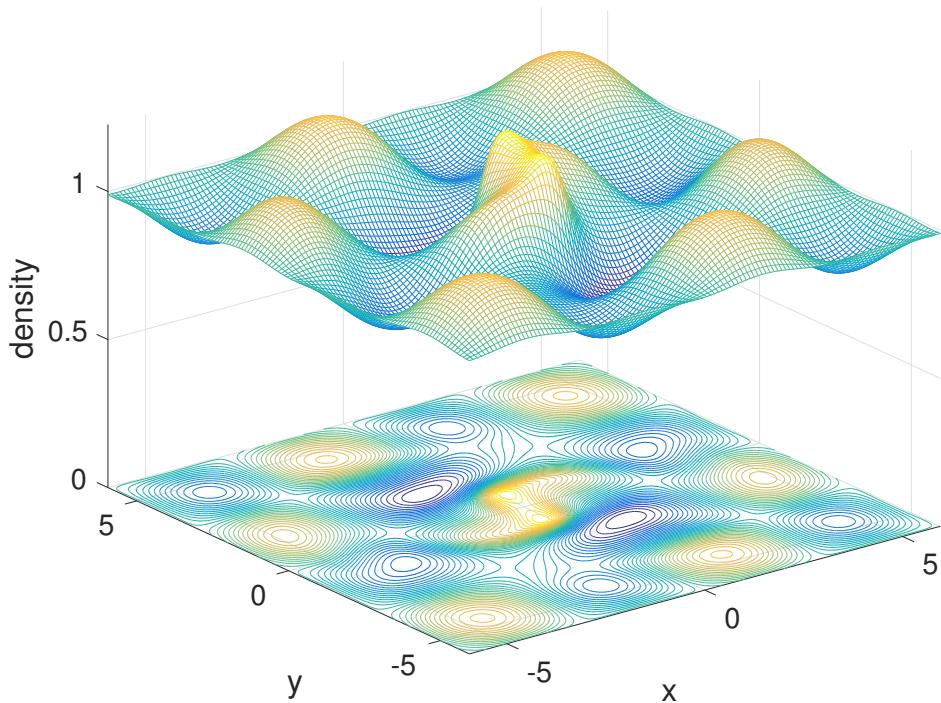
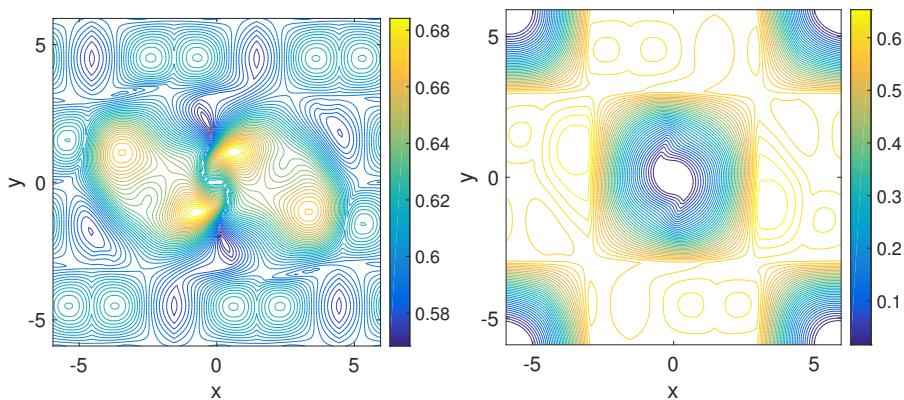
In the last experiment, the new scheme is applied to the two–dimensional system (7.7) with degenerate diffusion

$$\mu = (0.05 \sin((x-3)\pi/6) \sin((y-3)\pi/6))^+ .$$

For simplicity, we set $\lambda = -2\mu/3$. Initially, the flow is at rest and the density is a perturbed sin wave $\rho(0, x, y) = 1 + 0.2 \sin((x-6)\pi/3) \sin((y+3)\pi/3)$ with $\rho(0, x, y) = 0.5$ inside the rectangle $[-2, 2] \times [-1, 1]$.

The mesh size in both x and y –direction is 0.1 and the CFL number is 0.4. The density at later time is depicted in Fig. 7.5. Despite low and locally degenerate artificial viscosities, the scheme is stable and produces approximations free of spurious oscillations.

The artificial viscosity is adapted componentwise. As there is no natural diffusion in the continuity equation, the numerical viscosity for ρ is strictly positive, as seen in Fig. 7.6, left panel. Local and adaptive viscosities are identical in their first components. For the momentum, however, the adaptive scheme makes use of the given viscosity. Where enough natural viscosity is available the artificial viscosity degenerates to zero, see Fig. 7.6, right panel.

Figure 7.5: Density in (7.7) at $t = 2$.Figure 7.6: Numerical viscosities at $t = 2$ in density ρ (left) and momentum ρu (right).

7.6 Concluding remarks

The concept of E-schemes does not extend the class of first order, monotone schemes. However, it can be much easier to verify the E-condition than

to show monotonicity directly; enough effective diffusion automatically leads to E-schemes. Adaptive viscosity E-schemes are robust and less diffusive than standard methods when applied to degenerate, viscous conservation laws. They are ideal building blocks in high resolution finite volume methods.

Acknowledgments

The authors acknowledge funding by the Ethiopian Department of Education, and the Norwegian Agency for Development Cooperation.

Bibliography

- [1] E. A. ABDI, C.V. HANSEN AND H.J. SCHROLL, *An adaptive viscosity E-scheme for balance laws*, preprint SDU (2017), submitted.
- [2] M.G. CRANDALL AND A. MAJDA, *Monotone Difference Approximations for Scalar Conservation Laws* Math. Comput. **34** 149 (1980), 1–21.
- [3] H.O. KREISS AND J. LORENZ, *Initial-Boundary Value Problems and the Navier-Stokes Equations* Academic Press, Boston, 1989.
- [4] S. OSHER, *Riemann solvers, the entropy condition, and difference approximations*, SIAM J. Numer. Anal. **21** (1984), 217–235.
- [5] V.V. RUSANOV *Calculation of interaction of nonsteady shock waves with obstacles* J. Comp. Math. Phys. USSR **1** (1961), 267–279
- [6] E. TADMOR, *Numerical viscosity and the entropy condition for conservative difference schemes*, Math. Comput. **43** (1984), 369–381.

Part IV

Computational design of settlement layouts

Simulation of flow and view with applications in computational design of settlement layouts

Christian Valdemar Hansen, Anders Logg, Carl Lundholm

Abstract

We present methodology, algorithms and software for evaluating flow and view for architectural settlement layouts. For a given settlement layout consisting of a number of buildings arbitrarily positioned on a piece of land, in the present study an island situated on the west coast of Sweden, the methodology allows for evaluation of flow patterns and for evaluating the view experienced from the buildings. The computation of flow is based on a multimesh finite element methods, which allows each building to be embedded in a boundary-fitted mesh which can be moved around freely in a fixed background mesh. The computation of view is based on a novel and objective measure of the view which can be efficiently computed by rasterization.

8.1 Introduction

When designing settlement layouts, architects need to take a large number of variables into consideration, such as economic interests and connections to infrastructure (roads, water and electricity), the experienced quality of view from the buildings, wind conditions, and many more. In this study, we examine how to efficiently compute wind patterns and how to evaluate view. The aim is to provide architects with a computational tool that can be used as a guide as part of an iterative architectural design process. For a current example of a challenging design problem in architectural settlement layout, see [18]. There are several examples of CFD simulations in urban environments in the literature; see, e.g., [1, 4, 5, 15]. A recent study by [17] investigates the simulation of flow on geometries directly defined by point clouds.

A central issue when designing a computational tool for settlement layout design is that the tool should be able to quickly evaluate a multitude of suggested designs, either as part of an optimization loop or as part of a manual (artistic) iterative design process. Standard numerical methods for computation of flow require that a computational mesh or grid is generated around both buildings, ground and other objects. Generating such a body-fitted mesh is a costly procedure and even more so when a large number of different meshes must be created, one for each configuration of the buildings.

Instead, we examine in this work the use of *multimesh* finite element methods. Multimesh finite element methods allow a problem to be posed not on a single body-fitted mesh but instead on a collection of meshes that may overlap arbitrarily and which together define the computational domain. Figure 8.1 gives an example of such a configuration.

By allowing the computational domain to be discretized by multiple overlapping meshes, one may freely move the overlapping meshes relative to one another, which allows a multitude of configurations to be computed and evaluated without the need for costly mesh generation. However, this flexibility comes at a price. First, one must ensure that the finite element discretization remains stable and that convergence is retained independently of the relative positions of the meshes. If the finite element method is not carefully designed and does not incorporate the correct stabilization terms, certain configurations may lead to very ill-conditioned systems, low accuracy and even blow-up. Another concern is that the formulation of multimesh finite element methods requires integration over cut cells and interfaces. This means that an implementation faces challenging problems in computational geometry, when intersections and quadrature points must be computed efficiently and robustly.

For the computation of view, we develop in this work a novel and objective measure that can be used to quantify the view for any given design. This measure may be computed efficiently using a technique from computer graphics known as *rasterization*. The measure itself allows for easy incorporation of

weights that may be tweaked to give different weights to view of air, water, ground, buildings, herbage and other objects.

In the remainder of this paper, we first review in Section 8.2 multimesh finite element methods and their relation to existing methodologies for discretization of multiphysics problems and complex geometries. We then discuss in Sections 8.3 and 8.4 details of the algorithms and the implementation. Results are presented in Section 8.5 and finally we present our conclusions and discuss current limitations and future work in Section 8.6.

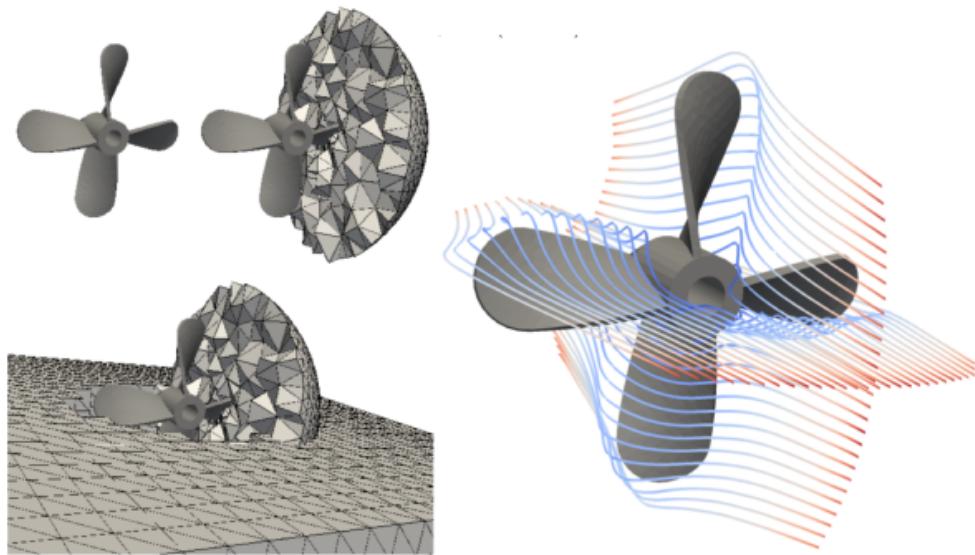


Figure 8.1: A collection of arbitrarily overlapping meshes together defining the computational domain, here the surroundings of a 3D propeller immersed in water. Images courtesy of [19].

8.2 Multimesh finite element methods

A number of methodologies have been proposed to circumvent the limitations of interface-fitted discretization. Notable examples are the fictitious domain (FD) method by [11] and the extended finite element method (XFEM) by Belytschko et al. [24]. Both methods have been successful in extending the range of problems that can be simulated, but both suffer from limitations in that the conditioning of the discretization cannot be guaranteed, and a theoretical framework for convergence analysis and error estimation is lacking. In particular, time-dependent multiphysics problems on evolving geometries are typically discretized using *ad hoc* low order discretization methods, which cannot easily be analyzed, nor extended to higher order.

Over the past decade, a theoretical foundation for the formulation of stabilized cut FEM has been developed by extending the ideas of [25] to a general weak formulation of the interface conditions, thereby removing the need for interface-fitted meshes. The foundations of cut FEM were presented in [12, 13] and the methodology has since been extended to a number of important multiphysics problems; see [7, 8, 2, 22].

We refer to multimesh finite element methods as finite element methods based on the stabilized finite element formulations of cut FEM in combination with a flexible and general treatment of multiple and arbitrarily overlapping meshes. In addition to the theoretical foundation in cut FEM, the implementation and application of multimesh finite element methods relies on efficient and robust computation of mesh-mesh intersections in 3D. This puts high demands on the implementation and requires much more sophisticated algorithms from computational geometry than what is normally the case for finite element problems. Figure 8.2 illustrates a collection of body-fitted meshes that move independently through a fixed background mesh to discretize the flow around a collection of moving bodies immersed in a fluid.

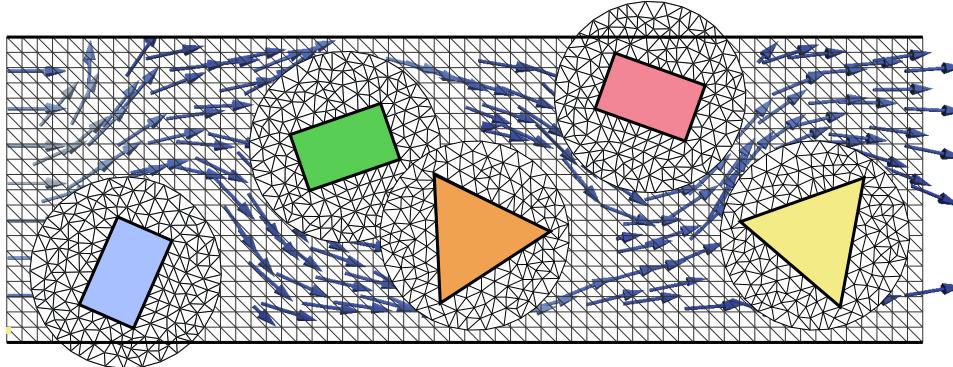


Figure 8.2: Multimesh discretization of the flow around a collection of bodies moving through a channel.

Support for the formulation of multimesh finite element methods has recently been added to the popular open-source finite element package FEniCS [21, 20]. With version 2016.1 of FEniCS, users may formulate and automatically discretize basic multimesh finite element formulations of systems of PDE such as the Stokes problem. The implementation in FEniCS relies on a novel implementation of mesh-mesh intersections, based on efficient generation and traversal of axis-aligned bounding box trees (AABB trees), robust low-level operations for computing and representing the intersections of triangles (2D) and tetrahedra (3D), and generation of quadrature points on cut cells. The implementation is integrated with the automatic code generation of FEniCS which allows multimesh discretizations to be formulated in (close to) natural mathematical language, as will be demonstrated in Section 8.4.

The present study constitutes the first application of the newly implemented multimesh framework of FEniCS.

For the flow problems studied in the present work (modeled by the Stokes equations), the finite element formulation is based on the stabilized Stokes discretization analyzed by [19] as an extension to higher-order function spaces of the discretization previously analyzed by [23].

8.3 Algorithms

We here present an overview of the methodology used to simulate the flow of air and the view for the application under consideration: the design of a settlement (placement of houses) on a small island on the west coast of Sweden.

8.3.1 Computation of flow

To model the flow over the island and houses, a finite element method for Stokes equations on overlapping meshes is used. The method is a slight modification of the cut finite element method for Stokes equations, presented by [19]. For a bounded domain $\Omega \subset \mathbb{R}^d$ with boundary $\partial\Omega$, the strong problem formulation for Stokes equations reads: Find the velocity $\mathbf{u} : \Omega \rightarrow \mathbb{R}^d$ and the pressure $p : \Omega \rightarrow \mathbb{R}$ such that

$$\begin{cases} -\Delta \mathbf{u} + \nabla p = \mathbf{f} & \text{in } \Omega, \\ \nabla \cdot \mathbf{u} = 0 & \text{in } \Omega, \\ \mathbf{u} = \mathbf{0} & \text{on } \partial\Omega, \end{cases} \quad (8.1)$$

where $\mathbf{f} : \Omega \rightarrow \mathbb{R}^d$ is a given right-hand side. To obtain a finite element formulation for Stokes equations on two overlapping meshes, we start by considering two separate bounded domains $\widehat{\Omega}_0$ and $\widehat{\Omega}_1$, called predomains. We may place $\widehat{\Omega}_1$ on top of $\widehat{\Omega}_0$. Let $\Omega_0 := \widehat{\Omega}_0 \setminus \widehat{\Omega}_1$ and $\Omega_1 := \widehat{\Omega}_1$, with boundaries $\partial\Omega_0$ and $\partial\Omega_1$, respectively. We define the solution domain by $\Omega := \Omega_0 \cup \Omega_1$ and the joint boundary between Ω_0 and Ω_1 by $\Gamma := \partial\Omega_0 \cap \partial\Omega_1$. Proceeding in the usual way, by introducing suitable function spaces, multiplying the equations with test functions, and integrating by parts, one arrives at the variational formulation of the problem. To get the corresponding finite element formulation, the predomains, $\widehat{\Omega}_0$ and $\widehat{\Omega}_1$, are tessellated to create the meshes $\widehat{\mathcal{K}}_{h,0}$ and $\widehat{\mathcal{K}}_{h,1}$, respectively. We call $\widehat{\mathcal{K}}_{h,0}$ the background mesh, and $\widehat{\mathcal{K}}_{h,1}$ the overlapping mesh. This gives us a mesh hierarchy on which we define a finite element space for the velocity, \mathbf{V}_h , and a finite element space for the pressure, Q_h . Here, Taylor-Hood elements are used to ensure the stability of the solution, i.e., polynomials of degree two for the velocity and polynomials of degree one for the pressure. The finite element formulation for Stokes equations on two overlapping meshes reads: Find $(\mathbf{u}_h, p_h) \in \mathbf{V}_h \times Q_h$ such that

$$\begin{aligned}
& (D\mathbf{u}_h, D\mathbf{v})_{\Omega_0} + (D\mathbf{u}_h, D\mathbf{v})_{\Omega_1} \\
& - (\langle (D\mathbf{u}_h) \cdot \mathbf{n} \rangle, [\mathbf{v}])_\Gamma - ([\mathbf{u}_h], \langle (D\mathbf{v}) \cdot \mathbf{n} \rangle)_\Gamma \\
& + \beta h^{-1} ([\mathbf{u}_h], [\mathbf{v}])_\Gamma + ([D\mathbf{u}_h], [D\mathbf{v}])_{\Omega_{h,0} \cap \Omega_1} \\
& - (\nabla \cdot \mathbf{u}_h, q)_{\Omega_0} - (\nabla \cdot \mathbf{u}_h, q)_{\Omega_1} + ([\mathbf{n} \cdot \mathbf{u}_h], \langle q \rangle)_\Gamma \\
& - (\nabla \cdot \mathbf{v}, p_h)_{\Omega_0} - (\nabla \cdot \mathbf{v}, p_h)_{\Omega_1} + ([\mathbf{n} \cdot \mathbf{v}], \langle p_h \rangle)_\Gamma \\
& + \gamma([p_h], [q])_{\Omega_{h,0} \cap \Omega_1} + h^2(\Delta \mathbf{u}_h - \nabla p_h, \Delta \mathbf{v} + \nabla q)_{\Omega_{h,0} \setminus \omega_{h,0}} \\
= & (\mathbf{f}, \mathbf{v})_{\Omega_0} + (\mathbf{f}, \mathbf{v})_{\Omega_1} - h^2(\mathbf{f}, \Delta \mathbf{v} + \nabla q)_{\Omega_{h,0} \setminus \omega_{h,0}}, \tag{8.2}
\end{aligned}$$

for all $(\mathbf{v}, q) \in \mathbf{V}_h \times Q_h$. Here, \mathbf{n} is the unit normal to Γ exterior to Ω_1 , $\langle v \rangle = (v_0 + v_1)/2$ is the average of v on Γ (v_i is the limit of v on Ω_i as we approach Γ , for $i = 0, 1$), $[v] = v_1 - v_0$ is the jump in v on Γ , $\Omega_{h,0} \cap \Omega_1$ is the *overlapped* part of all the background cells that are cut by Γ , $\Omega_{h,0} \setminus \omega_{h,0}$ is the *visible* part of all the background cells that are cut by Γ , h is the mesh size, and β and γ are stability parameters. For more details on the method, see [19].

The stability term $\gamma([p_h], [q])_{\Omega_{h,0} \cap \Omega_1}$ has been added to the original formulation of the method presented in [19], and is the slight modification mentioned earlier. Numerical tests have indicated that the presence of this stability term increases the robustness of the method, in particular for the pressure. This indicates that further analysis of the original formulation may be of interest.

For the application studied in this work, the background mesh is also referred to as the air mesh, since it discretizes the air above the island. The overlapping meshes are also referred to as house meshes, since they contain the houses. We are now ready to formulate the algorithm for obtaining the flow over the island and houses by solving (8.2).

Algorithm 1 Flow model

- 1: Geometries for the island and houses are imported. See Figure 8.3.
 - 2: Meshes are generated around the geometries. For the house meshes, cells are also generated *inside* the houses. See Figures 8.4 and 8.5.
 - 3: The house meshes are placed inside the air mesh. See Figures 8.6- 8.8.
 - 4: Boundary conditions (inlet, outlet and no-slip) are set on the air mesh.
 - 5: A no-slip boundary condition is applied to all facets belonging to house mesh cells that are not entirely inside the air mesh and also to facets belonging to house mesh cells that are inside the house. For more details see Figure 8.9.
 - 6: The linear system of equations, resulting from (8.2), is solved.
-

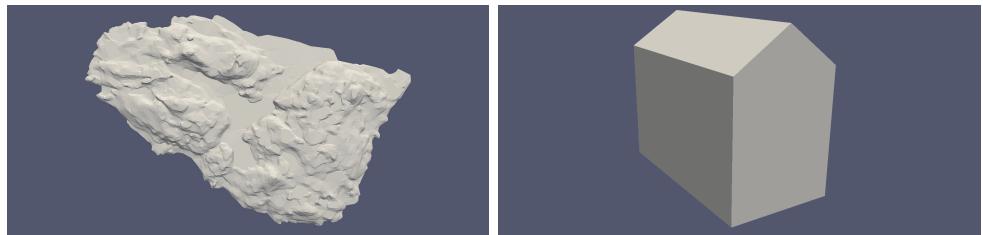


Figure 8.3: *Left:* Geometry of the island. *Right:* Geometry of a simplified house.

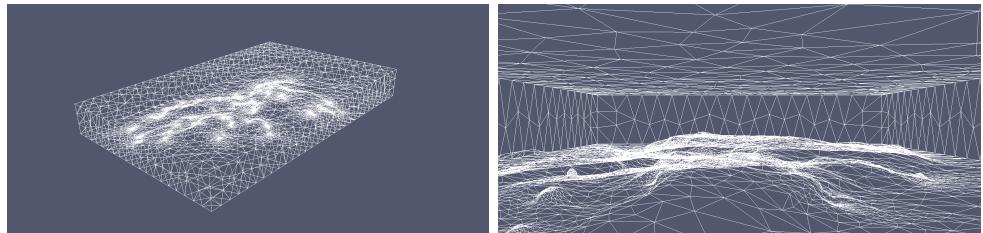


Figure 8.4: Air mesh viewed from the outside and from the inside.

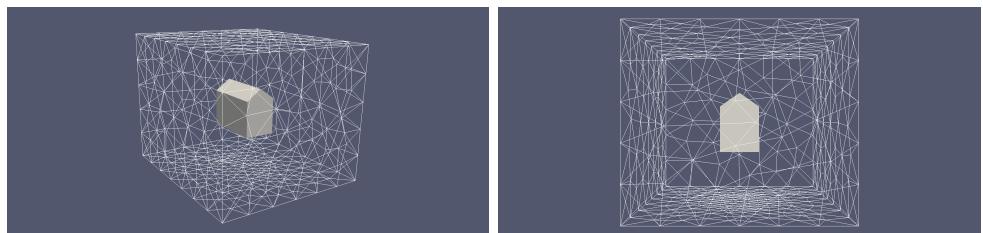


Figure 8.5: House mesh viewed from two different angles.

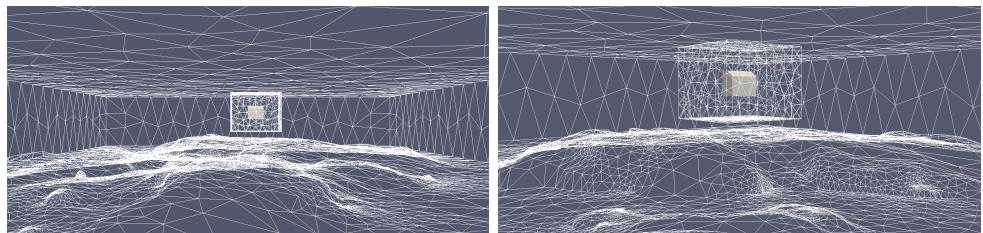


Figure 8.6: House mesh inside air mesh viewed from two different angles.

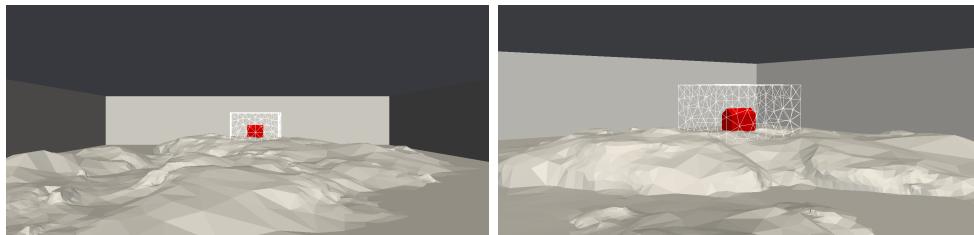


Figure 8.7: House mesh placed on the surface of the island viewed from two different angles.

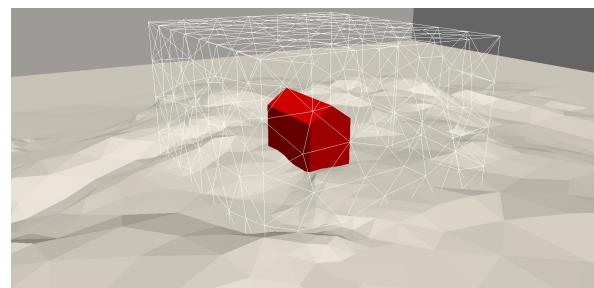


Figure 8.8: Closeup on the placed house mesh.

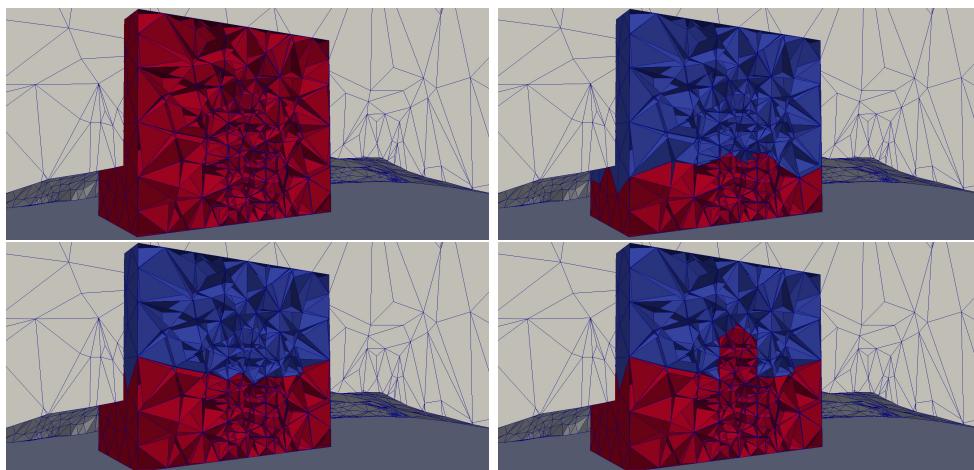


Figure 8.9: Cross sections of a house mesh with a cross section of the air mesh added for reference. Red facets are marked for no-slip, blue facets are not. *Top left:* Initially all facets are marked for no-slip. *Top right:* First facets intersected by the air mesh are removed. *Bottom left:* Then facets cut by the boundary of the air mesh are added. *Bottom right:* Finally facets of the house are added.

8.3.2 Computation of view

We here present a novel and objective measure for evaluating the view from a location such as the window of a building. The measure assigns a value V between zero and one, zero being the worst possible view and one the best possible view.

It is interesting to first consider the best and the worst cases. In the present case, the worst case would occur if the view is nothing else than another house; see illustration in Figure 8.10. Thus in this case the view should evaluate to $V = 0$. The best case $V = 1$ is a view consisting entirely of sea and sky; see Figure 8.11.



Examining again the worst case, the distance to a neighbouring house should influence the value of the view. In general, the negative impact of objects on the view should decrease by the distance. Our proposed measure of view is expressed as an integral over the integration domain $\omega = \omega_\phi \times \omega_\theta$ of size $|\omega|$:

$$V = \frac{1}{|\omega|} \iint_{\omega} \sigma(\phi, \theta) d\phi d\theta. \quad (8.3)$$

Here, $\sigma(\phi, \theta)$ is the weight of the object viewed at the angle (ϕ, θ) . This weight must take a value between zero and one. In the present study, we have used the following weights:

$$\sigma(\phi, \theta) = \begin{cases} 1, & \text{if water,} \\ 1, & \text{if sky,} \\ 2 \cdot s(w(\phi, \theta) \frac{l(\phi, \theta)}{L}) - 1, & \text{otherwise,} \end{cases} \quad (8.4)$$

where $s(t)$ is the Sigmoid function

$$s(t) = \frac{1}{1 + e^{-t}}, \quad (8.5)$$

and $l(\phi, \theta)$ is the distance to the nearest object viewed at the angle (ϕ, θ) . The specific element weight $w(\phi, \theta)$ is set to be 0.1 and 0.7 if there is a house or

ground viewed at (ϕ, θ) , respectively. Based on the premise that $\sigma(\phi, \theta) = 0.9$ for a house viewed at (ϕ, θ) and placed at the horizon approximately 5 kilometers away, the constant L is found to be 0.17 km.

Equation (8.3) weights the view independent of the cardinal direction. However, in northern countries like Sweden, a southern view is often weighted higher than northern view because of the sunlight. Thus, to obtain a view formula which can depend on the cardinal direction, we let $\theta \in [0, 2\pi]$ be the angle in the horizontal plane. Let south be at $\theta = 0$, consequently $\theta = \pi$ is the northern direction. Assuming that one would weight the view in the south direction three times as high as the view in the north direction, the cardinal direction weight function could be expressed by

$$D(\theta) = 1 + \frac{1}{2} \sin(\theta - 3\pi/2) . \quad (8.6)$$

For a 360° horizontal view valuation, we now introduce $D(\theta)$ in (8.3). We may thus make the following modification to the measure of view:

$$V_{360} = \frac{1}{|\omega|} \int_0^{2\pi} D(\theta) \int_{\omega_\phi} \sigma(\phi, \theta) d\phi d\theta . \quad (8.7)$$

Note that $V_{360} \in [0, 1]$.

8.4 Implementation

8.4.1 Implementation of flow computation

The flow model has been implemented with the open-source finite element software FEniCS. Simulations of the flow model have been run with FEniCS v2016.1 on a MacBook Pro with operating system OS X Yosemite version 10.10.5, a 3.1 GHz Intel Core i7 processor and 16 GB RAM.

The island geometry has been imported as an STL-file and placed on the bottom of a sufficiently large box. The air mesh has then been generated and stored as an XML-file by the use of functionality provided by the FEniCS-package mshr. The geometries for the simplified houses have been made in Gmsh [10]. The house meshes have also been generated with Gmsh and then converted to XML-files. To demonstrate how the FEniCS multimesh functionality is used to implement the flow model, we present the following code-snippets.

Python code

```
from dolfin import *

# Read the meshes from file
mesh_air = Mesh("mesh_air.xml")
mesh_house1 = Mesh("mesh_house1.xml")
```

```

mesh_house2 = Mesh("mesh_house2.xml")

# Initialize multimesh, add meshes and build
multimesh = Multimesh()
multimesh.add(mesh_air)
multimesh.add(mesh_house1)
multimesh.add(mesh_house2)
multimesh.build()

```

The order in which the meshes are added will create a mesh hierarchy, where the latest added mesh will be on the top. A multimesh may then be used to create finite element function spaces that are used to define the trial and test functions.

Python code

```

# Create function space
P2 = VectorElement("P", tetrahedron, 2)
P1 = FiniteElement("P", tetrahedron, 1)
TH = P2 * P1
W = MultiMeshFunctionSpace(multimesh, TH)

# Define trial and test functions
(u, p) = TrialFunctions(W)
(v, q) = TestFunctions(W)

```

Here, Taylor-Hood elements of degree 2/1 are created. The trial and test functions are used to define the bilinear and linear forms, which are in turn are used to assemble the system matrix and load vector, respectively, for the linear system of equations.

Python code

```

# Define facet normal, mesh size and stability parameters
n = FacetNormal(multimesh)
h = 2.0*Circumradius(multimesh)
beta = Constant(1e1)
gamma = Constant(1e8)

def tensor_jump(v, n):
    return outer(v('+'), n('+')) + outer(v(')'), n(')')

def a_h(v, w):
    return inner(grad(v), grad(w))*dX \
        - inner(avg(grad(v)), tensor_jump(w, n))*dI \
        - inner(avg(grad(w)), tensor_jump(v, n))*dI \
        + beta/avg(h) * inner(jump(v), jump(w))*dI

def b_h(v, q):
    return -div(v)*q*dX + jump(v, n)*avg(q)*dI

def l_h(v, q, f):
    return inner(f, v)*dX

```

```

def s_0(v, w):
    return inner(jump(grad(v)), jump(grad(w)))*d0

def s_C(v, q, w, r):
    return h*h*inner(-div(grad(v)) + grad(q),
                      -div(grad(w)) - grad(r))*dC

def l_C(v, q, f):
    return h*h*inner(f, -div(grad(v)) - grad(q))*dC

def s_P(q, r):
    return gamma * inner(jump(q), jump(r))*d0

# Define bilinear form
a = a_h(u, v) + b_h(v, p) + b_h(u, q) + s_0(u, v) \
    + s_C(u, p, v, q) + s_P(p, q)

# Define linear form
L = l_h(v, q, f) + l_C(v, q, f)

# Assemble linear system
A = assemble_multimesh(a)
b = assemble_multimesh(L)

```

Note the resemblance of the code to the mathematical notation used in the finite element formulation (8.2). Then boundary conditions for the air mesh are created and applied to the system matrix and load vector.

Python code

```

# Mark boundaries for air mesh
facet_markers_air = FacetFunction("size_t", mesh_air)
facet_markers_air.set_all(0)
noslip_boundary.mark(facet_markers_air, 1)
inflow_boundary.mark(facet_markers_air, 2)
outflow_boundary.mark(facet_markers_air, 3)

# Create boundary conditions
bc0 = MultiMeshDirichletBC(V, noslip_value,
                           facet_markers_air, 1, 0)
bc1 = MultiMeshDirichletBC(V, inflow_value,
                           facet_markers_air, 2, 0)
bc2 = MultiMeshDirichletBC(Q, outflow_value,
                           facet_markers_air, 3, 0)

# Apply boundary conditions
bc0.apply(A, b)
bc1.apply(A, b)
bc2.apply(A, b)

```

The no-slip condition for the concerned facets of the house meshes are created and applied in a similar fashion. Finally, the linear system is solved and the solution components on the different meshes are extracted.

Python code

```
# Compute solution
w = MultiMeshFunction(W)
solve(A, w.vector(), b)

# Extract solution components
u_air = w.part(0).sub(0)
p_air = w.part(0).sub(1)
u_house1 = w.part(1).sub(0)
p_house1 = w.part(1).sub(1)
u_house2 = w.part(2).sub(0)
p_house2 = w.part(2).sub(1)
```

8.4.2 Implementation of view computation

Given the mesh constructed for the flow simulation, it is now possible to visualize the view from a given point. In practice we do this by rendering an image from the 3D mesh with the use of the rasterization rendering technique. This technique goes back to [9, 26, 14]. The rasterization algorithm projects the triangles from the 3D mesh onto a 2D image.

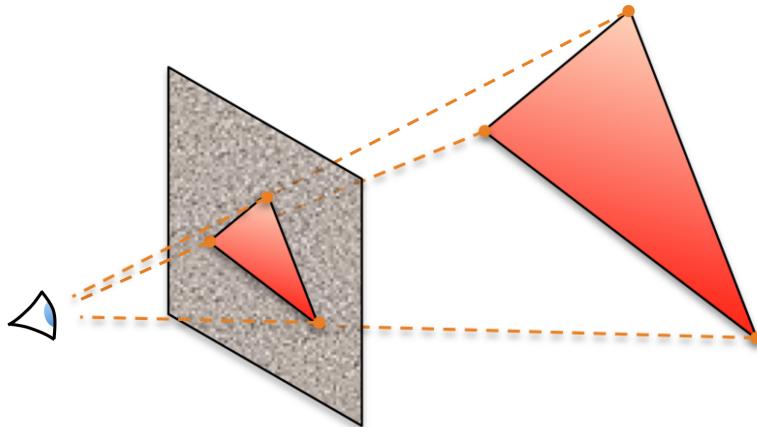


Figure 8.12: Projection of triangles from the 3D mesh onto the 2D image.

The projecting principle is sketched in Figure 8.12, where the vertices which are mapped onto the 2D image plane are used to check which pixels the triangle covers. The naive idea is to loop across all the pixels in the image and check if they are inside the projected triangle or not. The efficiency of this approach depends on the size of the triangles. To account for small triangles, one may optimize the search by only searching the pixels which lie inside the bounding box of the triangle. In Figure 8.13, the blue square around the triangle illustrates the bounding box for which the corner coordinates are rounded to the nearest pixel.

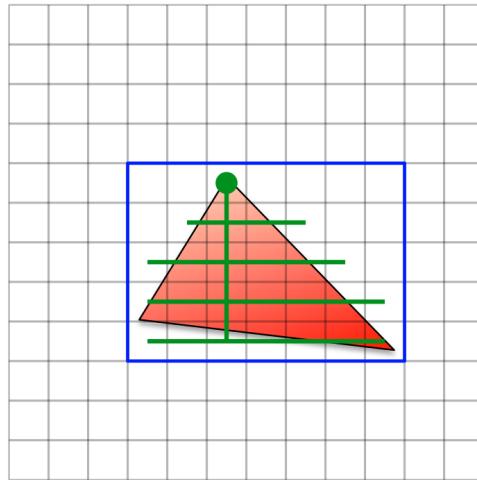


Figure 8.13: To check which pixels that lies inside the triangle, we start from the top-point of the triangle (the green dot) going down one step at a time to check if the pixels to the left and right are contained in the triangle.

This can be optimized further by not checking all the pixels inside the bounding box, but starting in the pixel which contains the top-point of the triangle. Then go stepwise down and check the pixels to the left and right of the reference point. If we already visited one or more pixels which are inside the projected triangle and then come to a pixel which is not in the triangle, the search stops in that direction. The search algorithm is illustrated with the green lines in Figure 8.13. There are several ways to optimize this, see [26] for further reading.

Looping over all the triangles to do the projections one by one, can cause two or more projected triangles to overlap. To decide which one that should be shown in the image we have to look at the distance to them. The distances to the triangles which are already shown in the image are stored in a two dimensional array with the same dimension as the image. Thus only the element with the shortest distance is shown in the image.

The rasterization algorithm is implemented in C++ and by the use of the SWIG interface compiler it is possible to access the rasterization algorithm from the C++ code in a Python script. The rasterization algorithm needs as input a list of FEniCS (.xml) meshes, where the first mesh in the list should be the main island mesh, and the rest house meshes. With the scene set, the algorithm needs to know the size of the image, both the size in pixels and the real size measured in the same units as the meshes. Also the position of the camera and the distance between the camera and the image is needed. The direction for the camera expressed by a vector should also be given. With these inputs, the algorithm generates the image and a matrix \mathbf{S} which has the

same size as the image and contains a value for σ for each pixel. The view valuation V is then found from \mathbf{S} and given as output.

The view V may thus be computed from the image of the view generated by rasterization. If we want to compute the 360° view from (8.7), the domain ω is assumed to be either a cylinder or a sphere. As the rasterization algorithm generates flattened images, (8.7) cannot be used directly. However (8.7) could be estimated by

$$V_{360} = \sum_{i=0}^{N-1} \frac{D(2\pi i/N)}{N} V_i, \quad (8.8)$$

where $N \geq 3$ is the number of images and V_i is the view valuation for image number i . It is important that no images are overlapping and that they, when joined together, form the boundary of a convex polygon with N edges, when seen from above. Thus V_i is evaluated in the direction with angle $2\pi i/N$ with respect to south and an image width of $2d \tan(\pi/N)$, where d is the distance between the camera and the image.

8.5 Results

8.5.1 Flow

Results from two simulations using the multimesh finite element method for Stokes equations on overlapping meshes (8.2) are presented. The first simulation was run with one overlapping house mesh and the results can be seen in Figures 8.14 and 8.15. The other simulation was run with two overlapping house meshes and the results can be seen in Figures 8.16 and 8.18. The results are in the form of tubed streamlines from the velocity field of the finite element solution. Tubes from the solution on the house mesh are made thicker than the ones from the air mesh for illustration purposes. The flow in all the figures goes from left to right.

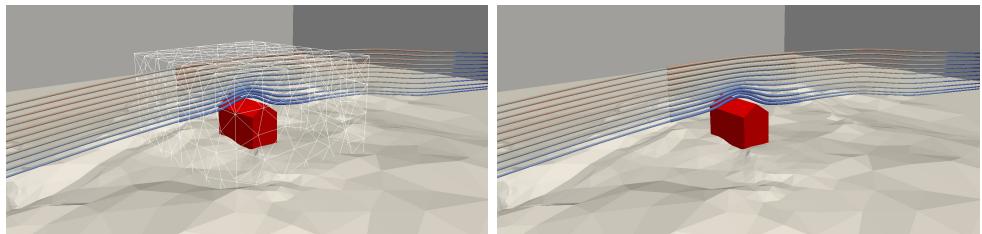


Figure 8.14: Tubed streamlines from the simulation with *one* overlapping house mesh. With and without the house mesh visible. Notice the continuous velocity field going between the two different meshes used for discretization.

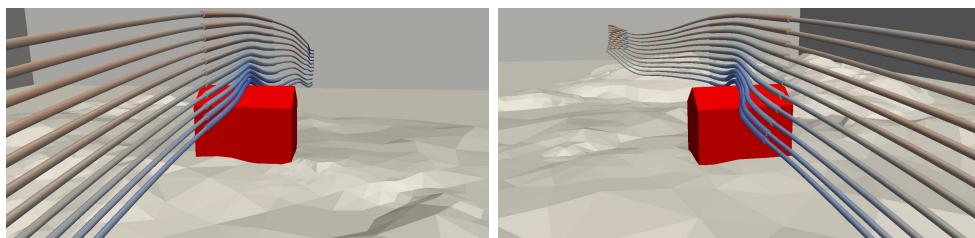


Figure 8.15: Tubed streamlines from the simulation with *one* overlapping house mesh. In front of and behind the house.

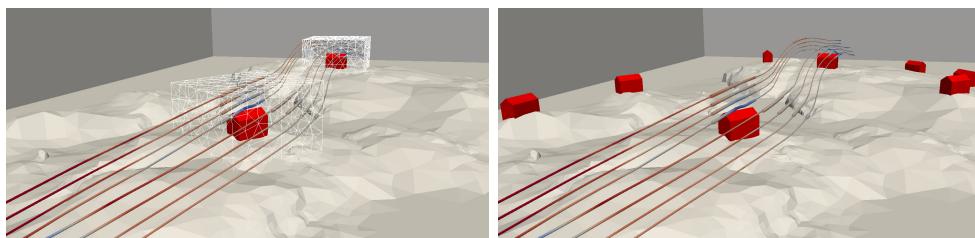


Figure 8.16: Tubed streamlines from the simulation with *two* overlapping house meshes. *Left:* Overview with the house meshes visible. *Right:* Overview without the house meshes visible but with some extra houses added for illustration.

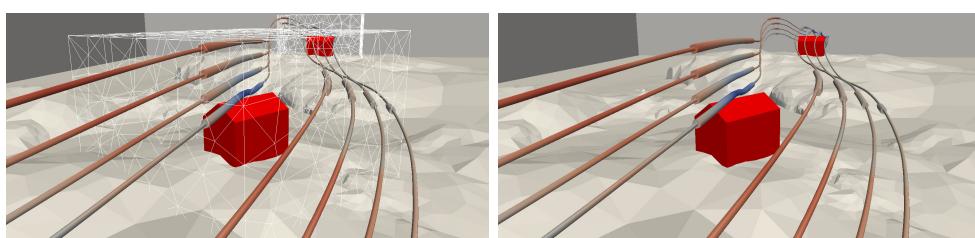


Figure 8.17: Tubed streamlines from the simulation with *two* overlapping house meshes. Closeup on the first house with and without the house meshes visible.

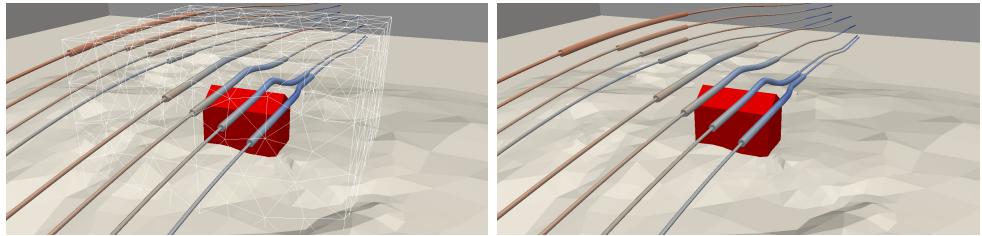


Figure 8.18: Tubed streamlines from the simulation with *two* overlapping house meshes. Closeup on the second house with and without the house mesh visible.

8.5.2 View

To test the computation of view, houses were arbitrarily positioned on the island as seen in Figure 8.19.



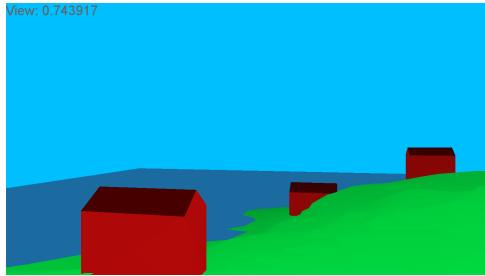
Figure 8.19: Overview of the island with ten houses. The yellow dot represents the position of the camera for the view computation.

For this test case we would like to examine how attractive it is to place a new house at the yellow dot, just next to the already existing house. The views in Figures 8.20–8.23 are all views from the camera located at the yellow dot in Figure 8.19.

Figures 8.20–8.23 show a variety of views from the selected location. Figure 8.20 shows a fair view with a lot of sky and sea view but also three houses and a part of the island can be seen. Therefore the view valuation becomes $V = 0.74$. To get a better view there should be as much sea or sky as possible and less houses and ground. A better example can be seen in Figure 8.21, where $V = 0.87$. Examples of views from the other end of the view scale can be found in Figure 8.22 and 8.23, where a neighbouring house blocks the view. In Figure 8.23 one can almost only see the neighbouring house, which also results in a very low value of the view.

The 360° horizontal view measure (8.8) naturally depends on the cardinal direction. We compute the value based on 32 computed rasterizations. If the south direction is chosen to be in the direction of the view, we obtain the

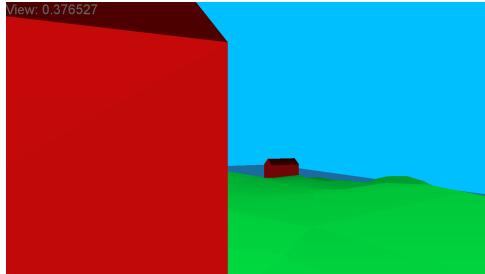
View: 0.743917

Figure 8.20: Fair view: $V = 0.74$

View: 0.869404

Figure 8.21: Good view: $V = 0.87$

View: 0.376527

Figure 8.22: Poor view: $V = 0.38$

View: 0.0219167

Figure 8.23: Bad view: $V = 0.02$

value $V_{360} = 0.68$ (as a result of looking straight at a neighboring house), whereas if we choose the south direction to be in the opposite direction of the neighbouring house we obtain $V_{360} = 0.73$.

8.6 Conclusions and future work

We have presented a generic framework for evaluating flow and view for settlement layouts. The framework allows multiple configurations to be computed and evaluated with relative ease. The current proof-of-concept implementation has several limitations that will be addressed in future work. These limitations and extensions fall into three different categories: efficiency, robustness and ease of use.

Regarding efficiency, the current implementation of multimesh methods adds a significant overhead in computational time compared to standard finite element methods. In particular, the assembly of multimesh finite element variational forms is significantly slower than standard assembly. Another limitation is the lack of a properly preconditioned iterative method (the current implementation uses the direct sparse solver UMFPACK). Both these limitations are the focus of ongoing work. Future work will also consider the extension of the current multimesh implementation to parallel architectures.

Regarding robustness, we noticed during this study that some particular configurations of buildings resulted in numerical instabilities. This is likely due

to bugs or untreated corner cases in the computational geometric framework of the FEniCS multimesh implementation. This issue is also the focus of ongoing work.

To be a useful tool in an iterative architectural process, the framework must not only be efficient and robust. It must also be easy to use. Future work will also consider the creation of a user-friendly interface. In particular, it would be highly relevant to consider the creation of VR or AR interfaces to our framework.

Acknowledgements

This work was supported by the profile area *Virtual Cities* as part of *Building Futures*, an Area of Advance at Chalmers. We are also greatful towards CREAM Architects in Gothenburg for ideas, feedback and inspiration during this project.

Bibliography

- [1] Baskaran, A., Kashef, A., 1996. Investigation of air flow around buildings using computational fluid dynamics techniques. *Engineering Structures* 18 (11), 861–875.
- [2] Becker, R., Burman, E., Hansbo, P., 2009. A Nitsche extended finite element method for incompressible elasticity with discontinuous modulus of elasticity. *Comp. Methods Appl. Mech. Engrg.* 198 (41), 3352–3360.
- [3] Benson, E. D., Hansen, J. L., Schwartz, A. L., Smersh, G. T., 1998. Pricing residential amenities: The value of a view. *The Journal of Real Estate Finance and Economics* 16 (1), 55–73.
- [4] Blocken, B., Carmeliet, J., Stathopoulos, T., 2007. Cfd evaluation of wind speed conditions in passages between parallel buildings—effect of wall-function roughness modifications for the atmospheric boundary layer flow. *Journal of Wind Engineering and Industrial Aerodynamics* 95 (9), 941–962.
- [5] Blocken, B., Stathopoulos, T., Carmeliet, J., Hensen, J. L., 2011. Application of computational fluid dynamics in building performance simulation for the outdoor environment: an overview. *Journal of Building Performance Simulation* 4 (2), 157–184.
- [6] Bourassa, S. C., Hoesli, M., Sun, J., 2004. What's in a view? *Environment and Planning A* 36 (8), 1427–1450.
- [7] Burman, E., Fernández, M. A., 2007. Stabilized explicit coupling for fluid-structure interaction using Nitsche's method. *C. R. Math. Acad. Sci. Paris* 345 (8), 467–472.
- [8] Burman, E., Hansbo, P., 2007. A unified stabilized method for Stokes' and Darcy's equations. *J. Comput. Appl. Math.* 198 (1), 35–51.
- [9] Catmull, E. E., 1974. A subdivision algorithm for computer display of curved surfaces. Ph.D. thesis, The University of Utah.

- [10] Geuzaine, C., Remacle, J.-F., 2009. Gmsh: A 3-d finite element mesh generator with built-in pre-and post-processing facilities. *International Journal for Numerical Methods in Engineering* 79 (11), 1309–1331.
- [11] Glowinski, R., Pan, T. W., Hesla, T. I., Joseph, D. D., Periaux, J., 2001. A fictitious domain approach to the direct numerical simulation of incompressible viscous flow past moving rigid bodies: application to particulate flow. *J. Comput. Phys.* 169 (2), 363–426.
- [12] Hansbo, A., Hansbo, P., 2002. An unfitted finite element method, based on Nitsche's method, for elliptic interface problems. *Comp. Methods Appl. Mech. Engrg.* 191 (47), 5537–5552.
- [13] Hansbo, A., Hansbo, P., Larson, M. G., 2003. A finite element method on composite grids based on Nitsche's method. *ESAIM, Math. Model. Numer. Anal.* 37 (03), 495–514.
- [14] Heckbert, P. S., 1989. Fundamentals of texture mapping and image warping. Tech. rep., University of California at Berkeley, Berkeley, CA, USA.
- [15] Heuveline, V., Ritterbusch, S., Ronnas, S., 2011. Augmented reality for urban simulation visualization. Preprint Series of the Engineering Mathematics and Computing Lab 0 (16).
- [16] Hussain, M. R. M., Tukiman, I., Zen, I. H., Shahli, F. M., 2014. The impact of landscape design on house prices and values in residential development in urban areas. *APCBE Procedia* 10, 316–320.
- [17] Ingelsten, S., Mark, A., Edelvik, F., Logg, A., Österbring, M., 2016. Urban cfd-simulation using point cloud data. In: Proceedings of NSCM-26: the 26th Nordic Seminar on Computational Mechanics.
- [18] Johansson, G., Karlén, F., Stark, M., 2014. Lilla fjellholmen, varsamt byggande i en unik skärgårdsmiljö. Master's thesis, Chalmers University of Technology, Gothenburg, Sweden.
- [19] Johansson, A., Larson, M. G., Logg, A., 2015. High order cut finite element methods for the Stokes problem. *Advanced Modeling and Simulation in Engineering Sciences* 2 (1), 1–23. URL <http://dx.doi.org/10.1186/s40323-015-0043-7>
- [20] Logg, A., Mardal, K.-A., Wells, G. N., et al., 2012. *Automated Solution of Differential Equations by the Finite Element Method*. Springer.
- [21] Logg, A., Wells, G. N., 2010. DOLFIN: Automated finite element computing. *ACM Transactions on Mathematical Software* 37 (2). URL <http://www.dspace.cam.ac.uk/handle/1810/221918/>

- [22] Massing, A., Larson, M. G., Logg, A., Rognes, M. E., 2013. An overlapping mesh finite element method for a fluid-structure interaction problem. submitted to Communications in Applied Mathematics and Computational Science.
- [23] Massing, A., Larson, M. G., Logg, A., Rognes, M. E., 2014. A stabilized nitsche overlapping mesh method for the stokes problem. Numerische Mathematik, 1–29.
- [24] Moes, N., Dolbow, J., Belytschko, T., 1999. A finite element method for crack growth without remeshing. Int. J. Numer. Meth. Engng 46, 131–150.
URL <http://venus.usc.edu/PAPERS/MultiScaleMechanics/XFEM.pdf>
- [25] Nitsche, J., 1971. Über ein Variationsprinzip zur Lösung von Dirichlet-Problemen bei Verwendung von Teilräumen, die keinen Randbedingungen unterworfen sind. In: Abhandlungen aus dem Mathematischen Seminar der Universität Hamburg. Vol. 36. Springer, pp. 9–15.
- [26] Pineda, J., 1988. A parallel algorithm for polygon rasterization. In: In Proceedings of Siggraph '88. pp. 17–20.