# INAUGURAL DISSERTATION

submitted to the
Faculty for Mathematics
of the Ruperto-Carola University of Heidelberg, Germany
for the degree of
Doctor of Natural Sciences

presented by
Diplom-Mathematiker
Sean Gillian Queisser
born in Hannover

Oral examination: .....

April 30, 2008

# The Influence of the Morphology of Nuclei from Hippocampal Neurons on Signal Processing in Nuclei

Gutachter: Prof. Dr. Gabriel Wittum

# Summary

Calcium signaling in neurons is a key regulator of gene expression and plays an important role in memory formation, learning and survival. One major route on which synapses and the nucleus of the neuron communicate, is through calcium waves that propagate from NMDA receptors – either synaptic or extra-synaptic – to the nucleus where they activate the transcription factor CREB.
Novel mechanisms were recently unveiled at the IZN in Heidelberg, that are responsible for nuclear plasticity. The morphology of nuclei undergoes changes in time, regulated by calcium influx through NMDA receptors. Changes in the nuclear morphology could affect the communication pathway between synapse and nucleus.
This thesis presents a novel inertia-based image processing filter that allows us to reconstruct the geometry of hippocampal nuclei from raw confocal microscopy data. A data base of reconstructed nuclei shows, that cell nuclei from hippocampal neurons contain deep infoldings of the nuclear envelope which increase nuclear surface size, minimize diffusion distances for calcium ions in the nucleus and form structural micro domains inside the nucleus.
In order to mathematically assess the function of these complex nuclear structures, a three dimensional model for calcium signaling in the nucleus, based on the reconstructed nuclear geometries, was developed. This model was fully implemented in the simulation environment *UG*, which offers multi-grid solvers for the model equations. Simulation results show, that the nuclear morphology in fact influences the way calcium signals propagate inside the nucleus. The infoldings of the membrane shorten diffusion distances, therefore calcium can reach more distal sites faster. Furthermore, infolded nuclei show higher levels of activity and are more adept at resolving signals at high frequencies.
In addition to the three dimensional model for calcium signaling a method for estimating the diffusion coefficient of nuclear calcium was developed. This method makes use of numerical optimization techniques with data from laser-assisted calcium uncaging experiments. Data driven simulations describe nuclear calcium to be in a buffered state, which is represented by an active diffusion coefficient of approximately 36 $\mu m^2/s$.

# Zusammenfassung

Kalziumsignale in Neuronen sind ein zentraler Regulator für Genexpression und spielen bei der Gedächtnisformation, dem Lernen und dem Überleben einer Zelle eine wesentliche Rolle. Ein bedeutender Kommunikationsweg zwischen Synapse und Zellkern basiert auf dem Transport von Kalziumionen, von synaptischen oder extra-synaptischen NMDA Rezeptoren zum Kern, wo diese den Transkriptionsfaktor CREB aktivieren.
Am IZN (Heidelberg) neu entdeckte Mechanismen sind verantwortlich für Zellkernplastizität. Die Morphologie von Zellkernen unterläuft zeitliche Veränderungen, die von Änderungen der zellulären Kalziumkonzentration durch NMDA Rezeptoren reguliert werden. Es besteht die Möglichkeit, dass die Kernmorphologie den Kommunikationsweg zwischen Synapse und Kern beeinflusst.
Diese Arbeit präsentiert einen neu entwickelten Trägheits-basierten Bildfilter, der es uns ermöglicht die Zellkerngeometrie von Neuronen aus dem Hippocampus aus konfokalen Mikroskopiedaten zu rekonstruieren. Eine Datenbank von rekonstruierten Zellkernen zeigt, dass diese Zellkerne tiefe Einfaltungen der Kernhülle aufweisen, welche die Kernoberfläche vergrößern, Diffusionsstrecken für Kalziumionen im Kern minimieren und strukturelle Micro-Domänen im Kern bilden.
Um die Funktion dieser komplexen Kernstrukturen mathematisch zu untersuchen wurde ein dreidimensionales Model zur Beschreibung der Kalziumsignalverarbeitung in Zellkernen entwickelt, basierend auf den rekonstruierten Kerngeometrien. Dieses Modell wurde vollständig in die Simulationsumgebung *UG* implementiert, welche Mehrgitter-Löser für die Modellgleichungen zur Verfügung stellt. Die Simulationsergebnisse zeigen, dass die Morphologie der Zellkerne in der Tat die in den Kern eindringenden Kalziumsignale beeinflusst. Einfaltungen der Kernmembran verringern Diffusionsstrecken, wodurch Kalziumionen entferntere Stellen im Kern schneller erreichen können. Zudem weisen eingefaltete Kerne höhere Aktivität auf und sind besser in der Lage hochfrequente Signale aufzulösen.
Zusätzlich zu dem dreidimensionalen Modell der Kalziumsignalverarbeitung wurde eine Methode zur Schätzung des Diffusionskoeffizienten von Kernkalzium entwickelt. Diese Methode verwendet Techniken der numerischen Optimierung zusammen mit experimentellen Daten, welche aus laserunterstützen Kalzium-uncaging Experimenten stammen. Diese datenunterstützten Simulationen beschreiben Kernkalzium in einem gepufferten Zustand, der von einem aktiven Diffusionskoeffizienten von ca. 36 $\mu m^2/s$ beschrieben werden kann.

# Contents

# List of Figures

# List of Tables

# Introduction

The brain constantly changes. It grows, forms and breaks connections between its cells and computes information passed on through a great network of cells. One of the brain's optimization tools is, that it can change its form, the branch structure of single cells and the connectivity between cell-networks. That way, information can be passed from one point to another faster or can be blocked entirely. It has been shown, that neurons undergo strong structural alterations under specific activation. We have been investigating the possibility, that not only cells themselves can change their form, but also their cell organelles, for example the cell nucleus. Each neuron has a cell nucleus that contains the DNA and is surrounded by a double-layered membrane, each built in a bi-lipide structure. Inside the nucleus, gene expression – the biochemical transformation of DNA into specific amounts of certain proteins – is activated depending on the signal passed on to the nucleus.
One of the most pronounced ways the cell communicates information to its nucleus is by calcium ion signals. These can be synaptic NMDA- or extra-synaptic NMDA-gated calcium signals. NMDA (N-methyl-D-aspartate)-induced calcium signals are essential for long-lasting responses, that play an important role in memory formation and learning.
Experiments carried out at the Interdisciplinary Center for Neurosciences (IZN) in Heidelberg revealed novel structures inside the neuron's nucleus. Questions about these structures, whether the form of the nucleus can undergo morphological alterations and how these might affect the communication pathway between neuron synapses and the nucleus, arose. In an interdisciplinary effort the department SiT (IWR Heidelberg) together with the IZN began addressing these questions from a mathematical and neurobiological point of view.
The development of an inertia-based nonlinear anisotropic diffusion filter allowed us to create three-dimensional reconstructions of nuclei from hippocampal neurons from raw microscopy image stack recorded with a confocal microscope. We began setting up a data base of reconstructions to investigate the nature of the newly discovered morphological features of nuclei. At the same time, Hilmar Bading's lab at the IZN was able to show that the impressive forms of nuclei were in fact regulated by NMDA receptors and that nuclei undergo structural changes in time. This nuclear plasticity, resulting in strong infolding of the nuclear envelope might influence the relay of information in and out of the nucleus.
This prospect led us to develop a mathematical model which can simulate signal pro-

cessing in nuclei, based on the newly won three-dimensional reconstructions. Since the model includes the actual form of single nuclei which is never the same (think of a fingerprint), we were in a position to investigate the morphological effect on the relay of information.
Our project revealed that nuclei are highly plastic and depending on their form minimize diffusion distances for calcium, separate the nucleus into structural and functional micro-domains that make it possible for the nucleus to interpret calcium signals in different ways depending on the location in the nucleus. Infolded nuclei are more adept at resolving high frequency signals and show overall higher levels of activity compared to spherical nuclei.
In order to have a more detailed picture of the propagation of calcium signals in nuclei we further developed a method, based on numerical optimization techniques for parameter estimation, to calculate the degree of calcium buffering in the nucleus which is revealed in the diffusion coefficient embedded in the data-driven model for nuclear calcium signaling. A. Eder from the Bading lab provided data from laser-assisted calcium uncaging experiments that completed a mathematical model for the estimation of the calcium diffusion coefficient.
This thesis is divided into three main parts. Following a short introduction into the neurobiological basics needed to understand the principles of the developed mathematical models, we present a novel inertia-based image filter in Chapter 2. In the second part, Chapter 3 will give a detailed presentation of the development of a data-driven, three dimensional model for signal processing in nuclei, along with a number of evaluation tools for interpreting the model results. The mathematical model, which is based on partial differential equations and multi-grid solving methods, was implemented in the simulation environment *UG*. A short introduction of *UG* is further given.
In Chapter 4 a data-driven model, also implemented in *UG*, with which the diffusion coefficient of nuclear calcium can be estimated, is introduced. Finally, results of three dimensional reconstructions and tests of the underlying inertia-based filter as well as modeling and parameter estimation results are presented in Chapter 5.
The thesis concludes with a discussion of the developed methods and their results and an Appendix containing an excerpt of the 3D-data base of hippocampal neuron nuclei.

# Chapter 1

# The Neurobiology of Hippocampal Neurons

## 1.1 The Hippocampal Neuron Nucleus

The nucleus of hippocampal neurons might be described as the brain of the single neuron cell, interpreting incoming signals and producing outgoing ones. Although seemingly important, until now little is known about detailed functionality of the nucleus.
The nucleus is a cell organelle surrounded by a double lipide membrane separating it from the *cytoplasm*. The diameter of the nucleus lies at approximately 10 $\mu m$, contained in the nucleus is the *nucleoplasm* and vast amounts of chromosomal *DNA*. The bi-lipide nuclear envelope is perforated by pore complexes, several thousands evenly distributed on the nuclear envelope. The pore complexes are almost 100 $nm$ in diameter, although only the innermost part of the complex functions as a passive pathway for the exchange of small, medium sized and even macro-molecules, [27, 25]. The outer membrane extends into the *endoplasmatic reticulum* (ER), whereas the inner membrane is connected to a protein grid called the nuclear *lamina*. Aside from recent findings about divers functionality of the nuclear lamina, it is considered to be the form-stabilizing component of the nucleus, [22].
The nucleus carries out a number of tasks, the two major functions are the synthesis of *RNA* and the synthesis and repair of *DNA*, [15].

## 1.2 Signaling Cascades through NMDA Receptors

Cell functions are finely regulated by changes in the concentration of molecules which interact, promote or inhibit steps in a *signaling cascade* that result in most divers products. These processes are so sensitively tuned as from where the signal originated or which molecular binding partners were released simultaneously, that every task the cell needs to carry out can be represented by this highly combinatorial set-up.

### 1.2.1 The Glutamate Receptor NMDA

NMDA (*N*-*m*ethyl-*D*-*a*spartate) receptors act as cation channels and are regulated by the glutamate-neurotransmitter. One feature of NMDA channels is, that they are blocked by magnesium ions. So in order for these channels to be activated, the magnesium block needs to be alleviated by depolarisation of the cell. Therefore, NMDA channels are highly inert at base-level activity of the cell and only become active channels, especially for calcium ions, when the right molecules are present at the right time, [59]. NMDA receptors are found to be a vital mediator of signals responsible for long-term potentiation (LTP) and long-term depression (LTD). These two processes are part of a memory formation and learning system, [9]. NMDA-receptors are found at different locations, at synaptic and extrasynaptic sites, [3].

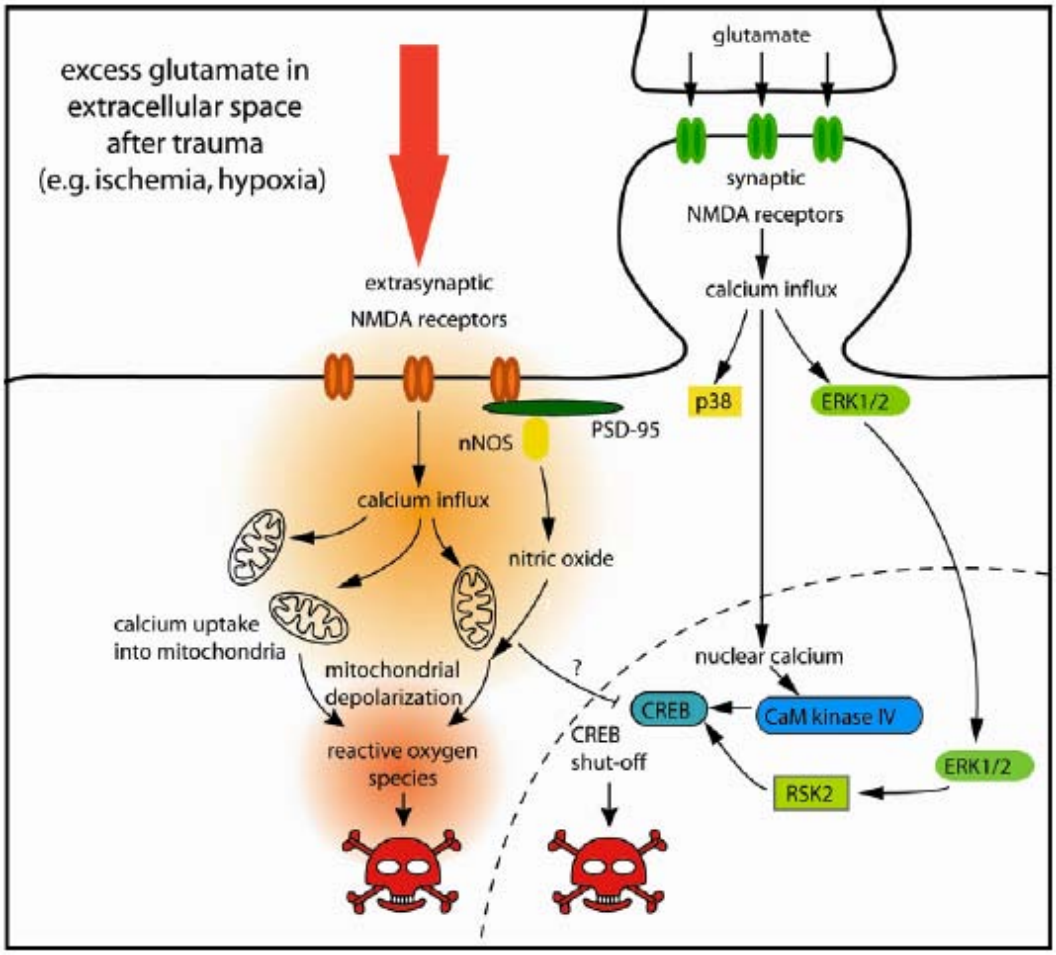


Figure 1.1: *Extrasynaptic vs. synaptic NMDA receptors: calcium gated through extrasynaptic channels results in cell death, while synaptically NMDA-gated calcium influx results in cell survival (Image by M. Wittmann (IZN)).*

### 1.2.2 Cell Death or Cell Survival?

The fate of a cell is closely related to the origin of cellular signals. As the result of extrasynaptic NMDA receptor activation, a calcium wave propagates into the mitochondria. This process generates reactive oxygen species that lead to cell death, [34, 67]. The same process – NMDA activation producing a calcium signal – but this time passing through activated synaptic NMDA receptors results in cell survival, [34, 67], Figure 1.1.

## 1.3 Information Processing via Calcium Oscillations

### 1.3.1 Calcium Waves: From Synapse to Nucleus

There are two major pathways for calcium ions to reach the nucleus. One is a direct calcium wave propagation through the cytosol which enters the nucleus through its nuclear pores. This calcium route is a fast signaling path, since no further molecules are involved, which means that no reaction lag-time occurs. A slower pathway is one that involves ERK1/2 and RSK2 activation, [7], Figure 1.2. In this case it is ERK1/2 that enters the nucleus via its nuclear pores.

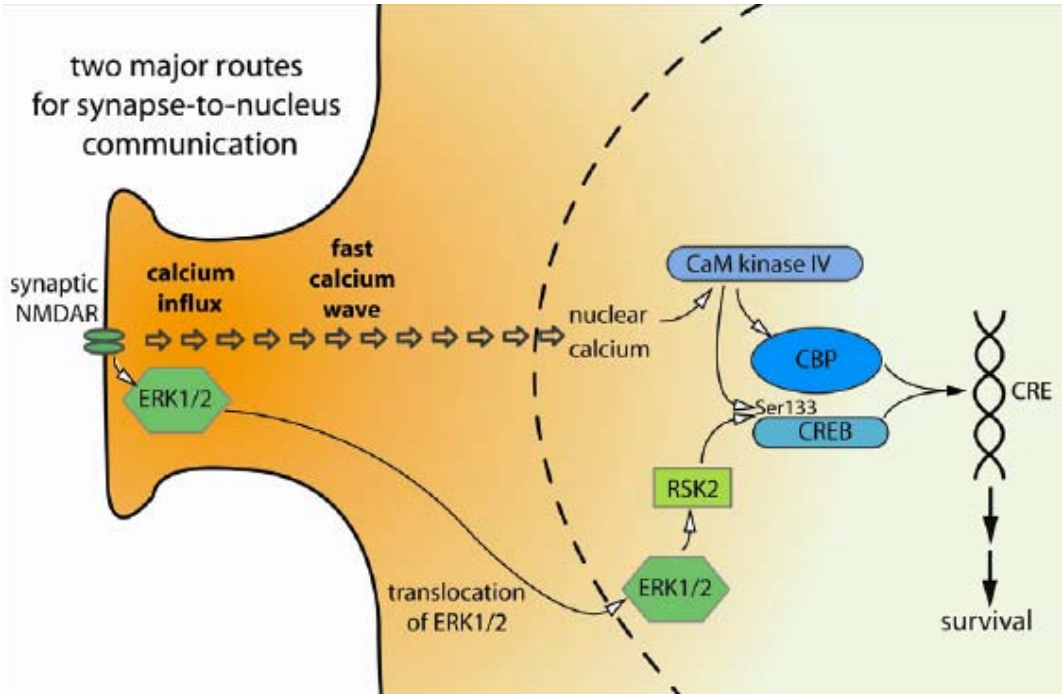


Figure 1.2: *Synapse-to-nucleus communication: Two major routes are important for communication between synapse and nucleus, a direct calcium wave generated at synaptic NMDA sites enters the nucleus and calcium induced ERK1/2 de-localization towards the nucleus (Image by M. Wittmann, IZN).*

### 1.3.2 Protein Synthesis in the Nucleus

What happens when calcium enters the nucleus? Calcium binds to calmodulin and this complex activates CaM Kinase IV. This induces CREB phosporylation and CREB-dependent gene expression. CaM Kinase IV then does three things, [33]:

1. Phosphorylation of CREB on Ser133.

2. Activation of CBP.

3. Stimulation of CREB/CBP-mediated transcription.

The activated CBP and phosphorylated CREB together produce the code for CREB-dependent gene transcription. The second ERK1/2 pathway is not able to activate CBP, but it can phosphorylate CREB on Ser133, raising the probability for the first pathway to result in gene transcription, [33, 12].

### 1.3.3 Governing Parameters of the Calcium Code

A rise in calcium ion concentration at a synaptic NMDA site can result in the transcription of CREB-dependent genes. And not only is it vital to the outcome if calcium is released from synaptic or extrasynaptic NMDA channels: The calcium wave itself carries information for the cell about if or if not to induce gene transcription. And if yes, how long and how much? These decisions are thought to be encoded in the cytosolic calcium signal that enters the cell nucleus.
The key for differential gene expression control lies in the different calcium signals that are produced by different neuronal firing patterns, [5]. Neuronal firing patterns produce calcium waves with distinct duration and distinct amplitude. Dendritic action potentials (dAPs), depending on their duration, elicit low to high calcium transients, which are responsible for short-lasting or enduring plasticity of the nervous system, [5]. Aside from duration and transient amplitude a third parameter, the transient frequency, regulates the efficiency and specificity of gene expression.
Using a calcium clamp technique, Ricardo E. Dolmetsch et. al. from the Department of Molecular and Cellular Physiology at Stanford University showed, that the frequency of calcium oscillations altered the calcium threshold for activating transcription factors, [19]. Fast calcium oscillations stimulate transcription factors better than a constant calcium signal when both signals have the same amplitude.
Governing parameters for coding information in a calcium transient are therefore:

1. Transient-*duration*,

2. Transient-*amplitude* and

3. Transient-*frequency*.

Calmodulin plays an apparent role in resolving frequency-coded calcium signals in the nucleus, [46]. Changes in the response of Calmodulin are regulated by factors such as amplitude, duration and frequency of individual calcium spikes and result in distinct amounts of kinase activity, [41, 46].

## 1.4 The Nuclear Morphology: A Lot More than Just a Sphere

Text book depictions of cell nuclei typically show a nucleus having a spherical shape. This is not always the case. Deformation of the nuclear membrane has been observed in several cases. Membrane creases of the nucleus can be witnessed in nuclei from the visual cortex, [11], in Hela cells the membrane forms tubular structures, [23]. In both cases, the outer and inner membrane are deformed uniformly. In other cells only the inner membrane undergoes deformation, for example in nuclei from Drosophila, [37]. The geometry of hippocampal neuron nuclei is controlled by NMDA receptor signaling, [67, 68], and undergoes significant morphological changes.

### 1.4.1 First, there was EM

Recent electron microscopical studies by A. Hellwig from the Interdisciplinary Center for Neuroscience (IZN) showed, that nuclei from hippocampal rat neurons contain infoldings of the inner and outer membrane, Figure 1.3. The studies show, that

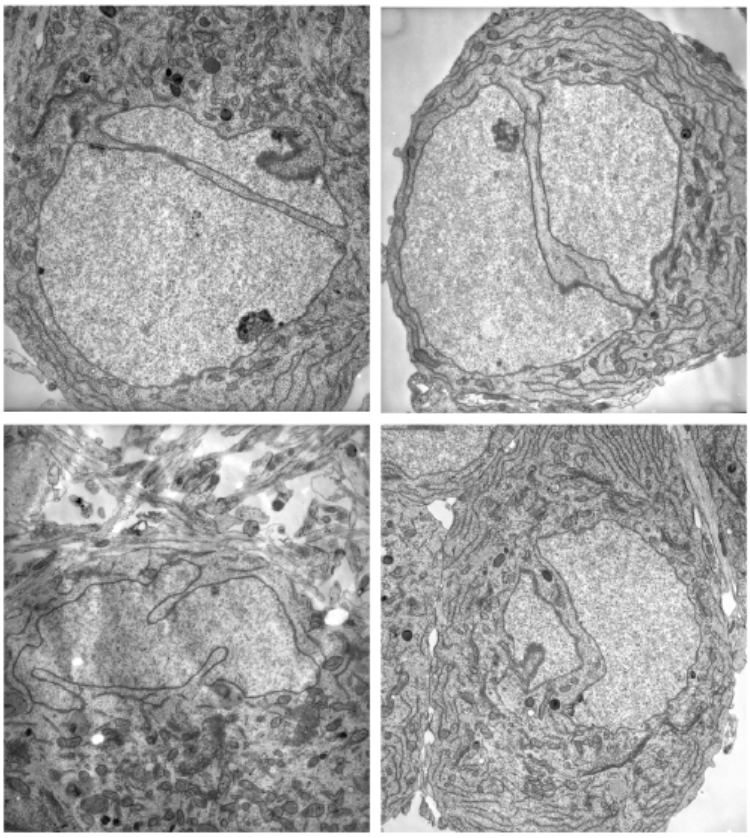

Figure 1.3: *Examples of electron microscopy (EM) recordings: EM reveals infoldings of the nuclear envelope (by A. Hellwig, IZN).*

nuclear pores are present at the site of infoldings, Figure 1.4. Questions about the morphology of these infoldings in three dimensions and their biological purpose were raised. In order to clarify the morphological question (infoldings or tubular) as well as determining whether the membrane surface increases or nuclear volume decreases compared to non-infolded nuclei, three dimensional reconstructions of hippocampal nuclei were necessary. These reconstructions could be retrieved from confocal microscopy images.

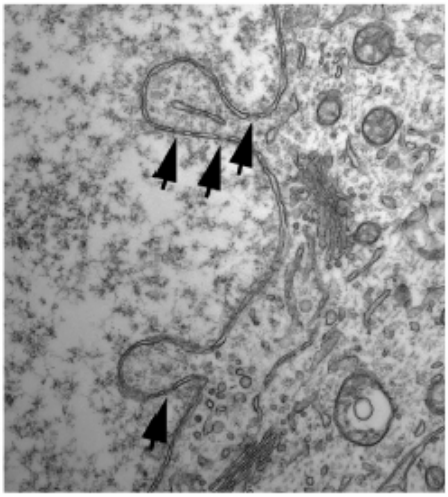

Figure 1.4: *EM close-ups: Electron microscopy proves that both the inner and outer membrane of the nuclear envelope are infolded and that infoldings contain nuclear pore complexes that allow diffusion of calcium ions into the nucleus (by A. Hellwig, IZN).*

### 1.4.2 Image Recordings for Reconstructions

Two major features for microscopy image recording for three-dimensional reconstructions have to be fulfilled. A nucleus has to be recorded in the x-y-plane as well as in z-direction, secondly signal-to-noise ratio has to be sufficient for certain image processing steps. The first demand rules out recordings from electron microscopy since the recording of a stack for a single nucleus bears significant technical problems. This leaves confocal fluorescence microscopy. Staining of the nuclear membrane as well as staining of the nuclear lamina was tested, [67]. Staining of the nuclear membrane proved to be technically difficult. Yet the staining of the nuclear lamina showed the desired effect in cultured cells, Figure 1.5. A strong signal from the lamina, combined with low background noise, produced sufficient images, [67]. Staining in live recordings proved insufficient due to bad signal-to-noise ratios, Figure 1.6. The nuclear lamina is the form-giving protein-grid for the nucleus and therefore adequate as a representation of the nuclear geometry. Using a Leica SP2 laser scanning confocal microscope, primary hippocampal neurons from rats stained with an

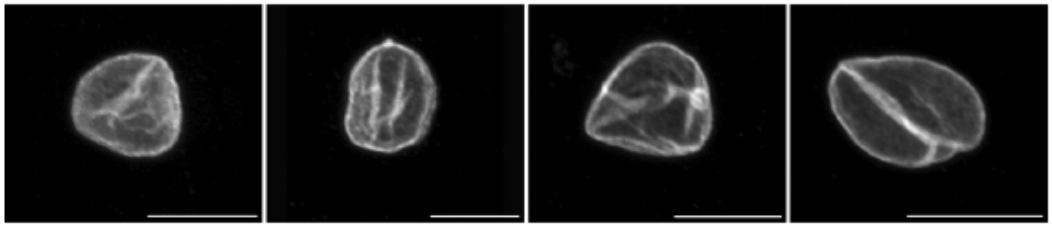

Figure 1.5: *Confocal fluorescence microscopy: LaminB staining visualizes the infolded envelope of nuclei with strong signal to noise ratio (by M. Wittmann, IZN). Scale bars: 10 µm*

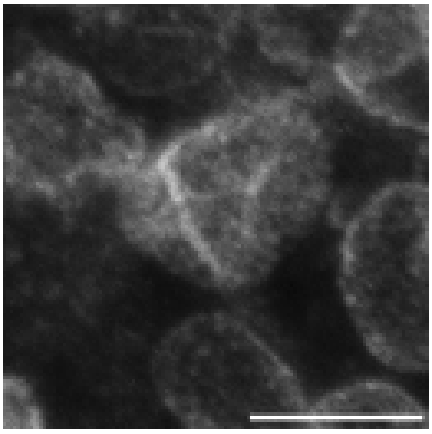

Figure 1.6: *Live recordings of cell nuclei lack necessary signal contrast for reconstruction purposes (by M. Wittmann, IZN). Scale bar: 10 µm*

Alexa488-labelled secondary antibody which binds to a primary laminB-antibody, were recorded. Stacks consisting of up to 70 images with constant distance were recorded.

## 1.5 Evaluation of Experimental Data

Stack recordings of single nuclei and calcium imaging data form two separate data sets, which are used for mathematical reconstruction of the geometry of hippocampal neuron nuclei and for verification of theories and results from mathematical simulations concerning nuclear calcium dynamics.

### 1.5.1 Evaluating Single Nuclei and their Microdomains

For the evaluation of image sequences; recorded in equidistant time steps, the focus lies on the global calcium concentration in single nuclei, as well as investigating

special *regions of interest* (ROIs) within a single (infolded) nucleus. With ImageJ, [38], single sequences in time can be evaluated by summing up gray value intensities which then are calibrated, according to [26], delivering a parameter for the global calcium concentration of the nucleus.
In order to evaluate regions of interest, a separate image of the nucleus, recorded with a different marker (nuclear contrast is thus enhanced), serves as a visual aid to distinguish and define ROIs within an infolded nucleus, Figure 1.7. Single ROIs, using the ROI manager of ImageJ, can be evaluated and calibrated.

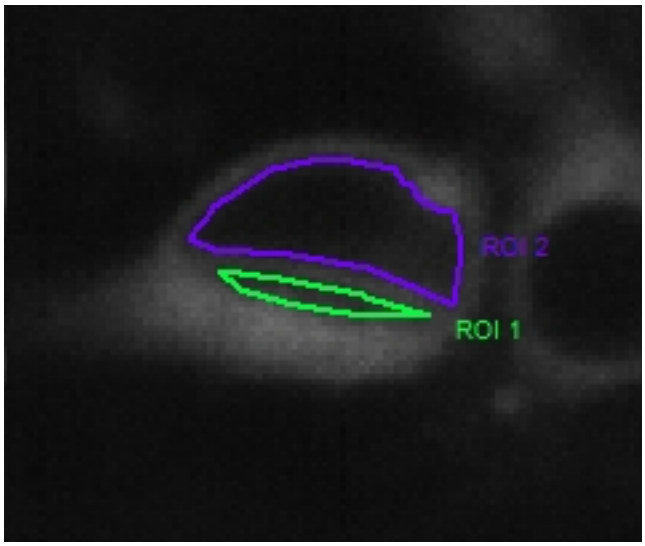


Figure 1.7: *Selection of different sized micro domains using ImageJ's ROI Manager. Separate regions of interest can be evaluated with specially developed macros.*

### 1.5.2 Measuring Nuclear Calcium Activity

Given the assumption, that molecular activation in the nucleus which initiates gene expression is threshold dependent, this concept needs to be investigated from an experimental and modeling side. Nuclear activity is defined as follows:

> *Nuclear activity* is the percentage value of nuclear area above a given threshold.

In order to measure nuclear calcium activity, a macro for ImageJ was developed. A gray value threshold $T$ is derived from a calibrated threshold $t$ using the following equation, [26]:

$$\left[Ca^{2+}\right] = \frac{F - F_{min}}{F_{max} - F}.$$

Having chosen a threshold $t$ yields:

$$t = \frac{T - F_{min}}{F_{max} - T}.$$

The gray value threshold that is included in the ImageJ macro can be calculated with the equation

$$T = \frac{t \cdot F_{max} + F_{min}}{1 + t}. \tag{1.1}$$

By segmentation of the entire image sequence using the threshold calculated by equation (1.1), all pixels above threshold are set to value 1 (active area) and every pixel below is set to 0 (inactive area). The Activity macro designed for ImageJ calculates the ratio between active area and entire ROI area. The macro then saves the activity data in each time step in a double rowed table, which can be plotted with *Gnuplot* or other plotting tools.

# Chapter 2

# Image Processing

## 2.1 Processing Data for Reconstruction of Hippocampal Nuclei

Processing raw microscopy data with high-end image processing tools is vital ground work for morphological identifications of hippocampal nuclei as well as for simulation purposes. The idea of applying a diffusion process on raw image data has been realized in *isotropic* diffusion filters; the convolution of the raw image with the Gaussian is one example. A major drawback is inadequate handling of intra-structures of the image, such as edges and contours. This problem gave rise to *anisotropic* diffusion approaches, P. Perona, J. Malik, [54], and J. Weickert, [66, 65], amongst many others proposed an adaptive diffusion tensor instead of a scalar coefficient to deal with substructures in image data.
Further steps in filter techniques were taken in the software project *NeuRA*, [49]. Based on concepts from Lenzen and Rumpf, [44], a powerful structure detection mechanism for linear substructures, based on the local moments of inertia, was added to the diffusion filter. This structure detection mechanism was designed to identify linear, dendritic structures of hippocampal neurons, [10], and further developed in order to identify two dimensional, nuclear membrane sub-structures, [56]. This filter can be described as an *inertia-based nonlinear anisotropic diffusion filter*. The combination of image filtering and image segmentation, [52, 56], defines the first part of image processing of hippocampal neuron nuclei.
In a second step, structural information of the nucleus can be extracted using an isosurface- and grid-generating software toolbox, developed by A. Heusel, SiT. Surface and volume grids can then be measured and visualized, [56, 55], and exported to a simulation environment.

### 2.1.1 Filtering with Nonlinear Anisotropic Diffusion Filters

Within the field of image processing, a broad spectrum of processing tasks for image data created a broad spectrum of image filters to carry them out. In the case of image data processing of hippocampal nuclei recordings from confocal microscopy, the following aspects need to be fulfilled by the filter:

- Enhance signal-to-noise ratio in 3D data sets.
- Homogenize membrane structure: close gaps caused by low fluorescence binding in image recording.
- Maintain structural information, e.g. membrane diameters and filigreed infolding structures.

To clarify "diffusion" of an image, one can imagine each gray value of a voxel as the concentration of a substance. When diffusion is allowed (think of an "adiabatic" wall being lifted in a thermodynamic system) the "gradients" in the image are resolved.

After a certain time, the image would turn into a single gray value image. One can stop diffusion after a sensible number of time steps and retrieve a blurred, but homogenized image. Often, this is not enough. In order to maintain borders of structures within the image, great effort has been made to add "edge detection" or "structure detection" to the filter.
Putting the above words into a mathematical formulation, a diffusion filter can be characterized in the following way, [66]:

$$\frac{\partial u}{\partial t} = div(D(\nabla u_\sigma) \cdot \nabla u) \qquad \text{on } \mathbb{R}^+ \times \Omega \tag{2.1}$$

$$u(x,0) = \text{raw microscopy data} \qquad \text{on } \Omega \cup \partial\Omega \tag{2.2}$$

$$(D(\nabla u_\sigma) \cdot \nabla u) \cdot \vec{n} = 0 \qquad \text{on } \mathbb{R}^+ \times \partial\Omega \tag{2.3}$$

where $u$ denotes the image's voxel set and $\vec{n}$ the outer normal. The diffusivity tensor $D$ is dependent on $u_\sigma$ which is defined as

$$K_\sigma := \frac{1}{2\pi\sigma^2} \cdot \exp\left(-\frac{|x|^2}{2\sigma^2}\right), \tag{2.4}$$

$$u_\sigma := (K_\sigma * \tilde{u}(.,t))(x), \quad \sigma > 0, \tag{2.5}$$

$\tilde{u}$ being the extension on $\mathbb{R}^n$ of $u$ on $\Omega$.
The key to directing diffusion in designated directions lies in the choice of the diffusivity tensor $D$. In [65] three different types of diffusion filters are listed, which are based on equations (2.1) – (2.3) and are defined by the choice of $D$:

**Linear isotropic diffusion filters:** A wide ranging technique which is equivalent to a convolution of the original data set with the Gaussian is derived from setting the diffusion tensor to the unit matrix

$$D(\nabla u_\sigma) = \mathbb{I}. \tag{2.6}$$

The drawback of this method is the fact, that edges are blurred and vital structure in the image is lost.

**Nonlinear isotropic diffusion filters:** Edges in the image are preserved, when diffusion is blocked at the location of an edge. At the same time diffusion is supposed to take place within an area defined by the surrounding edge. By defining a decreasing function $g \in C^\infty([0,\infty),(0,1])$ which satisfies $g(0) = 1$ and $g(s) \to 0$ for $s \to 0$ the above criteria are fulfilled. In [65]

$$D(s) := g(s) := \exp\left(-\frac{s^5}{5\lambda^5}\right) \tag{2.7}$$

is proposed.

**Nonlinear anisotropic diffusion filters:** By adding anisotropy to the nonlinearity of the filter, using the eigenvectors and eigenvalues of the diffusion tensor, an edge can be identified (nonlinear filter) and the direction of the edge approximated (nonlinear anisotropic filter). The *Weickert*-filter, [65, 66], uses the diffusion tensor

$$D(\nabla u_\sigma) := g(|\nabla u_\sigma|). \tag{2.8}$$

In [54] conduction coefficients – called edge enhancers – are introduced to add edge sharpening to anisotropic diffusion. This method substitutes other ill-posed approaches of backward diffusion problems that often result in numerically unstable designs. As an edge detector

$$g(s) := \frac{c}{1 + \left(\frac{s}{K}\right)^{1+\alpha}} \tag{2.9}$$

with constants $c$ and $K$ and a value $\alpha > 0$ is proposed in [54].

### 2.1.2 Adding More Structure Detection to the Filter

The edge detector as a function (2.7) or (2.9) does not entirely suffice for the underlying problem, see Chapter 5. The need for additional structure identification for gap closure and signal to noise ratio enhancement becomes obvious. The inertia-based nonlinear anisotropic diffusion filter, [55, 56], is based on the physical *moment of inertia*:

$$M^2(x_0) := \int_{B_\delta(x_0)} u(x)(x - M^1(x_0)) \otimes (x - M^1(x_0))dx \tag{2.10}$$

with $M^0$ being the "mass" of the integration volume $B_\delta$:

$$M^0(x_0) := \int_{B_\delta(x_0)} u(x)dx \tag{2.11}$$

and $M^1$ the center of mass

$$M^1(x_0) := \frac{1}{M^0(x_0)} \int_{B_\delta(x_0)} u(x)xdx. \tag{2.12}$$

The moment of inertia (2.10) is used instead of an edge detector. After applying the principal axes transformation

$$TM^2T^{-1} = \begin{pmatrix} \lambda_1 & & 0 \\ & \ddots & \\ 0 & & \lambda_n \end{pmatrix}, \quad T^tT = \mathbb{I} \tag{2.13}$$

with eigenvalues $\lambda_1 \geq \ldots \geq \lambda_n$ in $\mathbb{R}$ the diffusion tensor $D$ is defined as

$$D := T^t \begin{pmatrix} \lambda_1 & & 0 \\ & \lambda_2 & \\ 0 & & \lambda_3 \end{pmatrix} T \tag{2.14}$$

Since the nuclear membrane is a two-dimensional substructure in the three-dimensional raw image set, diffusion needs to be allowed in the two main directions (directions with the largest two eigenvalues) and blocked perpendicular to the membrane structure (third eigenvector direction). For this purpose, the first two eigenvalues in (2.14) are set to maximum diffusion value 1 and the third eigenvalue to a near zero value $\varepsilon$:

$$D(u) := T^t \begin{pmatrix} 1 & & 0 \\ & 1 & \\ 0 & & \varepsilon \end{pmatrix} T. \tag{2.15}$$

The inertia-based nonlinear anisotropic filter has a dimension-independent implementation with semi-implicit time discretization and finite volume discretization in space, [56]. This is an advancement of earlier nonlinear anisotropic diffusion approaches, which inherits the sensitivity to edges and border from other nonlinear anisotropic filters, [54, 66], and adds advanced capabilities in gap closure and signal-to-noise ratio enhancement.

### 2.1.3 Segmentation

Following image data filtering, the image stack is segmented. This means, that based on a gray value threshold every voxel in the image stack is transformed into a membrane voxel (white) or a background voxel (black). One of the most prominent segmentation algorithms is the one of Otsu, [52]. Within a sub-volume that scans the entire image stack, a local threshold is determined by maximizing the variance of the overlapping parts of the signal and noise histograms. Based on this threshold every sub-volume of the stack is segmented. Opposed to other segmentation techniques (e.g. global thresholding), the Otsu segmentation algorithm closes gaps in the membrane, yet at the same time preserves filigreed infolding and membrane structure, Figure 2.1.

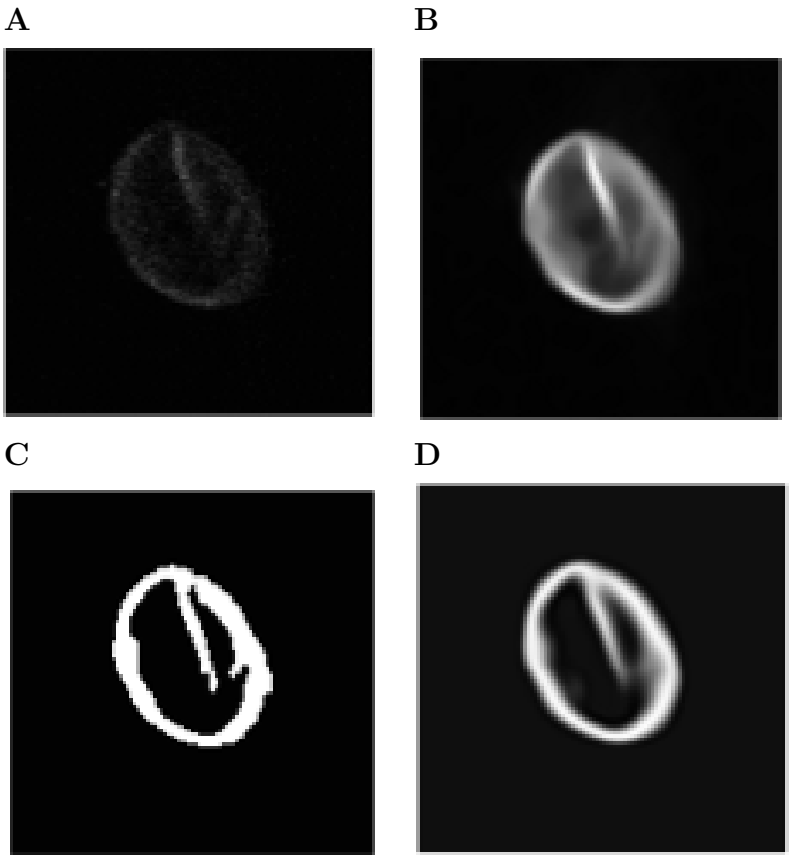


Figure 2.1: *Image processing steps: (A) raw microscopy data. (B) filtered with inertia-based filter. (C) segmented with Otsu-method. (D) second pass of inertia-based filter.*

## 2.2 Generating Grids from Processed Microscopy Data

The purpose of the image processing steps in Section 2.1 is to transform the microscopy raw data in order to extract a closed and smooth surface grid as a representation of the nuclear membrane. Based on such a surface grid, a volumetric grid of the nucleus can be derived, both grids form a discretization of the nucleus for the mathematical model.

### 2.2.1 Surface Grids

A surface grid is a discrete representation of the nuclear membrane, to be more specific, a representation of the recorded nuclear lamina. An isosurface-generating software toolbox, developed by A. Heusel and S. Reiter (SiT), is used to compute triangular surface grids from processed microscopy data. Based on a chosen gray value, a cube divided into five tetrahedra scans the image stack, Figure 2.2. The

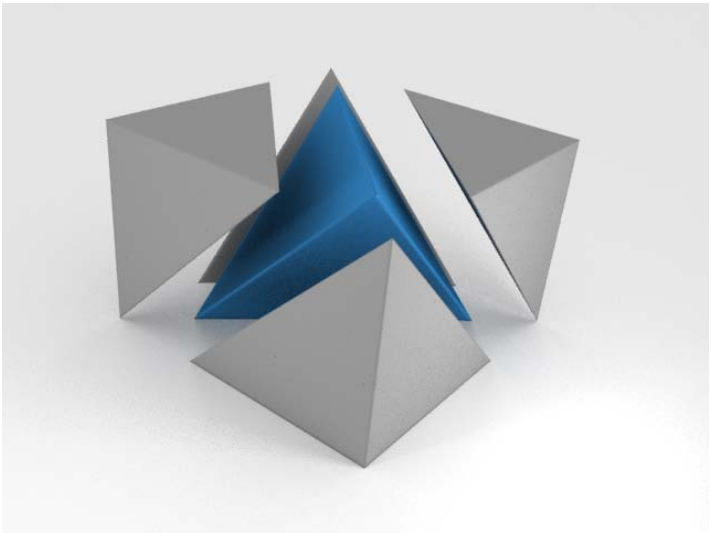

Figure 2.2: *Interpolation scheme: Cube is divided into 5 tetrahedra. Along tetrahedral lines gray values are interpolated linearly (by A. Heusel, SiT).*

corners of each tetrahedra contain the corresponding voxel values, between which the program interpolates values along the tetrahedra edges linearly. The determined position of the predefined gray value in saved and added to the grid as a grid node. Later, all grid nodes are connected to form a triangular and closed grid, i.e. every node has at least four neighbor nodes. Due to scattering of the laminB marker, the grid generating program will produce an "outer membrane grid" and an "inner membrane grid".
Of interest is the inner membrane, since it contains the infolded structure, Figure

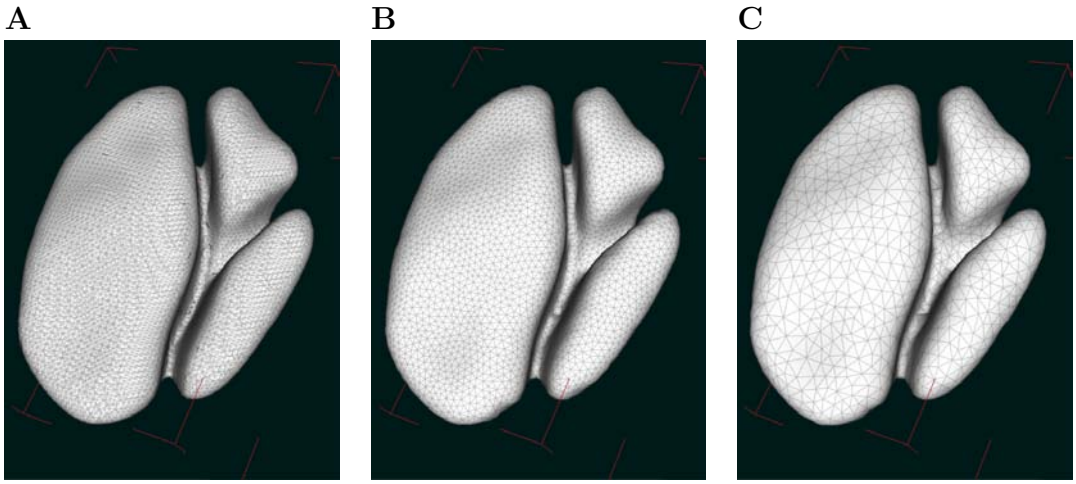


Figure 2.3: *Optimizing surface grid: (A) In-homogeneous fine surface grid. (B) First coarsening and homogenization of grid. (C) Second step in optimizing surface grid.*

2.1. This is not because the outer membranes is not infolded – because it is (see Section 1.4) – but because the resolution of confocal fluorescence microscopy is lower compared to electron microscopy and therefore not capable of tracing the infolding of both membranes separately. The grid that is generated using a modified marching tetrahedra algorithm still contains both inner and outer grid. By applying a Dijkstra graph search algorithm, [16], both grids are separated and the inner grid can be further processed.
Further processing means optimizing the triangular surface grid. The "raw" grid still contains a large number of triangles, often with strongly varying size. This leads to rough and "spiky" parts in the infoldings. To smoothen the surface structure (this is not an artificial alteration of the nuclear geometry, rather a convergence towards its original form), minimize the number of triangles while also minimizing the variance in triangle size, S. Reiter (SiT) has developed grid optimization tools, which can be applied, see Figure 2.3. The inner surface grid is then ready to be exported for volume grid generation.

### 2.2.2 Volume Grids

In analogy to surface grids, a volume grid has to be a sufficient representation of the volumetric structure of the nucleus. The software tool *Ansys* ICEM CFD, [2], offers the necessary procedures to create a volumetric tetrahedral grid derived from the surface grid generated in Section 2.2.1. Filters designed by A. Hauser (SiT) are used to transform the surface grid file into the file format used by ICEM CFD. The first tetrahedral grid can be coarsened in ICEM CFD, if the grid is used for simulation purposes in *UG* (*UG* has its own multi-grid refinement tools).

## 2.3 Measuring Hippocampal Nuclei

When investigating the phenomenon of the infolding of nuclear membrane in time, two theoretical scenarios are conceivable. Either the cell activates a protein integrating process which adds membrane to the existing nuclear envelope, thus enlarging the nuclear surface; or the process is comparable to a deflating soccer ball, meaning the membrane surface stays constant, while the infoldings express nuclear content into the cytosol. In the first case an increase in membrane surface size should be observable, while at the same time the nuclear volume stays constant, in the second case, membrane surface would stay constant, at the same time nuclear volume would decrease. To examine these hypotheses, a measuring tool for neuron nuclei was developed, [56]. Realized in *Perl* script, the nuclear volume and surface size can be evaluated based on the surface and volume grids generated according to Sections 2.2.1–2.2.2.

## 2.4 Visualizing 3D-Reconstructions

The novel morphological features of nuclei from hippocampal neurons had first been observed in microscopy studies, see Chapter 1. Since the visualization in that case is of single slices in two dimensions, it is hard to anticipate the three-dimensional structure of membrane infoldings. Having obtained 3D-grids of nuclei, these can be visualized. In early stages of the project, the Visualization Toolkit *VTK* was used, as the state of the art application an in-house visualization tool *ProMesh*, developed by A. Heusel (SiT), is used. Examples of novel 3D-visualizations of hippocampal neuron nuclei are demonstrated in Figure 2.4 and the Appendix. Unlike other cell types, e.g. HeLa cells, it becomes obvious, that the infoldings in hippocampal nuclei are not tubular, but actual infoldings that can extend deep into the nucleus. One can observe various infolding structures, from single to multiple, branched or unbranched infoldings.
Having identified, reconstructed, measured and visualized the morphologies of nuclei from hippocampal neurons, the next objective, in the following chapter, is to investigate the morphological influence on calcium signaling in neuron nuclei.

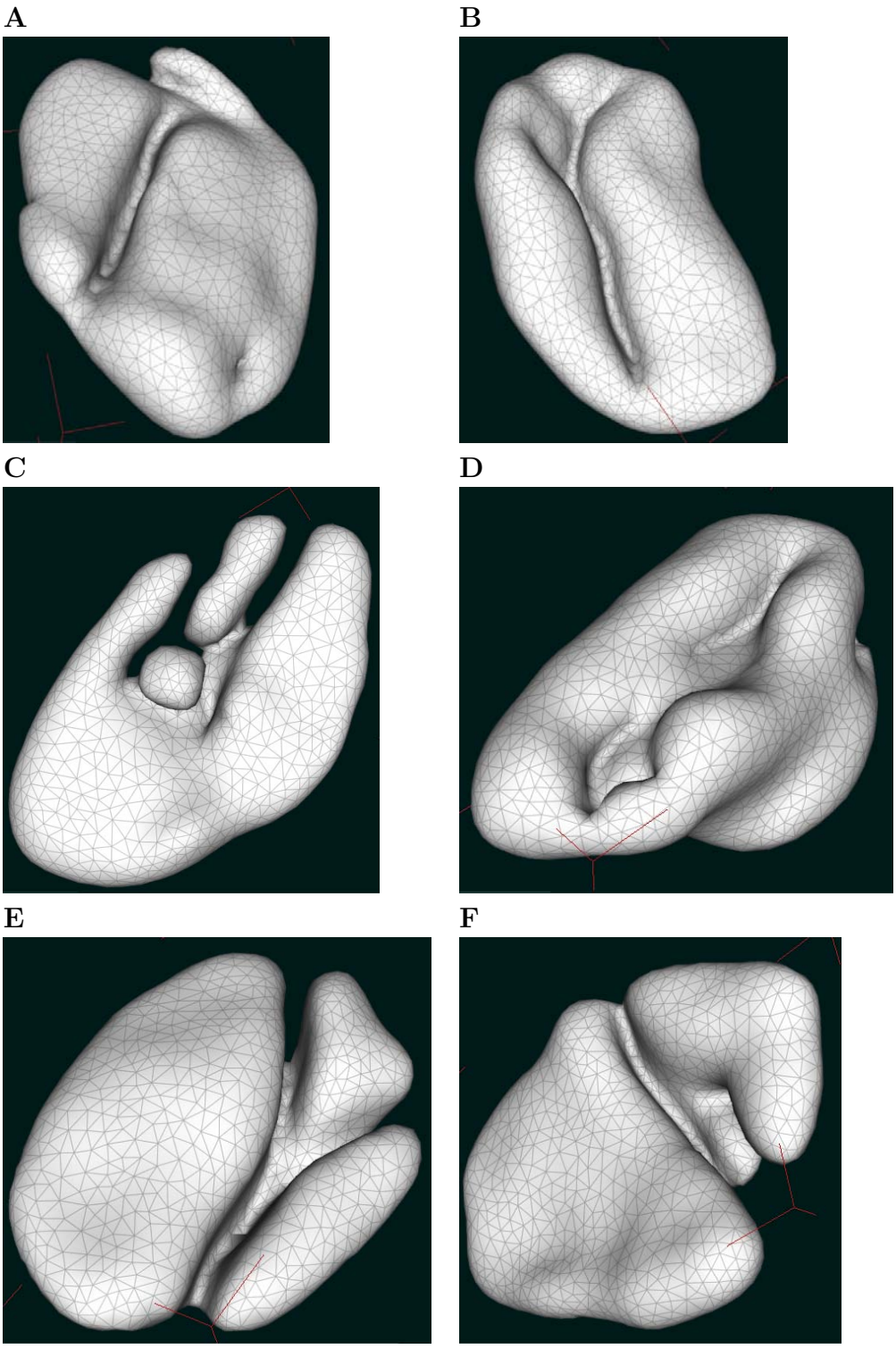


Figure 2.4: *Examples of three-dimensional reconstructions: Infolded nuclei contain single or multiple infoldings.*

# Chapter 3

# Modeling Calcium Signaling in Hippocampal Neurons

## 3.1 Modeling Nuclear Calcium Signaling

In Chapter 1 major communication routes between synapses and cell nucleus were introduced, in particular those, that have controlling influence on the morphology of the nucleus. The pathway that is driven by calcium ion diffusion activates Calmodulin kinase IV. The activated CaM kinase complex then diffuses to sites in the nucleus where further processes, as mentioned in Section 1.3.2, are activated. Since the nuclear membrane allows free passage of calcium ions and no active transportation pathways for calcium are present within the nucleus, the driving physical force for calcium is that of *diffusion*.
The parameters influencing this diffusion process are the *buffering* intensity of calcium ions as well as *diffusion distances* calcium has to pass before reaching vital biological sites. Especially diffusion distances are affected by changes in the form of the nucleus. The stronger the membrane of a nucleus is infolded, the shorter diffusion distances for calcium become, the faster calcium can reach its destination. Another aspect worth investigating is the reorganization of micro organelles in the nucleus due to infolding of nuclear membrane. Organelles in the nucleus are not randomly distributed in the nucleus, but are located in specific regions. It might be vital for the cell to give different calcium signals to different *micro-domains* of the nucleus.
A mathematical model offers the necessary surrounding for systematical evaluations which can not be realized in an experimental setting, where too many parameters are not controllable. Chapter 2 dealt with the extraction of the three dimensional geometry of neuron cell nuclei. Combining these geometries with the mathematical model of nuclear calcium signaling is the task of this chapter.

### 3.1.1 Signal Transduction in the Nucleus

Before deriving the specific equations for the characterization of signal transduction within the nucleus, the axiomatic structure of *thermodynamics of irreversible processes* is outlined. From this the basic equations of diffusion can be formulated and finally – with the specific knowledge of the biological system – adapted to the case of calcium diffusion in the nucleus.
Following [42], the first three (of six) axioms for thermodynamics of irreversible processes can be described in the following way:

**Classical thermodynamics apply:** The presumption of classical thermodynamics, that a system reaches equilibrium after finite time in an isolated state, applies to irreversible processes. An added assumption is, that the scales of the system are small compared to long range exterior forces, so that these can be neglected in the observation of the system.

**Local and instantaneous equilibrium:** It is assumed, that a system that is not

in equilibrium can be subdivided into volume elements – infinitesimally small from a macroscopic point, large enough from a microscopic point (number of molecules in the volume element is large enough for a deterministic claim) – where each volume element is in local equilibrium. This assumption implies continuum theory in thermodynamics of irreversible processes. In non-equilibrium states of such a system, gradients between physical sub-volumes exist, that activate *gradient driven processes* (such as the one of signal transduction in nuclei).

**The balance equations:** Based on empirical facts, the third axiom of thermodynamics for irreversible processes states, that a set of balance equations can be formulated in respect to quantities such as mass, energy, momentum and electrical charge. With the knowledge, that an extensive property changes in a defined sub-volume of the system, when either there is a flux across the borders of this sub-volume or it is produced within the volume, the *flux theorem* can be formulated.

This is done in Section 3.2 since the flux theorem also motivates the numerical procedures chosen for the discretization of the mathematical and physical concepts derived in these sections. With the basics of classical thermodynamics and thermodynamics for irreversible processes, the biological context of nuclear communication as described in Section 1.3 can be formulated with the help of physical theorems.

**Nuclear calcium communication:** Electrical signals transmitted from synapses to the cell soma activate NMDA gated calcium channels through which a calcium wave propagates towards the cell nucleus. This signal propagates through local up-take regions within the cell soma. The following statements, based on neurobiological research, form the physical concepts implemented in the mathematical model for nuclear calcium communication.

1. Local up-take regions in the soma create an integrated signal at the nucleus itself. Therefore, local calcium up-take regions need not be incorporated directly into the mathematical model, but are represented as an integrated calcium wave that enters the nucleus.

2. The calcium wave entering the nucleus is the vital process for information processing within the nucleus (it indirectly activates CBP and CREB, see Section 1.3), the information coming from electrical signals from the dendrites.

3. Calcium can diffuse freely into and out of the nucleus through its nuclear pores located on the nuclear membrane. It is assumed, that no active calcium pumps exist on the inner nuclear membrane, that can create a calcium flux between the endoplasmatic reticulum and the inner volume of the nucleus.

4. Information processing in the nucleus is a concentration gradient driven process, due to the fact that no active pumps exist between cytosol and nucleus (see point 3).

In the given context, signal processing in nuclei can be characterized by *Ficks' law* which states that a mass gradient $\nabla u$ in a force-free system tends to zero in reverse to a flux $\phi$:

$$-\nabla u = \phi \tag{3.1}$$

The law of conservation of mass states

$$-\int_{\partial V} \phi \cdot \vec{n} ds = \int_V \frac{\partial u}{\partial t} dx \tag{3.2}$$

with $V$ a defined volume element, $\partial V$ the boundary surface and $\vec{n}$ the corresponding surface normal, or

$$\frac{\partial u}{\partial t} = -div\phi \tag{3.3}$$

The signal equation for nuclear calcium can therefore be stated as

$$\frac{\partial u}{\partial t} = div(D \cdot \nabla u) \tag{3.4}$$

This equation defines the modeling equation which will be included in the mathematical model in addition with equations that complete a well defined numerical model. In the following section, the diffusion coefficient $D$ that appears in equation (3.4) will be addressed.

### 3.1.2 Diffusion Coefficients in Hippocampal Neuron Nuclei

Signal transduction in hippocampal nuclei relies on changes in calcium ion concentration which are mediated through opening of voltage-gated calcium channels in the membrane, receptor binding which results in channel opening on internal calcium stores or the activation of ryanodine-receptor channels, [1]. Depending on the form in which calcium ions are present, free or in a buffered state, the range of messenger action changes as well as the speed at which a calcium-regulated signal propagates through the cytosol or the nucleus. It has been shown, that free calcium has an effective action range of 0.1 $\mu m$, [1], comparatively small to ranges of buffered calcium systems. In most of the cases calcium is released from internal stores that are sensitive to Inositol 1,4,5-triphosphate ($IP_3$), [64], a second messenger to calcium which increases the effective action range to approximately 24 $\mu m$. In between, the action range for buffered calcium is stated at 5 $\mu m$.
The "trade-off" for longer effective action ranges is a decrease of the diffusion coefficient which is a measure for the signal transduction speed. Local up-take regions for calcium in the cytosol are regulated by immobile or slowly moving second messengers decreasing the effective diffusion coefficient of calcium, [1]. A table listing the

| **Messenger** | **Diffusion coefficient** ($\mu m^2/s$) | **Time scale** ($s$) | **Range** ($\mu m$) |
|---|---|---|---|
| **Calcium** | | | |
| Free ion | 223 | $3 \cdot 10^{-5}$ | 0.1 |
| Buffered | 13 | 1 | 5 |
| **Inositol** | | | |
| 1,4,5-trisphosphate | 280 | 1 | 24 |

Table 3.1: *Estimated range and time scale of messenger action of $Ca^{2+}$ and inositol 1,4,5-trisphosphate, [1].*

diffusion coefficients, time scales and effective action range for the different messengers, based on experiments done in a cytosolic medium from *Xenopus laevis oocytes*, see Table 3.1, is presented in [1]. In the case of calcium signaling in the nucleus, most of the calcium ions are present in buffered form. Depending on the rise of calcium concentration caused by one of the three possibilities mentioned above, the diffusion coefficient for buffered calcium differs in the range of 13 – 65 $\mu m^2/s$. This interval will be used as the choice of the diffusion coefficient for buffered calcium in the mathematical model based on the diffusion equation (3.4) and will be verified by methods presented in Chapter 4.

## 3.2 Numerical Methods for Nuclear Signaling

Putting all physical and biological concepts and all experimental data derived in Section 3.1 about information processing in the nucleus into a mathematical framework is the task of this section. The mathematical theory of *partial differential equations* (PDE) deals with mathematical formulations of an approximation of "reality", e.g. a physical process, an engineering task or observations in nature. Here, this means the mathematical formulation of a model which describes information processing in the nucleus. Following a brief introduction of PDE, a mathematical model for information processing in the nucleus is stated, a numerical approximation of this model is derived and methods for solving this numerical problem are presented.

### 3.2.1 Introduction to Partial Differential Equations

Partial differential equations consist of an unknown function of two or more variables and a number of the unknown function's (partial) derivatives. A k$^{th}$-order partial differential equation can be defined on a subset $\Omega \subset \mathbb{R}^n$ as:

**Definition 1** *An equation of the form*

$$F(D^k u(x), D^{k-1}u(x), \ldots, Du(x), u(x)) = f(x) \quad (x \in \Omega) \tag{3.5}$$

*is called a* k$^{th}$-order partial differential equation*, where*

$$F : \mathbb{R}^{n^k} \times \mathbb{R}^{n^{k-1}} \times \cdots \times \mathbb{R}^n \times \mathbb{R} \times \Omega \to \mathbb{R}$$

*and*

$$u : \Omega \to \mathbb{R}$$

*is the unknown function.*

To solve a partial differential equation means finding a function $u$ that fulfills equation (3.5), perhaps under additional restrictions. Not always does the PDE have a solution. Therefore, a PDE is called *well-posed*, [21], if

1. the PDE has one and only one solution.

2. the solution of the PDE depends continuously on the data in the PDE-problem. This is a requirement of *stability* for the posed problem.

PDE can be categorized into several types of equations, where each type has different properties and different demands must be made for certain types of equations. The general second-order partial differential equation has the form

$$\begin{aligned} a(x,y)u_{xx} + \; & 2b(x,y)u_{xy} + c(x,y)u_{yy} \\ + \; & d(x,y)u_x + e(x,y)u_y + f(x,y)u + g(x,y) = 0. \end{aligned} \tag{3.6}$$

**Convention:** In some cases it is more convenient to write $u_x$ or $u_{xx}$ instead of $\frac{\partial u}{\partial x}$ or $\frac{\partial u}{\partial x^2}$; both expressions are equivalent.

**Definition 2** *Equation (3.6) is called*

1. *an* elliptic *PDE, if*
$$a(c,y)c(x,y) - b^2(x,y) > 0; \tag{3.7}$$
2. *a* hyperbolic *PDE, if*
$$a(x,y)c(x,y) - b^2(x,y) > 0; \tag{3.8}$$
3. *a* parabolic *PDE, if*
$$ac - b^2 = 0 \quad \text{and} \quad Rank \begin{pmatrix} a & b & d \\ b & c & e \end{pmatrix} = 2 \tag{3.9}$$

*for all* $(x,y) \in \Omega \subset \mathbb{R}^{\not\in}$.

Not every PDE can be assigned to one of the three types of partial differential equations. Yet recalling the PDE equation (3.4)

$$\frac{\partial u}{\partial t} = \text{div}(D\nabla u), \tag{3.10}$$

it can be assigned to the type of parabolic PDE. To demonstrate this, one can consider the diffusion equation (3.10) in one dimension and in time. It has the form (following expression (3.6))

$$1 \cdot u_{xx} + 2 \cdot 1 \cdot u_{xt} + 1 \cdot u_{tt} + 1 \cdot u_t = 0 \tag{3.11}$$

with

$$a(x,t) = 1, \quad b(x,t) = 1, \quad c(x,t) = 1, \quad e(x,t) = 1,$$
$$d(x,t) = f(x,t) = g(x,t) = 0$$

and therefore

$$ac - b^2 = 0 \quad \text{and} \quad Rank \begin{pmatrix} 1 & 1 & 0 \\ 1 & 1 & 1 \end{pmatrix} = 2.$$

Parabolic PDE are sufficiently characterized, meaning they fulfill the above well-posedness criteria, when *initial value* and *boundary conditions* are defined. In this case the set of equations

$$\frac{\partial u(x,t)}{\partial t} = div(D \cdot \nabla u(x,t)) \qquad \text{on } \Omega \subset \mathbb{R}^{3\times 1} \tag{3.12}$$
$$u(x,t_0) = u_0(x) \tag{3.13}$$
$$u(x,t) = f(t) \qquad \text{on } \partial\Omega \subset \mathbb{R}^3 \tag{3.14}$$

are solved by one and only one function $u$. The definition of the boundary function $f(t)$ in a biologically sensible way will be addressed in Section 3.4. The following section gives rise to numerical methods for the implementation of the set of equations (3.12) – (3.14).

### 3.2.2 Discretization of the Signaling Equation

Where the theory of partial differential equations forms a link between "reality", i.e. biological and physical observations, and the "mathematical reality", numerical mathematics forms links between mathematical models and numerical models. And where a mathematical model is a simplification or an abstraction of observational reality, a numerical model is a finite mathematical description of a mathematical model. The last statement is the definition of *discretization*: Turning a mathematical model in an infinite subset of $\mathbb{R}^n$ into a finite problem on a discrete subset $\Omega_h \subset \mathbb{R}^n$. In order to derive a numerical discretization of the model defined by (3.12) – (3.14) some definitions are made.

**Definition 3** *A* discretization *of a region $\Omega \subset \mathbb{R}^n$ is the replacement of $\Omega$ with a finite set $\Omega_h$ made up of points $\omega_i \in \Omega$. The set $\Omega_h$ is called a grid (or triangulation) of $\Omega$.*

**Definition 4** *A* triangulation *$\tau := \{T_1, \ldots, T_t\}$ with triangles $T_i$ of $\Omega$ is called valid, if the following points are fulfilled, [29]:*

1. *$T_i$ are open triangles.*
2. *$T_i \cap T_j = \emptyset$ for $i \neq j$.*
3. *$\bigcup_{1 \leq i \leq t} \bar{T}_i = \bar{\Omega}$.*
4. *For $i \neq j$ there are three possibilities for $\bar{T}_i \cap \bar{T}_j$:*
   (a) *$\bar{T}_i \cap \bar{T}_j = \emptyset$.*
   (b) *$\bar{T}_i \cap \bar{T}_j$ contains one common side.*
   (c) *$\bar{T}_i \cap \bar{T}_j$ contains one common corner of element $T_i$ and $T_j$.*

After having defined a discrete form of the region $\Omega$ in which the PDE is to be solved, the discretization being a triangulation of $\Omega$ with its *nodes* (corners of triangle elements), the PDE can be solved at the nodes of the triangulation. Here, various methods exist for discretization of a PDE on a discrete subset $\Omega_h$. One concept, the method of *finite differences*, relies on the approximation of derivatives of $u$ in the PDE by the difference quotient (in one dimension)

$$u'(x_i) \approx \frac{u_{i+h} - u_{i-h}}{2h}$$

where $h$ is the distance between neighboring nodes in an equidistant grid and the approximation of $u$ with a numerical solution $u_h$, where $u_h$ is element of a defined space, e.g. $u_h \in C^2(\bar{\Omega})$. As long as one deals with simple and structured geometries the finite differences method is easy to apply, yet hard to realize on unstructured geometries, which is the case for hippocampal nuclei.

More sophisticated methods that can handle complex geometries and have positive physical properties demand the search for a numerical solution of the PDE in spaces other than, say $C^2(\bar{\Omega})$. The theory of *finite elements* and *finite volumes* offers, with concepts from functional analysis, "weak" solutions of $u$ in *Sobolev*-spaces $H^k(\Omega)$, [29]. The necessary basics of functional analysis for discussing finite volume methods are found in [29]. The finite volume discretization of (3.12) – (3.14) is as follows.

**Finite Volume discretization:**

The "weak" formulation of

$$\frac{\partial u(x,t)}{\partial t} = div(D \cdot \nabla u(x,t)) \tag{3.15}$$

is derived by integrating both sides of the equation:

$$\int_\Omega \frac{\partial u}{\partial t} = \int_\Omega div(D \cdot \nabla u). \tag{3.16}$$

Applying *Gauss's theorem*

$$\int_V divF dV = \int_S F \cdot \vec{n} dS \tag{3.17}$$

with $F$ a continuously differentiable vector field, $V$ a volume element, $S$ a surface element and $\vec{n}$ the corresponding normal, which states that the change in volume $V$ is equal to the flux over the volume surface, equation (3.16) can be written as

$$\frac{\partial}{\partial t} \int_\Omega u = \int_{\partial\Omega} D \cdot \nabla u \cdot \vec{n} \tag{3.18}$$

with $\partial\Omega$ the bounding surface of $\Omega$. Now, in finite volume discretization, a *dual*

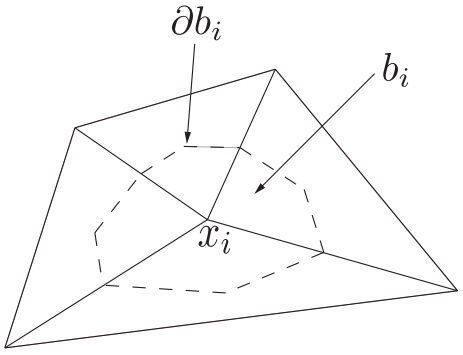


Figure 3.1: *Constructing a dual grid: Centers of mass of each element are linked together to form a dual mesh.*

*mesh* is constructed from the original triangulation. Here too, different methods

are applicable, one of them is linking center points of edges and centers of mass of the triangles to an integration volume $b_i$, Figure 3.1. Thus, a dual triangulation $\bigcup_{i=1}^{m} b_i = \Omega_h, \quad m = \#$ *volume elements* is created, which is valid in the sense of Definition 4.
The right hand side of equation (3.18) can now be written as

$$\int_{\partial\Omega} D \cdot \nabla u \cdot \vec{n} = \sum_{i=1}^{m} \int_{\partial b_i} D \cdot \nabla u \cdot \vec{n} \tag{3.19}$$

For reasons of stability the preferred choice is *implicit time discretization*:

$$\frac{\partial u}{\partial t} = \frac{u(t_{k+1}) - u(t_k)}{\Delta t} + O(h). \tag{3.20}$$

The left hand side of equation (3.18) can be approximated by the above difference quotient. The approximated weak formulation on a single box element $b_i$ with equations (3.18) – (3.20) yields

$$\int_{b_i} u(t_{k+1}, x_i)dx - \int_{b_i} u(t_k, x_i)dx = \Delta t \int_{\partial b_i} \sum_j u(t_{k+1}, x_j) D \nabla \xi_j(\gamma) \vec{n}_i d\gamma. \tag{3.21}$$

Numerically, $\nabla u$ is approximated by a choice of *Ansatzfunctions* $\nabla \xi_j$, e.g. linear

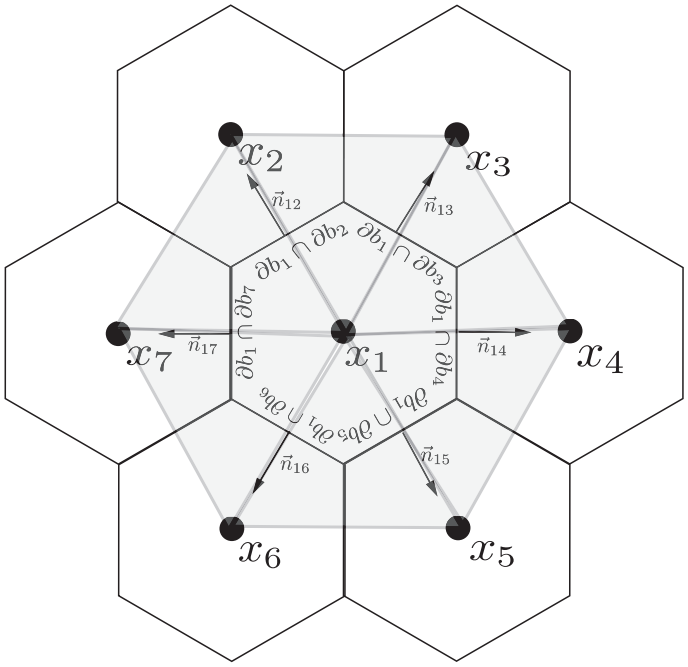


Figure 3.2: *Regular dual grid: Depiction of numerical discretization components, in transparent gray the triangulation, with its hexagonal dual grid (white).*

finite elements, [29], defined at integration points $\gamma$. In a last approximation step, the integrals in equation (3.21) can be approximated numerically:

$$|b_i|\,(u(t_{k+1},x_i)-u(t_k,x_i)) = \Delta t \sum_{j,l} |\partial b_i \cap \partial b_l|\, u(t_{k+1},x_j) D \cdot \nabla \xi_j(x_{il}) \cdot \vec{n}_{il} \quad (3.22)$$

Figure 3.2 shows the schematics of the components in equation (3.22) on a regular dual grid. The finite volume discretization is applicable to PDE embedded in highly unstructured grids, since it is based on grid independent mathematical theory. An additional advantage to finite differences or finite elements is the fact that the physical flux theorem is valid for each element of the dual grid $B_h := \bigcup_i b_i$, [42], (in three dimensions $b_i$ is a tetrahedra). Equation (3.22) poses a system of equations in the variables $x_i$ for each time step $\Delta t$. Finally, solving the diffusion equation for signal processing in nuclei is reduced to solving a system of equations. The method for solving the system defined by (3.22) is addressed in the following section.

### 3.2.3 Solving the Discrete Model Equation

During the search for mathematical solving-techniques of the discretized PDE, two major guidelines are that the PDE is defined on random unstructured grids and the model is three-dimensional. The first point calls for a solving method that is highly independent of grid structure, the second poses large computational effort that needs to be addressed for the solution to be calculated in a reasonable time span.
The well established concepts of *multi-grid methods* address exactly these points. The general approach of multi-grid methods outlines the mathematical concept in a problem-independent way and contains various multi-grid components that are then chosen according to the individual problem. Following an introduction of the multi-grid approach, multi-grid components for the solution of the model equation (3.12) – (3.14) in the discrete form (3.22) are given.

**Multi-grid method:**

Equation (3.22) represents a system of equations, the number of equations defined by the number of grid nodes and can be written as

$$K \cdot u = f \quad (3.23)$$

where $K$ is the system's matrix defined by the applied finite volume discretization scheme, $u$ the solution and $f$ a predefined – in this case predefined by the biological context – right hand side. For the multi-grid method one defines a hierarchy of grids $\Omega_l \supset \Omega_{l-1} \ldots \supset \Omega_0$ and on each grid level a grid-dependent system of equations $K_l \cdot u_l = f_l$ is formulated. Now, instead of calculating an approximation of the solution $u$ on one grid, multi-grid methods *smooth* the system's *defect* until the

algorithm reaches a *base level* on which the defect is calculated. Then, moving up again in the grid hierarchy the solution of the PDE is corrected iteratively on each grid level until the highest level is reached.
For this method, three components need to be chosen in a problem-dependent matter:

1. **Smoother:** Classical iterations, such as the Jacobi-iteration can be used as a smoother $S$ in multi-grid methods. They have the effect of eliminating high frequency parts of the defect at each grid transfer and thus "smoothing" the defect.

2. **Restrictor:** A restriction procedure $r$ defines the routine with which all necessary components can be formulated downwardly in grid hierarchy.

3. **Prolongator:** The prolongation $p$ defines an inverse routine to the restriction, with which all components are transferred upwards in the grid hierarchy.

Algorithm 1 shows the scheme of a multi-grid procedure applied on level depth $l$.

**Algorithm 1** *Given a system of equations by $K_l u_l = f_l$ derived from finite volume discretization on grid level $l$, the multi-grid method* `mgm` *can be formulated as*

```
mgm(l,u,b) {
if l == 0 then u=solve(l, K_0, d)
else {
u=smooth(l, nu_1, u, b);
d=defect(l, u, b);
d=restriction(l, d);
v=0;
for i = 0 to γ do mgm(l-1,v,d);
u=u-prolongate(l − 1, v);
u=smooth(l, nu_2, u, b);
}
}
```

As a smoother, the usage of the Gauß-Seidel smoother showed to be ideal for the implementation of the described model.

### Gauß-Seidel smoother:

In order to solve a given equation system

$$Ku = f \tag{3.24}$$

with $K$ being a sparse matrix the objective is to invert $K$. Since an exact inversion of $K$ surpasses computational power quickly (the model is three-dimensional with a

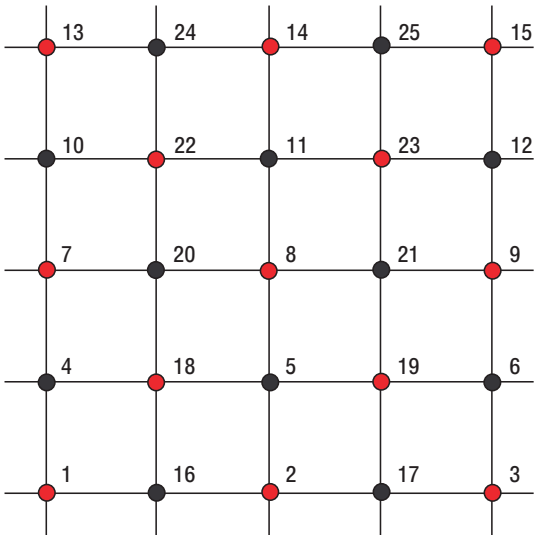


Figure 3.3: *Checkerboard ordering of grid nodes.*

very large number of grid nodes) an approximation of $K^{-1}$, namely a *pre-conditioner* $M^{-1}$ is chosen. This matrix $M$ must be chosen in such a way, that inverting it is simple in computational terms.
The Gauß-Seidel method proposes

$$M = L + D, \tag{3.25}$$

with $L$ a strictly lower sparse matrix and $D$ a diagonal matrix. The iteration rule given by this method is

$$u^{(i+1)} = u^{(i)} - M^{-1}(Ku^{(i)} - f) \tag{3.26}$$

and offers highly effective smoothing of the defect $d := Ku^{(i)} - f$ for symmetric and positive definite matrices, [28]. Since the Gauß-Seidel smoothing method is dependent on the numbering of the grid nodes, a choice has to be made. In [28] different numbering schemes are introduced. In order to leave the opportunity to parallelize the discretized model equations, it is necessary to apply *red-black* ordering (also known as *checkerboard* ordering), see Figure 3.3.

**Time solver:**

A backward difference formula can be applied by the time solver. Approximation of $\frac{\partial u}{\partial t}$ is done with

$$\frac{\partial u}{\partial t} \approx \frac{u(t + \Delta t) - u(t)}{\Delta t} \tag{3.27}$$

and is effectively implemented using a nonlinear Newton-solver. The implementation of the discrete model is done in the simulation software package *UG*, which is content of the next section.

## 3.3 The Simulation Software *UG*

As the development of a mathematical model for nuclear information processing in Sections 3.1–3.2 already shows, the simulation of processes, particularly in neurosciences, makes use of the theory of partial differential equations which are solved numerically with the well established concept of multi-grid methods. Next to mathematics and applicational sciences, a third discipline is involved in the successful realization of scientific simulations, *scientific computing*. Implementation of all methods used in Section 3.1 and 3.2 is a very large task; a task realized in the software development project *UG*.
Since *UG* is implemented in such a way to address flexible data structures for highly unstructured geometries and simulations in three dimensions and incorporates a wide range of solving techniques based on the multi-grid concept, it presents itself very useful for the realization of the simulation of information processing in nuclei of hippocampal neurons.

### 3.3.1 The *UG* Concept

The *UG* project began in the early nineties with the goal of developing a software environment to solve partial differential equations. While the problems to be solved are specific, the basic structure of *UG* was to remain problem independent and is therefore a platform for varying problem definitions. The *UG* hierarchy can be divided into three levels, the *UG kernel*, containing problem independent library components, the *problem class* level, in which discretizations, solvers etc. for specific PDE are included and into the *application* level, where concrete problems are defined. Schematics of this structure is visualized in Figure 3.4, adapted from [8].
*UG*'s data types can be categorized into *geometric* data types and *algebraic* data types. Geometric data types control large and complex grids, where the algebraic ones are used to handle algebraic structures needed to store vectors and matrices, as well as numerical solutions of the underlying PDE-problem. Basic data types of *UG* are listed in Table 3.2.

### 3.3.2 Implementing the Nucleus Model

Implementing the "nucleus model" takes place on the problem class level, where the discretization of Section 3.2 is made available and on the application level where grids from either test geometries or from real 3D-data of hippocampal neuron nuclei are included. Both solver- and grid-procedures are defined in script files that are executed by *UG*.

#### *UG*-script files:

*UG* script files are written in a *UG* scripting language, offering the user the possibility to make fix definitions of problem specifications, solver specifications, evaluation

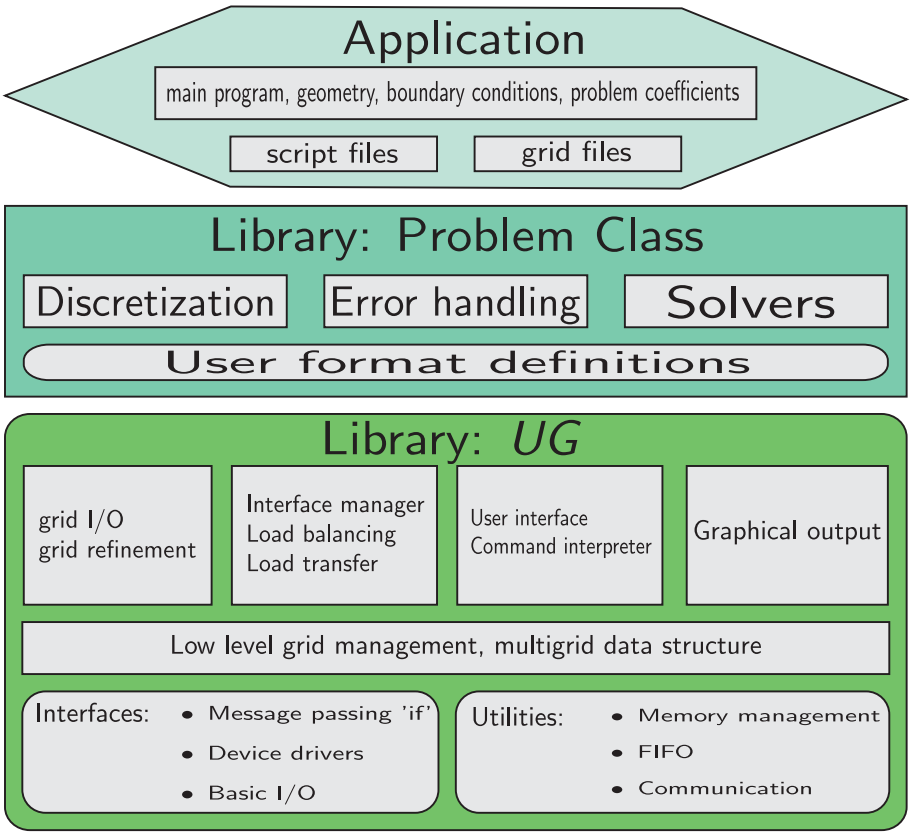


Figure 3.4: *UG-structure: The simulation environment is designed in a hierarchical order, from library to application level.*

| | Data Type | Description |
|---|---|---|
| Geometric Data Types | `MULTIGRID` | Bundles all the information of problem |
| | `GRID` | Contains data concerning one grid level |
| | `ELEMENT` | Component of finite volume grid |
| | `NODE` | Grid level dependent mesh node data |
| | `VERTEX` | Grid level independent mesh node data |
| | `LINK` | List of neighbor nodes |
| | `EDGE` | Connection between two nodes |
| Algebraic Data Types | `VECTOR` | Data storage for defined geometric objects |
| | `MATRIX` | Matrix data structure |

Table 3.2: *Important basic data types of UG: Data types in UG are separated into geometric and algebraic data types.*

tool commands and also graphical output specifications.

### 3.3.3 Grid Integration for Modeling

*UG* contains two different domain models, meaning two different ways to include the geometry of the object on which the PDE is to be solved. These can be user-defined objects in two or three dimensions, structured or unstructured grids, test geometries or ones retrieved from microscopy data.
Where the *standard domain model* is useful for implementing structured geometries, the *linear grid model* (lgm) is designed to handle unstructured geometries that cannot be expressed in mathematical function terms.

**Standard domain model:**

The standard domain model requires the definition of the geometry's borders with piecewise continuous functions, and can also contain user-defined inner grid nodes. As the requirements of the standard domain model show, it is easy to use when dealing with geometric objects that can be defined by mathematical functions, such as circles in two dimensions, cubes or spheres in three dimensions.
Having defined the border segments of a geometric object with the standard domain model, it can be included into *UG*, where refinement tools build and refine a volumetric grid that, with refinement, converges to the borders of the standard domain model. The geometry-border is defined together with the problem in *UG*, a sample excerpt of the definition of a unit circle in two dimensions – made up of two segments, an upper halve circle and lower halve circle – is shown in Figure 3.5.
Furthermore, boundary and inner nodes of the volume grid can be defined in the *UG*-script file (see Figure 3.6 for an example of a nucleus test-geometry further addressed in Section 3.6). Here, boundary nodes are defined in their relative position to a given boundary segment, inner nodes are defined by their absolute coordinates.

**Linear grid model (lgm):**

When the boundary of an object is random, i.e. abides to no piecewise continuous mathematical function, it is advantageous to use the lgm-model as a means of describing the surface of an object. Rather than implementing the boundary information in *UG*, it is imported from an `.lgm`-file and included into the multi-grid components of *UG*.
A linear grid model file contains point-, line-, and triangle-information about the surface area of an object, describing the surface of a three dimensional object as a triangular grid with defined triangle orientation. An example `.lgm`-file is shown in Figure 3.7. The lgm-path requires the generation of a separate volumetric grid which is stored in a `.ng`-file. This filetype contains all volumetric information of the geometry, in three dimensions this would be point information and the connectivity

```
static INT circleBoundaryUpper (void *data, DOUBLE *param, DOUBLE *result)
{
    DOUBLE lambda;

    /* retrieve parameter value */
    lambda = param[0];

    /* check range */
    if ((lambda<0.0)||(lambda>1.0)) return(1);

    /* fill result */
    result[0] = cos(PI*lambda); /* x */ /* PI defined in misc.h */
    result[1] = sin(PI*lambda); /* y */

    /* return ok */
    return(0);
}

static INT circleBoundaryLower (void *data, DOUBLE *param, DOUBLE *result)
{
    DOUBLE lambda;

    /* retrieve parameter value */
    lambda = param[0];

    /* check range */
    if ((lambda<0.0)||(lambda>1.0)) return(1);

    /* fill result */
    result[0] = cos(PI+PI*lambda); /* x */ /* PI defined in misc.h */
    result[1] = sin(PI+PI*lambda); /* y */

    /* return ok */
    return(0);
}
```

Figure 3.5: *Example of a standard domain implementation of a unit-circle.*

```
invnucleusmesh = "{
    # insert nodes:
    # patch1: 0/1.4 (min/max lambda): 0.0  +  (1.4 - 0.0)*n/4;
    bn 0  0.35000; # corner  9, upper right segment of nucleus (lambda_min + 1/4 delta lambda)
    bn 0  0.70000; # corner 10, upper right segment of nucleus (lambda_min + 2/4 delta lambda)
    bn 0  1.05000; # corner 11, upper right segment of nucleus (lambda_min + 3/4 delta lambda)

    # patch2: 1.5708/3.14159 (min/max lambda):
    bn 1  2.44159; # corner 12, right begin of invagination (lambda_min + 1/2 delta lambda)

    # patch3: 0.30/0.88545 (min/max lambda): 0.30   + (0.88545 - 0.30)*n/5;
    bn 2  0.41709; # corner  , right boundary of invagination (lambda_min + 1/5 delta lambda)
    bn 2  0.53418; # corner  , right boundary of invagination (lambda_min + 2/5 delta lambda)
    bn 2  0.65127; # corner  , right boundary of invagination (lambda_min + 3/5 delta lambda)
    bn 2  0.76836; # corner  , right boundary of invagination (lambda_min + 4/5 delta lambda)

    # patch4: 3.14159/6.28319 # (min/max lambda):
    bn 3  4.71239; # corner  , lower half circle in invagination

    # patch5: 0.30/0.88545 (min/max lambda): 0.30   + (0.88545 - 0.30)*n/5;
    bn 4  0.41709; # corner  , right boundary of invagination (lambda_min + 1/5 delta lambda)
    bn 4  0.53418; # corner  , right boundary of invagination (lambda_min + 2/5 delta lambda)
    bn 4  0.65127; # corner  , right boundary of invagination (lambda_min + 3/5 delta lambda)
    bn 4  0.76836; # corner  , right boundary of invagination (lambda_min + 4/5 delta lambda)

    # patch6: 0/1.5708 (min/max lambda):
    bn 5 0.78540; # corner  , left begin of invagination (1/2 delta lambda)

    # patch7: 1.74159/3.14159 (min/max lambda): 1.74159  +  (3.14159 - 1.74159)*n/4;
    bn 6  2.09159; # corner  , upper left  segment of nucleus (lambda_min + 1/4 delta lambda)
    bn 6  2.44159; # corner  , upper left  segment of nucleus (lambda_min + 2/4 delta lambda)
    bn 6  2.79159; # corner  , upper left  segment of nucleus (lambda_min + 3/4 delta lambda)

    # patch8: 3.14159/6.28319 (min/max lambda): 3.14159  +  (6.28319 - 3.14159)*n/8;
    bn 7  3.53429; # corner  , lower half circle of nucleus (lambda_min + 1/8 delta lambda)
    bn 7  3.92699; # corner  , lower half circle of nucleus (lambda_min + 2/8 delta lambda)
    bn 7  4.31969; # corner  , lower half circle of nucleus (lambda_min + 3/8 delta lambda)
    bn 7  4.71239; # corner  , lower half circle of nucleus (lambda_min + 4/8 delta lambda)
    bn 7  5.10509; # corner  , lower half circle of nucleus (lambda_min + 5/8 delta lambda)
    bn 7  5.49779; # corner  , lower half circle of nucleus (lambda_min + 6/8 delta lambda)
    bn 7  5.89049; # corner  , lower half circle of nucleus (lambda_min + 7/8 delta lambda)

    # inner nodes:
    in  -0.350  0.750; in  0.350  0.750;
    in  -0.480  0.550; in  0.480  0.550;
    in  -0.500  0.300; in  0.500  0.300;
    in  -0.480  0.000; in  0.480  0.000;
    in  -0.350 -0.320; in  0.350 -0.320;
    in  -0.120 -0.420; in  0.120 -0.420;
```

Figure 3.6: *Example of a standard domain definition of a volume grid of an infolded circle on script level.*

between these points that form a tetrahedral mesh. Furthermore, the information about points, lines, and sides being either inner or boundary components is stored and passed on to the grid manager of *UG*.

**Generating a `.ng`-file:**

There exists a number of filters, developed by A. Hauser (SiT), that were designed in such a way to incorporate the CAD-meshing tool ICEM CFD into *UG*. A list of useful filters are given in Table 3.3. ICEM CFD's volumetric data is stored in

| Name of filter | Description |
|---|---|
| `lgm2tetin` | Translates `.lgm`-file into the ICEM CFD format `.tin` |
| `projectPts` | Projects inexact boundary points of ICEMs `.vrt`-file onto lgm surface |
| `lgm+star2ng` | Translates the ICEM CFD volumetric files together with the lgm into a `.ng`-file |

Table 3.3: *Filters for UG-grids: Three main filters allow the use of ICEM CFD for volume grid generation and import of these grid into UG as lgm- and ng-files.*

the file-formats `vrt`, `cel` and `inp`, which contain point and tetrahedral information. Several solvers for the volumetric meshing are offered by ICEM CFD. The filters from Table 3.3 are written to incorporate the "`star-CD`"-solver of ICEM CFD. For more information, see [2].
The filter `lgm2tetin` is used to translate the linear grid model into the ICEM CFD geometry file-format `tin`. After having generated a volumetric grid, the data is written into the above files using the `star-CD` solver. Since ICEM CFD lacks the necessary precision in floating point representation of boundary point data, the filter `projectPts` projects the boundary points onto the lgm-surface so *UG* can identify them as such. In a third step, `lgm+star2ng` translates `vrt`, `cel`, `inp`, and `lgm` into one `ng`-file. Both `lgm` and `ng` carry all the information for *UG* to incorporate the surface- and volume-model into the multi-grid solving process of calcium diffusion in hippocampal nuclei.

```
# Domain-Info
name = unitsquare
problemname = unitsquare
convex = 1

# Unit-Info
unit 1 DUMMY

#Line-Info
line 0: points: 0 1;
line 1: points: 1 2;
line 2: points: 2 3;
line 3: points: 3 0;
line 4: points: 4 5;
line 5: points: 5 6;
line 6: points: 6 7;
line 7: points: 7 4;
line 8: points: 0 4;
line 9: points: 1 5;
line 10: points: 2 6;
line 11: points: 3 7;


#Surface-Info
surface 0: left=1; right=0; points: 0 1 2 3; lines: 0 1 2 3; triangles: 0 1 2; 0 2 3;
surface 1: left=1; right=0; points: 1 0 4 5; lines: 0 8 4 9; triangles: 0 4 5; 1 0 5;
surface 2: left=1; right=0; points: 1 2 5 6; lines: 1 10 5 9; triangles: 1 5 6; 1 6 2;
surface 3: left=1; right=0; points: 2 3 6 7; lines: 2 11 6 10; triangles: 2 6 7; 2 7 3;
surface 4: left=1; right=0; points: 0 3 4 7; lines: 3 8 7 11; triangles: 0 3 7; 0 7 4;
surface 5: left=1; right=0; points: 4 5 6 7; lines: 4 5 6 7; triangles: 4 7 6; 4 6 5;

#Point-Info
0 0 0;
1 0 0;
1 1 0;
0 1 0;
0 0 1;
1 0 1;
1 1 1;
0 1 1;
```

Figure 3.7: *Example of an* `lgm`*-file of a unitsquare.*

## 3.4 Cytosolic Calcium Transients as Boundary-Conditions for PDE

To complete the mathematical model based on a diffusion equation for calcium ions in the nucleus combined with test- or real 3D-geometries of neuron nuclei, the boundary conditions for the diffusion equation need to be defined. Rather than applying mathematical boundary conditions, it is necessary to model the effect of local uptake regions for calcium from cytosolic calcium stores which generate an integrated calcium signal to the nucleus. The cytosolic calcium signal cannot be modeled as direct onset of calcium concentration to a given concentration value followed by an instantaneous offset to base level calcium concentration. Experimental data shows the variety of cytosolic calcium transients (CCTs), which are differentiable by parameters such as duration, amplitude and frequency of the CCTs.
In order to represent these parameters in a mathematical model, the general structure of a cytosolic calcium transient was investigated and modeled by mathematical functions. These mathematical representations stand for either single or multiple CCTs which can be manipulated systematically for evaluation purposes and prognoses of the behavior of nuclei under different stimuli. An additional development finally offers the possibility of incorporating experimentally measured cytosolic calcium transients into *UG* for modeling signal processing in nuclei on real 3D-geometries as well as with real CCT data as boundary conditions for the model.
A *C++* program was written to generate the above types of CCTs with a set of parameters with which the form of calcium transients can be manipulated. The structure of the program *Burstfunction* is such, that it can be directly exported to *UG*.

### 3.4.1 Single CCT Bursts

The integrated signal of a single burst in a cytosolic calcium transient can be described by a rapid linear increase of calcium concentration followed by an exponential decay. Figure 3.8, (A), taken from [31], depicts a typical spike train originating in the cytosol and entering the nucleus. The parameters for manipulating the form of a single CCT are listed in Table 3.4. The mathematical function is formulated as a

| Parameter | Description |
|---|---|
| `tau` | Regulates form of exponential decay |
| `t_stop` | Defines peak time of CCT |
| `y_intercept` | Shift in base level of calcium concentration |
| `Amp` | Amplitude of CCT |

Table 3.4: *Parameters for generating and manipulating single cytosolic calcium transients.*

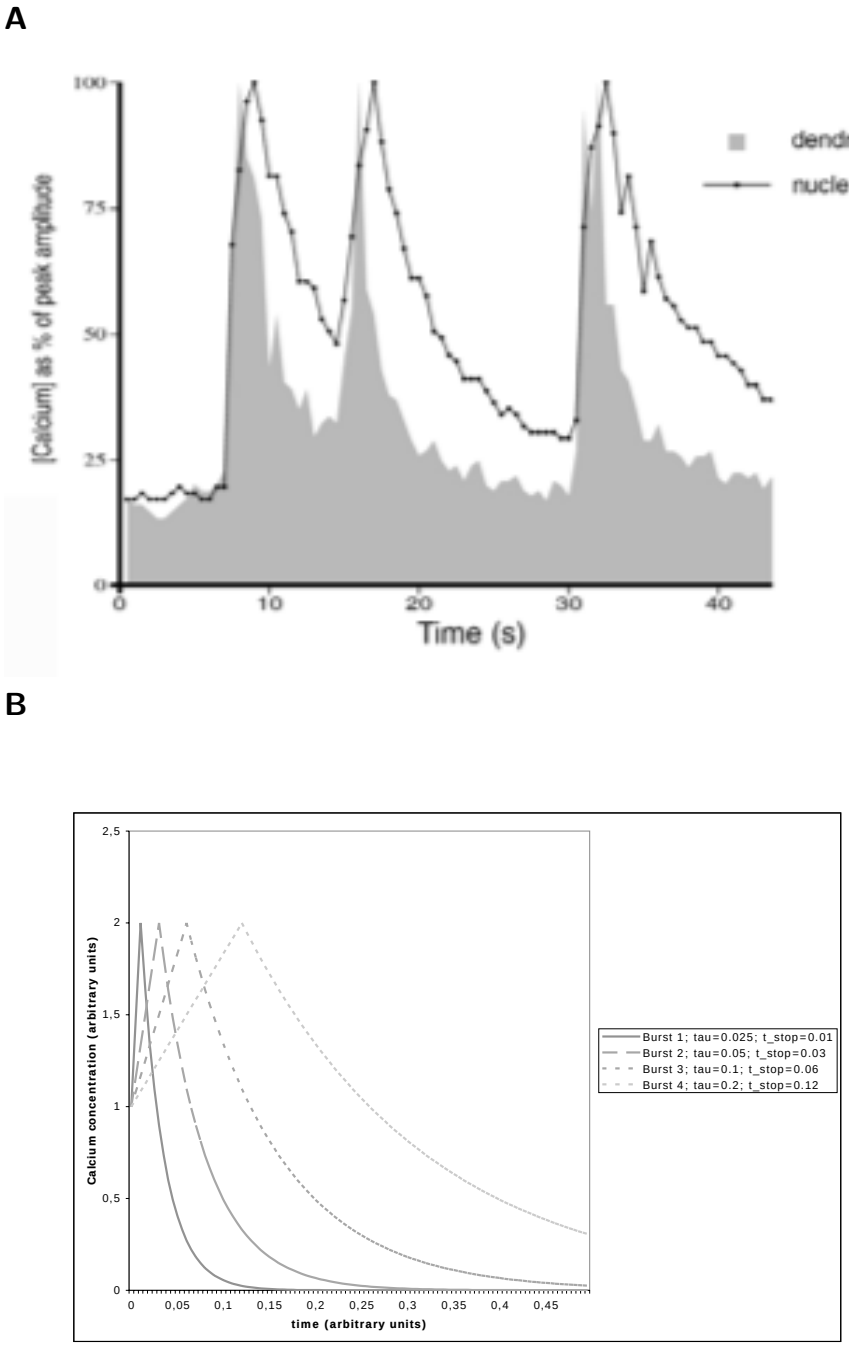


Figure 3.8: *Cytosolic calcium transients: (A) Experimentally measured dendritic and nuclear calcium transients. (B) Mathematically designed single CCTs.*

piecewise continuous function with linear rise and exponential decay:

$$CCT_s := \begin{cases} r \cdot x + y_{intercept} & \text{for } x \leq p \\ r \cdot p + y_{intercept} \cdot \exp(\frac{p-x}{\tau}) & \text{for } x > p \end{cases} \tag{3.28}$$

with $x$ the simulation time in the model, $p$ the peak time regulated by `t_stop` and $\tau$ the decay parameter `tau`. In Figure 3.8, (B), mathematically generated burst functions, simulating different types of single cytosolic calcium transients are plotted. Depending on the parameters, transient length and rise/decay speed can be influenced. Not shown is the regulation of the amplitude by `Amp`.
An adjustable implementation of single calcium bursts is vital for systematic investigations on the effect of different boundary inputs for the signaling results in nuclei. The present implementation is designed to be responsive to the variance in CCTs in biological systems with changing burst durations, intensities and rise/decay speeds.

### 3.4.2 Multiple CCT Bursts

A further mathematical transient design was constructed to mimic the repetition of single bursts within a released calcium signal. As can be seen in Figure 3.8, (A), multiple bursts of increasing calcium concentration in the cytosol followed by a rapid exponential decay, form a calcium signal that enters the nucleus. Next to amplitude and single burst duration, the frequency of the calcium signal oscillation becomes a further parameter in the control of mathematically generated calcium transients. This parameter, in the program *Burstfunction*, is defined as parameter $k$, the frequency regulator.
The multiple burst function is based on function (3.28) and can be written as:

$$CCT_m := \begin{cases} r \cdot x + y_{intercept} & \text{for } s \cdot \pi < x \leq s \cdot \pi + p \\ r \cdot p + y_{intercept} \cdot \exp(\frac{p-x}{\tau}) & \text{for } s \cdot \pi + p < x \leq (s+1) \cdot \pi \end{cases} \tag{3.29}$$

with $s = 0, 1, 2, \ldots$ and period length $\pi$.
Figures 3.9 and 3.10 contain examples of different multiple calcium oscillations, that can be used as boundary conditions in the mathematical model. To test the effect of spike trains with different frequencies, the frequency is adaptable, see Figure 3.9, as well as the duration of single bursts, see Figure 3.10.
Not only spike trains with the distance between single burst large enough for the signal to decay to base level concentration are observed experimentally. If the induction of CCTs is strong, a rapid spike train can be measured, where the bursting intervals are so small, that the signal does not retract to base level but is overlapped by the following burst, see Figure 3.10, (E).

### 3.4.3 Importing Experimentally Recorded CCTs

While the two boundary functions of single and multiple burst sequences are mathematical functions, which approximate experimentally recorded cytosolic calcium

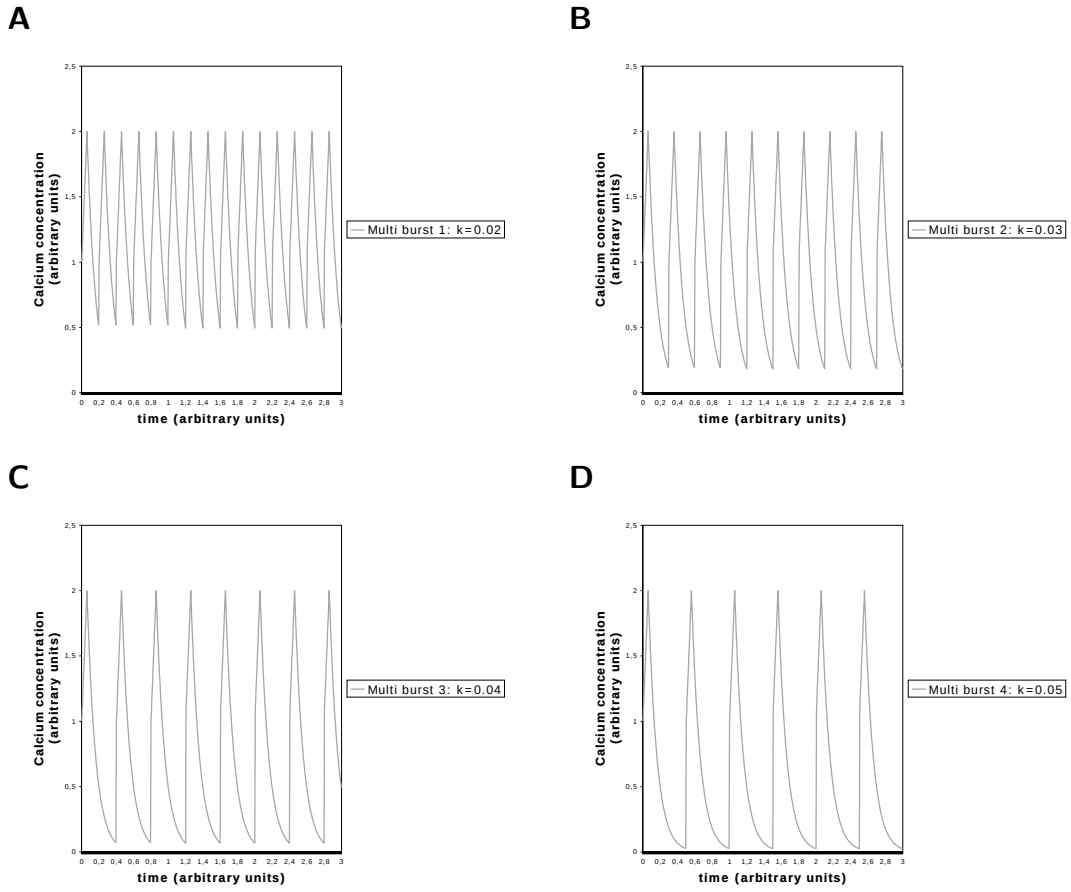


Figure 3.9: *Mathematically generated multiple CCTs: (A)-(D) Variation of the frequency parameter generates different types of multiple CCTs.*

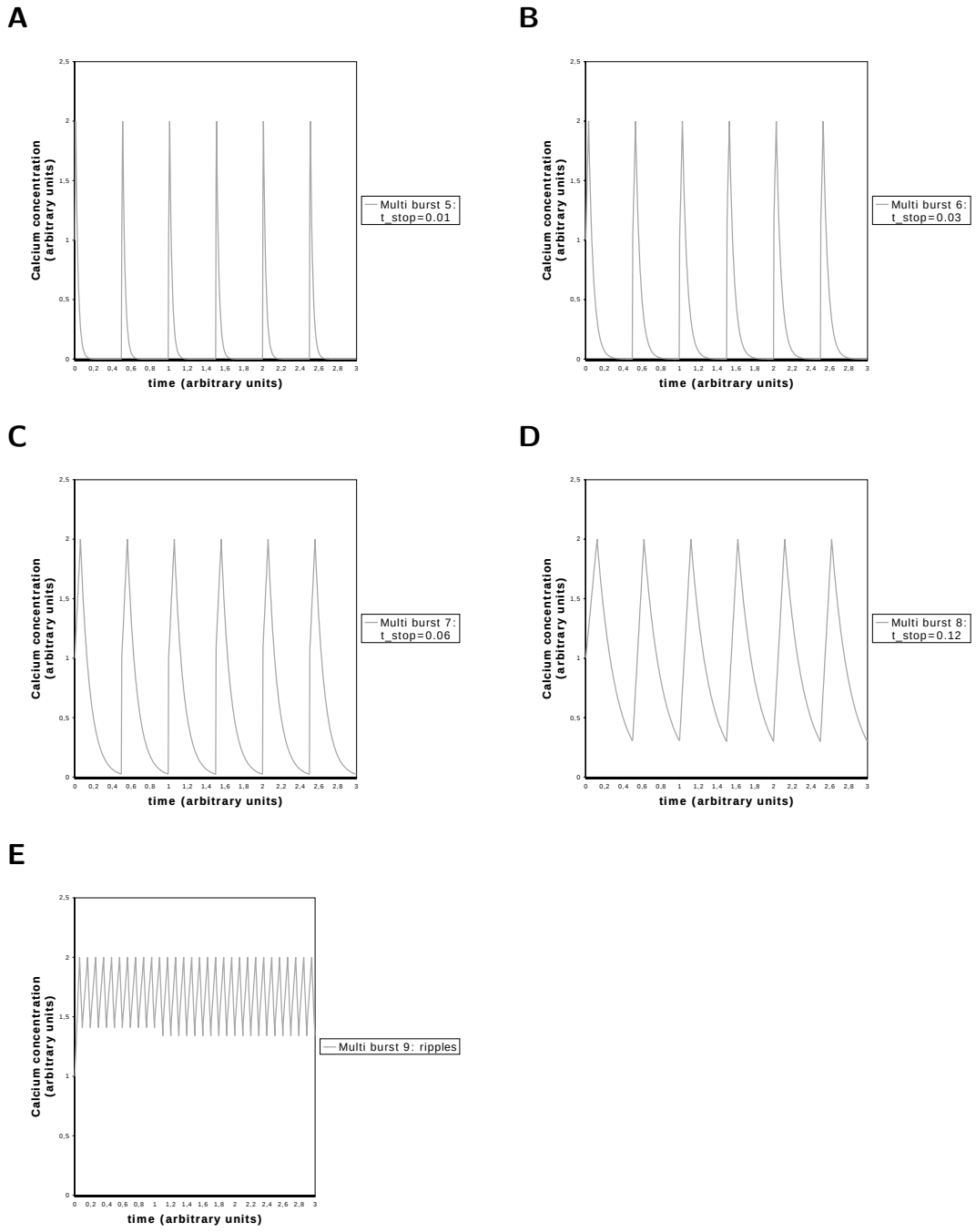


Figure 3.10: *(A)-(D) Manipulating the peak time t_stop generates different types of CCTs. (E) Calcium ripples can be induced.*

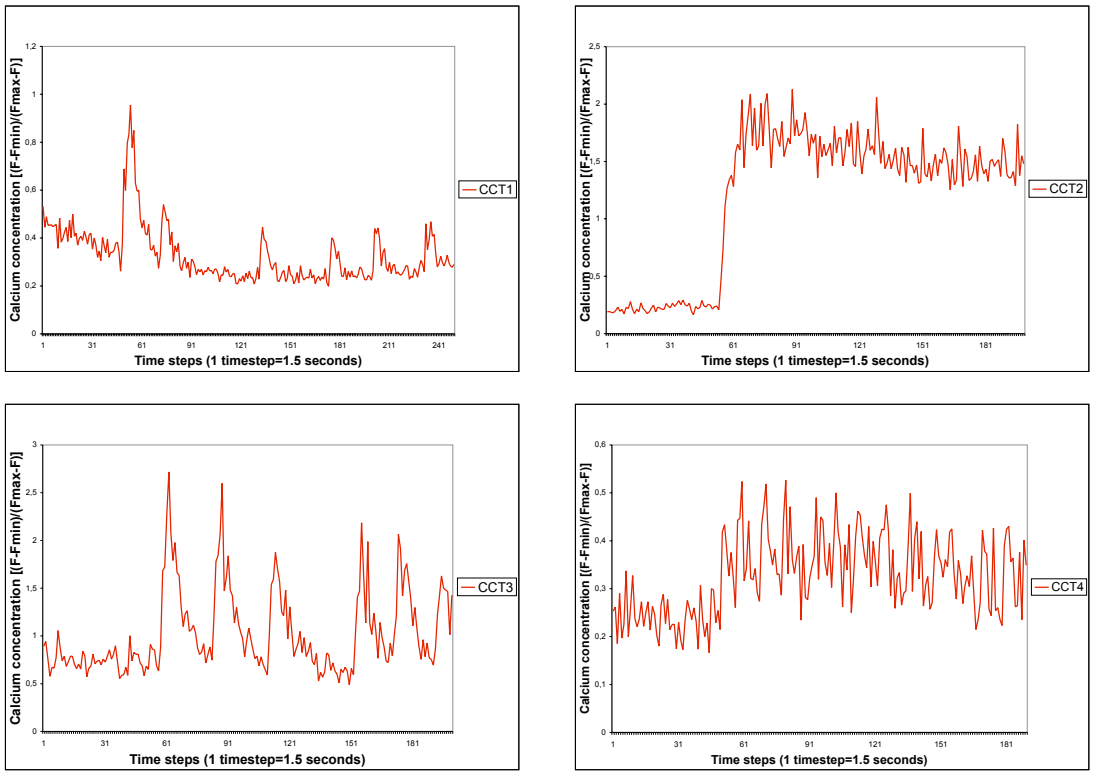


Figure 3.11: *Different types of experimentally measured cytosolic calcium transients.*

transients, and are ideal for systematic evaluations of the effects on signaling outcomes caused by changes in the parameters mentioned above, the third approach for defining boundary condition functions is driven by the idea of constructing a most realistic scenario.

Not only can the model be based on realistic three-dimensional geometries that are accessible from a nucleus data base (Appendix and [55]), it can also incorporate experimentally measured calcium transients. For this purpose, a file reading function was integrated into *UG*, that can access specially formatted CCT-data files. These files consist of a two-columned array, the first column containing the time step values, the second the calcium concentrations.

Experimental data was recorded at the IZN Heidelberg by A. Eder and M. Wittmann as part of this project. This data was used to create a realistic simulation environment, some results of these simulations can be found in [68]. In Figure 3.11 some examples of experimentally recorded cytosolic calcium transient data are depicted. At this point, the mathematical model has been sufficiently described, meaning PDE with initial and boundary conditions are formulated.

## 3.5 Several Evaluation Tools for Nuclear Information Processing

As the simulation environment for calcium signaling in nuclei is fully described in a mathematical sense and all accessible biological data was exploited for constructing a realistic model of the present biological context, the next objective is to retrieve vital data from simulation runs, and also retrieve data that is not accessible from the experimental side. Where in experiments only *calcium load* (summed calcium concentration in a pre-defined region of interest) in slices from hippocampal neuron nuclei can be evaluated, the model can go further into the analysis to make prognoses for calcium signaling effects, which can lead the way for further biological research. It is necessary to assess current knowledge and hypotheses from biological research, which can then be used as the basis for the developed simulation environment.
In this section, tools for evaluating simulation data are introduced. Each tool was developed on the basis of current biological hypotheses as well as knowledge of calcium signaling in hippocampal neuron nuclei, and implemented in the simulation software *UG*.

### 3.5.1 General Implementation

The evaluation tools were included at two *UG*-locations:

1. In `commands.c`: located at `UG/ug/ui/`
2. In `disctools.c`: located at `UG/ug/np/udm/`

In `commands.c` the routines described in this section, which are implemented in `disctools.c`, are made known at *UG*-scripting level with the names *vmean*, *activity*, *regionalact*, *saveVolPreprocess*, *saveVolume* and *centernuccal*. Further sub-routines are addressed in the following sections. Figure 3.12 depicts the schematics of the execution of the evaluation routines at scripting level. The command names in Figure

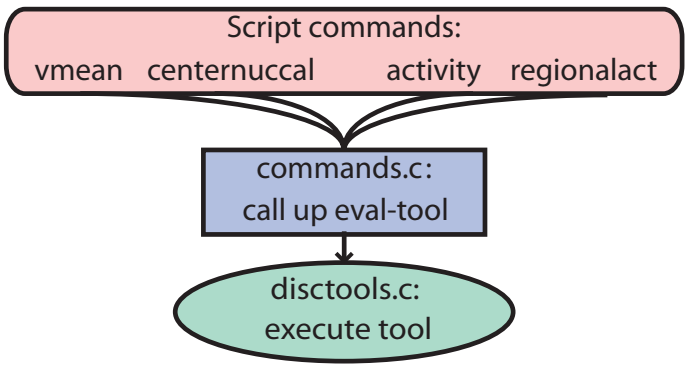


Figure 3.12: *Implementation structure of the evaluation tools in UG.*

```
# functions for solving process:
postprocess_step = "{
    centernuccal $x sol $t @TIME;
    vmean $x sol $t @TIME; # $l @l;
    activity $x sol $t @TIME; # $l @l;
    regionalact $x sol $t @TIME; #l @l;

    if (DOSCREENGRAPH==1) @updategraph; # update plot on screen
    if (DOMETAGRAPH==1)   @mkframe;     # make frame of film;   calls '@updategraph'
    if (DOPSGRAPH==1)     @mkpsframe;   # make postscript plot; calls '@updategraph'
}";
```

Figure 3.13: *Using the evaluation tools: Example of UG-script implementation of centernuccal, vmean, activity and regionalact.*

3.12 can be included on script-level according to Figure 3.13, their call-up routine is implemented in `commands.c` and finally the algorithmic routine implemented in `disctools.c`. Having outlined the general implementation of the evaluation tools, the following sections address the biological background of each tool as well as the translation to *UG*.

### 3.5.2 Calcium Load

The calcium load measuring tool computes the calcium concentration within the nucleus. The concentration level of calcium plays a vital role in the signaling results in respect to fluctuations in the cytosolic calcium concentration. To cite a few, [32, 41, 19] address the influence of calcium waves on fate and reaction of hippocampal nuclei. It is to be noticed, that pathologically high or pathologically low levels of NMDA receptor activity are cell death-promoting, and thus define an activity interval for NMDA receptor activity in which cell survival is promoted, [32].
Furthermore, the frequency of a calcium signal seems to play an additional role for information processing in the nucleus. Calmodulin, a receptor for calcium ions, see Section 1.3, is sensitive to calcium wave frequency, [41]. Depending on the frequency of calcium waves, Calmodulin varies in its activity. Higher frequencies in calcium oscillations lower the activity threshold for calmodulin, [41]. Even different outcomes in the expression of genes could be observed, depending on the form of the calcium signal, [19].
This shows the importance to measure calcium concentration in the nucleus as a measure for nuclear fate and the influence of nuclear morphology on the concentration levels and distribution within the nucleus. With the script name `vmean`, a routine is implemented to measure calcium concentration in each element of the finite volume discretization of the nucleus at each time step of the simulation. The concentration levels can be either read out at specific points in the nucleus, or as the sum over all nodes in the grid normed with the volume of the nucleus. The latter method represents the evaluation method of calcium load in biological experiments.

**User information:** The output of `vmean` is tagged with the string "VMEAN" and

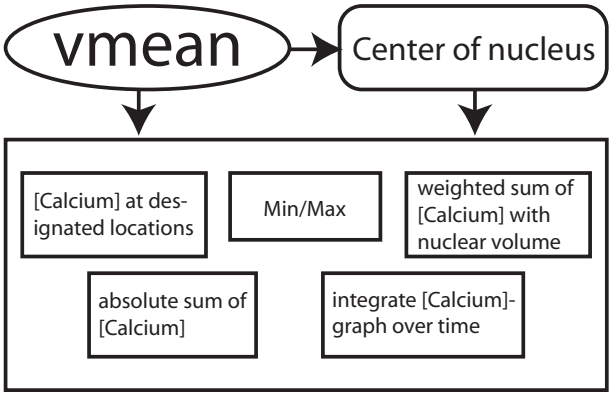


Figure 3.14: *Evaluation of mean nuclear calcium concentration: Different options are offered by* vmean *for evaluating calcium dynamics.*

can so be piped into a `.dat`-file for data evaluation and visualization:
`grep VMEAN log/nucleus3d.log > dummy.dat`

Furthermore, `vmean` has the option of integrating the graph of calcium load in time. This value is used in this project as a measure for the state of a nucleus. This state represents the influence of the nuclear morphology on the calcium dynamics in the nucleus over a given time span. Figure 3.14 shows an overview of the possible applications of `vmean`.

### 3.5.3 Calcium Activity

The concentration of calcium ions in the nucleus is dependent on the ability of passing on cytosolic calcium signals into the nucleus. This happens by passage through nuclear pores that line the nuclear envelope. That means the more nuclear pores are located on the envelope or the larger the envelope itself is, the faster calcium can enter and leave the nucleus. Measuring the calcium load as described above, is a representation of the size of the nuclear envelope, and the number of pores on it, in relation to the overall nuclear calcium concentration.
It becomes clear, that the actual morphology of the nucleus only influences envelope size and the number of nuclear pores. Yet an increase in envelope size and pore number would also be generated by a larger spherical structure with increased diameter, compared to a smaller spherical nucleus. The question, why the nuclear envelope forms complex infolded structures can only be answered by a further biological observation, which can be included into the mathematical model by means of the measuring tool *calcium activity*.
In Section 1.3 different pathways for calcium entry into the nucleus are described, as well as the different functions calcium carries out in the nucleus. The fine regulation of biological processes – if a process is activated and which of the different processes are addressed by a calcium signal – is controlled by activation thresholds

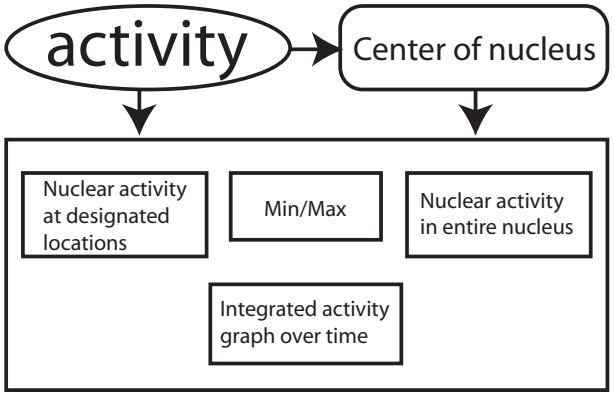


Figure 3.15: activity *allows the user to evaluate threshold dependent nuclear processes, either globally or at designated locations.*

for calcium, [18, 19]. In [18] it is proposed, that the two governing, calcium induced processes of CREB phosphorylation through calmodulin activation and the MAPK pathway induction are regulated by a calcium threshold.
The data indicates, that if the calcium concentration exceeds a biological threshold, the first of the mentioned processes is activated, whereas the second one predominates when the calcium concentration drops below the threshold. Therefore, not only the global calcium concentration, regulated by the number of pores, is vital for the fate of the cell, but also the local signals, meaning the areas within the nucleus that are above a given threshold at a certain time. This is the point, where the actual geometry of the nucleus comes into play.
Where the size of a spherical nucleus can influence the global calcium concentration, spherical nuclei with varying diameters cannot directly regulate local areas above threshold. On the contrary, nuclei with an infolded nuclear envelope reduce diffusion distances without needing to reduce the size of the nuclear envelope and therefore are prone to increase the area inside the nucleus that is above threshold and will be more dynamic in interpreting biologically relevant signals, [19]. Calcium waves from the cytosol will enter the nucleus from adjacent sides and will therefore overlap faster.
In order to evaluate the effect of the nuclear morphology on the percentage of region above and below a given threshold, the tool with the script name `activity` was implemented in *UG*. The tool `activity` is derived from `vmean` with additional threshold and "active volume" to volume ratio evaluation. `activity` loops over all grid elements of the current *UG* grid and calculates the entire volume of the grid as well as the summed volume of those elements, whose calcium concentration values lie over a given threshold. The ratio between the active volume elements and the nuclear volume is called nuclear activity at a given time step. Activity graphs in time can thus be retrieved from simulation runs. A further option is the integration of the current activity graph which offers a scalar value for each nucleus with which nuclei with different geometries can be compared directly.

**User information:** The output of `activity` contains the string "ACTIVITY" and can be directed into a file with the shell command:
`grep ACTIVITY log/nucleus3d.log > dummy.dat`.

Figure 3.15 shows the different components of `activity` which can be used for nuclear activity evaluation in a simulation run.

### 3.5.4 Concentration Minimum and Nuclear Center

A further option for measuring minimum and maximum calcium concentration values in the nucleus are included as a subprocess of `vmean`. Especially the location and value of the concentration minimum plays an interesting role in threshold- and regionally-dependent processes, [18, 19]. Measuring the concentration minimum allows systematic geometry investigations, especially on test geometries (see Section 3.6). This additional parameter is directly linked to the geometry of a nucleus. Shortened diffusion distances by infolding of the nuclear envelope shifts the location of the minimum and the value of the minimum.

In order to retrieve information about the inertness of a nucleus, this being the ability to react to a given cytosolic calcium input, the evaluation of the calcium dynamics within the center of a nucleus showed to be prosperous. The center area of a nucleus is furthest away from the nuclear envelope, therefore the most inert area of the nucleus. To find the center of the nucleus, the application loops over inner nodes of the grid with a distance check to boundary nodes and identifies the inner nodes furthest away from the boundary elements and therefore farthest away from the nuclear envelope.

### 3.5.5 Splitting the Nucleus Into Micro-Domains and Measuring Compartments

Chapter 5 will show that infolding of the nuclear envelope has the tendency to separate the nucleus into micro-domains within the nucleus. To investigate the ability of the changes in nuclear morphology to produce subdivisions within a single nucleus that can act as independent units, a method for splitting the nucleus into subdivisions was developed. It is then furthermore possible to evaluate separated subunits of the nucleus independently in respect to their calcium dynamics.
The process of separating and evaluating regions within a nucleus is done in four steps:

1. **Visualizing** the nucleus in a three dimensional way.
2. **Identifying** the separating plane for subunits.
3. **Separating** the grid into multiple pools of nodes and elements.
4. **Evaluating** calcium dynamics in defined groups of nodes and elements.

**Visualizing the nucleus with *VisIt* and identifying the separation plane:**

In order to identify the separation planes, a procedure was introduced in *UG* to write visualization output for the open source visualization tool *VisIt*, [63]. The data is written in the `vtk`-format and can thus be imported as a vtk-stack into *VisIt*. This open source tool offers the opportunity to define planes in a three dimensional space. This plane can be positioned in such a way, that it is aligned with a membrane infolding of the nuclear envelope. In addition, the plane's normal vector is delivered by *VisIt* and can be used in *UG*. Figure 3.16 shows a nucleus visualized with *VisIt* with its separating plane.

**Separating the nucleus into subunits and evaluating nuclear sub-compartments:**

With the coordinates of a point on the separation plane and the normal vector of the plane, the grid nodes and elements can be separated into a group "left" and "right" of the plane. Figure 3.17 illustrates the following mathematical check for a node being "left" or "right" of the plane. With "right" the half space is defined, in which the normal vector of the plane points into, with "left" the opposite. We define a random node $\vec{v}$, a point on the plane $\vec{g}$ and the normal $\vec{n}$. From vector analysis, one can apply the equation

$$\cos\vartheta = \frac{\vec{g}\cdot\vec{n}}{|\vec{g}|\cdot|\vec{n}|} \tag{3.30}$$

for defining the angle between the two vectors $\vec{g}$ and $\vec{n}$. To check whether a node is on the right or left side, the angle between the reference vector $\vec{g}-\vec{v}$ and the normal vector is calculated:

$$\cos\vartheta = \frac{(\vec{g}-\vec{v})\cdot\vec{n}}{|\vec{g}-\vec{v}|\cdot|\vec{n}|}. \tag{3.31}$$

If the angle is larger than $\frac{\pi}{2}$, the node lies on the left side of the plane, otherwise on the right side. With $\cos\frac{\pi}{2}=0$ it follows

$$\begin{aligned} 0 &= \frac{(\vec{g}-\vec{v})\cdot\vec{n}}{|\vec{g}-\vec{v}|\cdot|\vec{n}|} \\ \Rightarrow 0 &= (\vec{g}-\vec{v})\cdot\vec{n}. \end{aligned} \tag{3.32}$$

The second equation stands, since it is guaranteed that $(0,0,0)$ lies outside of the nucleus. It can be concluded

$$\begin{aligned} \vec{g}\vec{n}-\vec{v}\vec{n} \geq 0, &\quad \vec{v} \text{ lies on right side.} \\ \vec{g}\vec{n}-\vec{v}\vec{n} < 0, &\quad \vec{v} \text{ lies on left side.} \end{aligned} \tag{3.33}$$

The "left/right"-check for separation of grid nodes and elements equips each node and element with an appropriate tag. In the end, all evaluation tools described

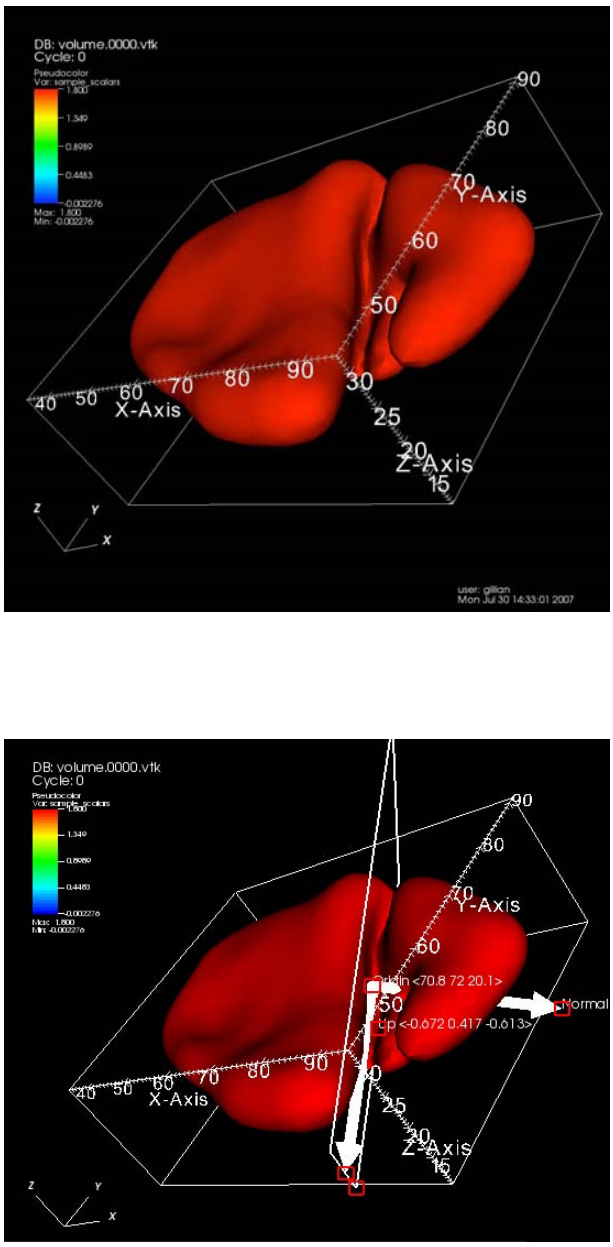


Figure 3.16: *Defining the separation plane of different micro-domains using the visualization tool VisIt.*

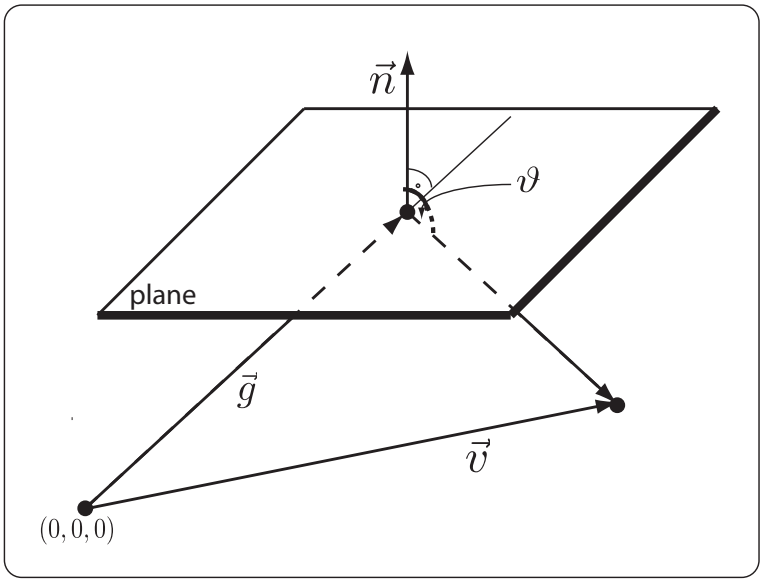


Figure 3.17: *Vector structure for evaluating whether node is "left" or "right" of the plane.*

in this section are also applicable to single sub-compartments of the nucleus. Calcium load and activity in single sub-compartments can be measured, as well as minima/maxima and concentrations in the centers. Figure 3.18 demonstrated the structure and applications for nuclear sub-compartment evaluation.

### 3.5.6 Adaptability to Experimental Measuring Techniques

Aside from measuring calcium load and nuclear activity in subdomains of the nucleus, a further aspect needs to be addressed. In all described cases, calcium load or activity is measured in every node of the nucleus or subcompartment. This also includes boundary nodes. Yet when considering the way biological experimental data for subcompartments is evaluated, special care is taken in the region of the nuclear membrane.
In experiments both membrane and calcium are tagged with a fluorescent. To avoid mistaking membrane fluorescence as a calcium signal, regions of interest are chosen in such a way, that they are slightly drawn away from the nuclear membrane, see Figure 3.19. This will cause an evaluation effect not yet modeled in the simulation environment.
In order to synchronize experimental and model evaluation of calcium dynamics, a final add-on can be activated, which offers different modes of adapting simulation evaluation to fit experimental data evaluation. By neglecting the membrane-nearest parts of the nucleus in the evaluation, inertia effects caused by different nuclear morphologies become visible the way they do in experimental evaluation. Different modes can be chosen for that type of evaluation:

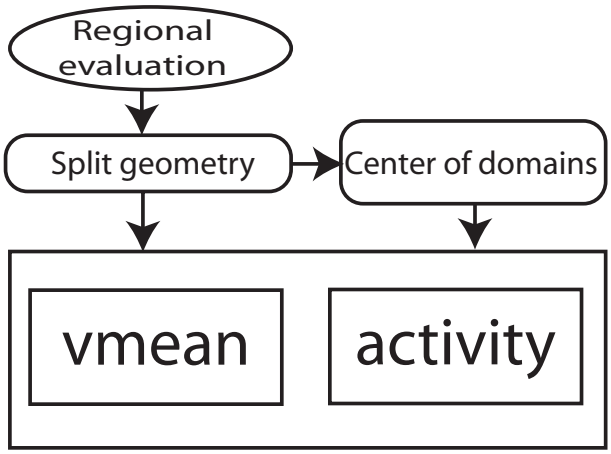


Figure 3.18: *Implementation structure of the evaluation tool* Regional evaluation.

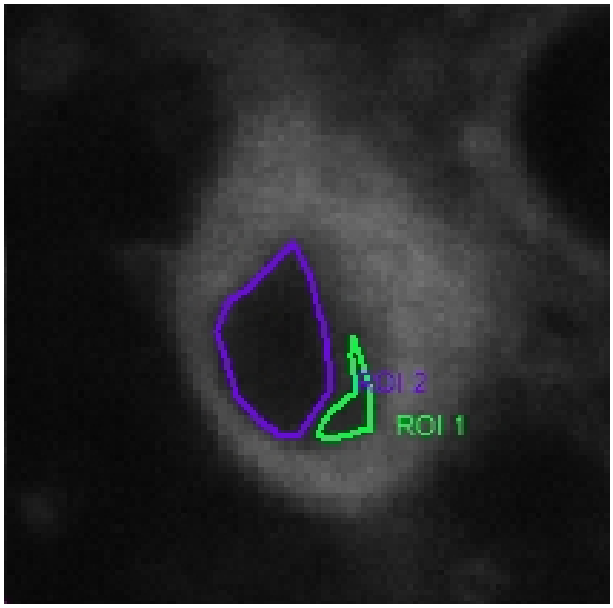


Figure 3.19: *Example of region of interest definition in microscopy data.*

1. Measure only inner nodes (not boundary nodes) in grid.
2. Measure only second level of inner nodes and below. This means that boundary nodes as well as the neighboring inner nodes are exempt from evaluation.
3. Measure nodes within a defined spherical or cubical area within subcompartments.

These different evaluation modes are outlined graphically in Figure 3.20. The above adaptability modes are implemented in the function *centernuccal* which can be executed within the *UG*-script. Figure 3.18 can thus be expanded with extra modes, shown in Figure 3.21.

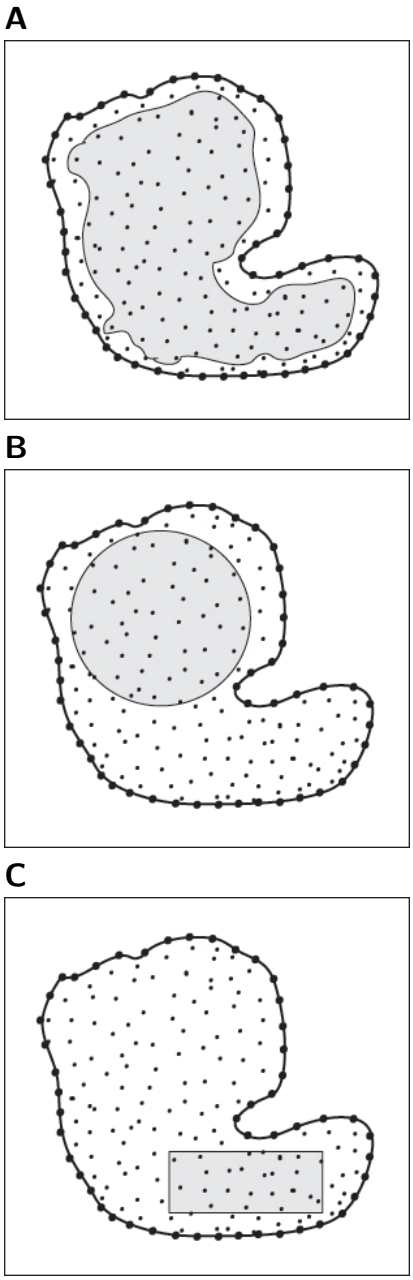


Figure 3.20: *Modes of evaluation: (A) All inner nodes except those that are neighbors to boundary nodes. (B) All nodes inside a defined sphere. (C) All nodes within a defined cuboid.*

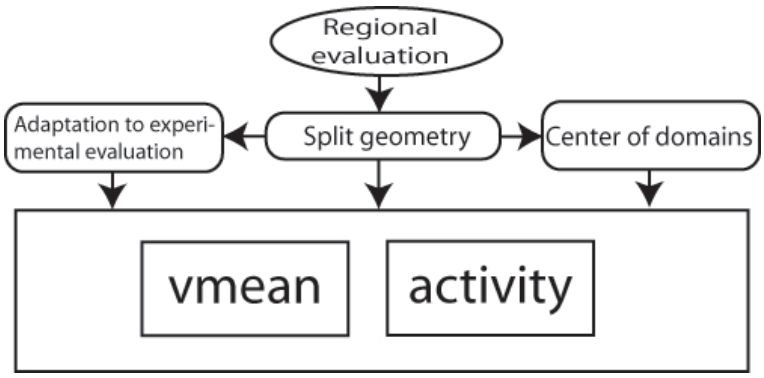


Figure 3.21: *Regional evaluation with the added adaptation option to experimental evaluation.*

## 3.6 Diffusion Models on Test Geometries

Controllable test geometries allow systematic testing of the implemented mathematical diffusion model as well as evaluation procedures. Furthermore, the test geometries were designed in such a way to produce biologically relevant modeling prognoses, concerning the geometric structure of cell nuclei. Since the reconstructed three-dimensional data of hippocampal neuron nuclei show a specific form of membrane infolding, this type of infolding was included in test models, where different parameters are variable, such as size and position of infoldings.

### 3.6.1 2D-Test Models

In a first approach, two dimensional geometries were developed and integrated into a diffusion test model. The need for starting with simulations in two dimensions was mainly for computational reasons, allowing fast simulation runs with broad flexibility in investigating the effect of different set-ups, thus guiding the way for the following, more "expensive" three dimensional simulations.
For the 2D-model the following components were included:

1. Discrete diffusion model from Section 3.2.
2. Spherical and infolded geometries.
3. "Continuous" and "pore" boundary conditions.
4. Two-dimensional infolded test geometry in the standard domain model (Section 3.3.3).

Boundary conditions for the 2D-model can be chosen from the list presented in Section 3.4. In addition, two different modes of boundary conditions were implemented. The first mode can be called a continuous mode, where all boundary nodes are given the same input, be it a constant cytosolic calcium feed or an oscillating signal. The second mode was designed in such a way, that cytosolic calcium intrusion occurs through nuclear pores. This means, that in designated pore regions on the boundary the boundary condition is a Dirichlet condition, outside the pore region the boundary is sealed with Neumann boundary conditions.

**Pore- and continuous mode boundary condition:**

Connected to the boundary of the 2D-model is a sinusoidal function

$$P(x) := \sin(c \cdot x) \tag{3.34}$$

where $c \in \mathbb{N}$ is a parameter which regulates the number of pores on the membrane, $x$ defines the location on the boundary. With a threshold $t$ the pore-region size can be defined, see Figure 3.22:

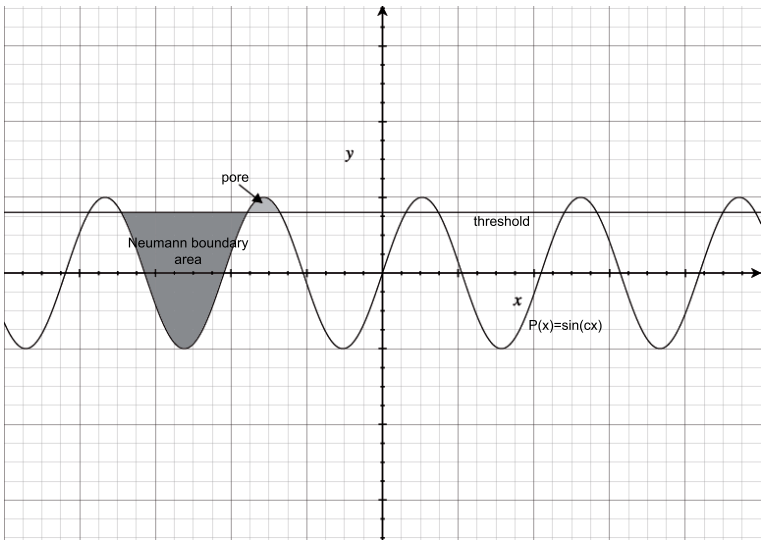


Figure 3.22: *Defining nuclear pore complexes in the standard domain model: A modifiable sinusoidal function is linked to the boundary conditions. Boundary area above a certain threshold is pore area, everything below is nuclear membrane.*

$$\begin{array}{rcl} P(x) \geq t & \Rightarrow & \text{Boundary condition is Dirichlet} \\ P(x) < t & \Rightarrow & \text{Boundary condition is Neumann} \end{array} \tag{3.35}$$

The continuous mode sets the entire boundary to Dirichlet boundary conditions with a constant signal or cytosolic calcium transients from Section 3.4. Graphical depictions of the effect of the two different modes are presented in Figure 3.23.

**2D-test geometries:**

Two-dimensional test geometries were designed to model nuclear forms, exemplarily depicted in Figure 3.24. One is a sphere with two parts, upper and lower halve of a sphere, the second consists of 8 separate parts which are implemented in the most adaptable form. Parameters, that can be chosen by the user are:

1. Center $m_1$ $(= (0, 0))$ of geometry.
2. Radius $r_1$ $(= 1)$ of sphere.
3. Length parameter $t$ $(= 1.4)$ for part 1 and 7 of geometry.
4. Radius $r_2$ $(= 0.1)$ of infolding dip.
5. End point $n$ $(= -0.564 + m_1^{(2)})$ of infolding.

The 8 parts of the geometry shown in Figure 3.25 are defined in the following way:

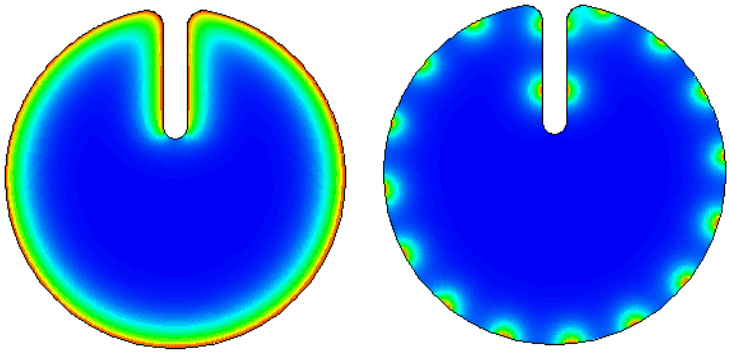

Figure 3.23: *Two types of boundary conditions: The left image depicts a continuous boundary condition, where the entire envelope is surpassable, the right image shows a nuclear pore complex boundary condition.*

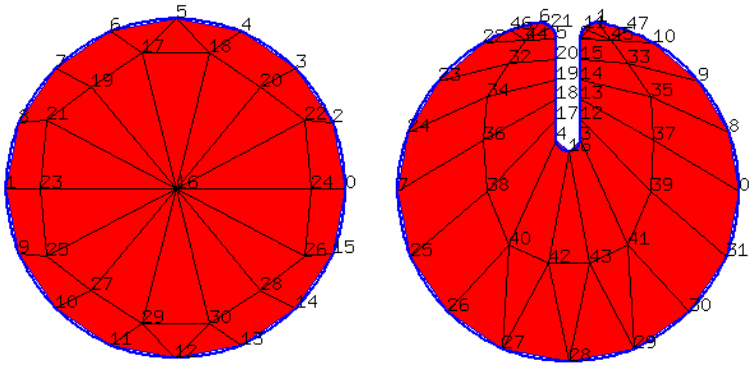

Figure 3.24: *Left: Spherical standard domain geometry. Right: Infolded standard domain geometry.*

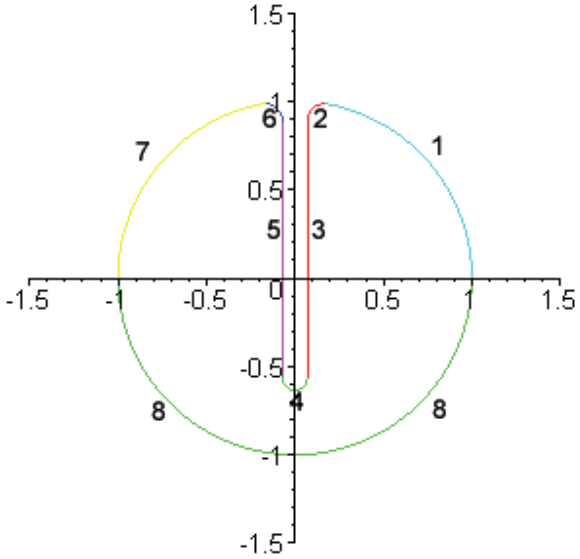


Figure 3.25: *Parts of the standard domain geometry: 8 components define an infolded geometry in the standard domain implementation.*

**Parts of test geometry:**

1. $x = m_1^{(1)} + \cos(l) \cdot r_1, \quad (l = [0, t])$
   $y = m_1^{(2)} + \sin(l) \cdot r_1$

2. $x = m_2^{(1)} + \cos(k) \cdot r_2, \quad (k = [\frac{\pi}{2}, \pi]), \quad m_2^{(1)} := m_1^{(1)} + \cos(t) \cdot r_1 - \cos(\frac{\pi}{2}) \cdot r_2$
   $y = m_2^{(2)} + \sin(k) \cdot r_2, \qquad\qquad\quad m_2^{(2)} := m_1^{(2)} + \sin(t) \cdot r_1 - \sin(\frac{\pi}{2}) \cdot r_2$

3. $x = (m_1^{(1)} + \cos(t) \cdot r_1 + \cos(\pi) \cdot r_2) =: r_k$
   $y = [m_1^{(2)} + \sin(t) \cdot r_1 - \sin(\frac{\pi}{2}) \cdot r_2, n]$

4. $x = m_1^{(1)} + \cos(m) \cdot r_i, \quad (m = [\pi, 2\pi])$
   $y = m_1^{(2)} + n + \sin(m) \cdot r_i, \quad r_i := \cos(t) \cdot r_1 + \cos(\pi) \cdot r_2$

5. $x = r_k - 2 \cdot r_i$
   $y = [m_1^{(2)} + \sin(t) \cdot r_1 - \sin(\frac{\pi}{2}) \cdot r_2, n]$

6. $x = r_k - 2 \cdot r_i - r_2 + \cos(p) \cdot r_2, \quad (p = [0, \frac{\pi}{2}])$
   $y = m_2^{(2)} + \sin(p) \cdot r_2$

7. $x = m_1^{(1)} + \cos(q) \cdot r_1, \quad (q = [\arccos(\frac{r_k - 2r_i - r_2 - m_1^{(1)}}{r_1}), \pi])$
   $y = m_1^{(2)} + \sin(q) \cdot r_1$

8. $x = m_1^{(1)} + \cos(u) \cdot r_1, \quad (u = [\pi, 2\pi])$
   $y = m_1^{(2)} + \sin(u) \cdot r_1$

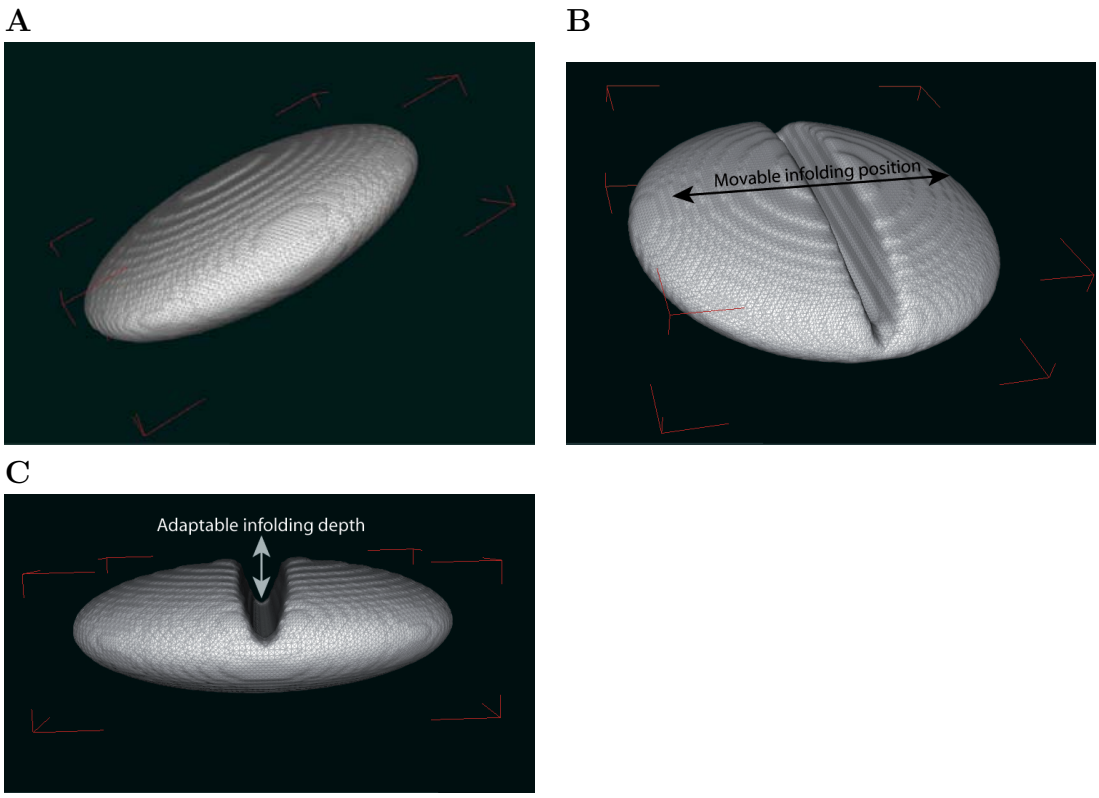


Figure 3.26: *Three dimensional test geometries: (A) Ellipsoid without infolding. (B) Ellipsoid with infolding adjustable in horizontal direction. (C) Ellipsoid with infolding adjustable in vertical direction.*

The above test models with standard domain geometries are implemented in *UG* under `UG/sc/appl2d/scalar2d.c`.

### 3.6.2 3D-Test Models

In analogy to the two dimensional test geometries, three dimensional geometries were generated, one to represent the ellipsoidal morphology of nuclei without infoldings, and one to represent an infolded nucleus in three dimensions. For this purpose, test geometries were developed with the software tool *NeuRA*, [49]. In NeuRA, three dimensional `tiff`-stacks can be generated as a basis for the construction of a linear grid model (lgm). Figure 3.26 shows the ellipsoid and infolded ellipsoid test geometries on which signal processing can be simulated.

# Chapter 4

# Parameter Estimation for Nuclear Calcium Diffusion

# 4.1 Introduction

Signal propagation in nuclei of hippocampal neurons is described as a diffusion process, based on diffusion laws from thermodynamics of irreversible processes. The diffusion coefficient $D$ in the mathematical equation 3.4 is dependent on the biological context. In Section 3.1.2 well integrated literature is cited, that deals with the diffusion coefficient of calcium ions in buffered and free states.
Yet experimental data is retrieved under entirely different circumstances from the biological context presented in this project. While the values for diffusion coefficients for calcium in [1] are good guidelines, the objective of this chapter is to develop a data driven *inverse model*, based on the model from Chapter 3 and experimental data in order to mathematically verify the accuracy of the diffusion coefficients stated in current literature and investigate whether they are applicable to diffusion of calcium ions in hippocampal nuclei. An optimization algorithm is developed and implemented in the simulation software *UG*. The package that contains the parameter estimation implementation is named *scPE* and located at `UG/scPE`.

## 4.1.1 Numerical Optimization

The field of numerical optimization covers a broad spectrum of research fields that are concerned with optimization, ranging from risk management in the financial world, form-optimization in the car industry to identifying parameters in biological systems. In all cases, an *objective* is defined, which is object to an optimization procedure. This objective is dependent on one or more parameters. The task of an optimization algorithm is to identify the value of these parameters that minimize or maximize the objective.
Parameters of the objective can be unconstrained or constrained. In the case of constrained parameters, additional demands are made for the objective. The mathematical formulation of any kind of optimization problem can therefore be formulated as

$$\min_{p\in\mathbb{R}^n} F(p) \qquad \text{subject to (s.t.)} \qquad \begin{array}{ll} c_i(p) = 0, & i \in I_e \\ c_i(p) \geq 0, & i \in I_{ie} \end{array} . \tag{4.1}$$

The optimization problem consists of three parts:

- The *objective function* $F$, which is to be minimized or maximized.
- A set of *parameters* $p$ on which the objective function depends. These are the unknowns to be identified by the optimization procedure.
- *Constraints* $c_i$, these can be equality or inequality constraints which need to be fulfilled by the parameter set $p$.

In order to solve an optimization problem one needs to find an *Optimization Algorithm*, which is an iterative procedure that converges towards a solution for the

optimization problem (4.1). Furthermore, a good optimization algorithm fulfills the following criteria:

- **Efficiency:** Ideally, convergence rates for the algorithm should exceed linear convergence.
- **Robustness:** The algorithm ideally does not depend on the choice of a starting point and is applicable to a broad range of problems.
- **Accuracy:** The algorithm should converge towards the solution of the optimization problem without being strongly influenced by initial errors in the input data or errors caused by exceeding computational accuracy.

Under consideration of the above criteria and the biological context, and the study of fundamental methods in numerical optimization described in Section 4.2, an optimization algorithm for the presented objective is developed in Section 4.3 and finally implemented in *UG*.

### 4.1.2 Biological Context

In [1] experiments in xenopus laevis oocytes were made to retrieve the diffusion coefficients of buffered and unbuffered calcium. It can be assumed, that these data are representative for the context of hippocampal neurons. Until now, no experiments were carried out directly in actual hippocampal neurons. Instead of setting up experiments to measure diffusion coefficients in hippocampal neurons, a new approach is taken in this project. A parameter estimation model is developed in order to minimize the "distance" between model and experiment in respect to the diffusion coefficient of calcium.
Experimental data needs to be recorded in such a way, that is can be included into the mathematical model. Work done by A. Eder (IZN, Heidelberg), [20], where the function of the nuclear envelope as a diffusion barrier for calcium is investigated offers the basis for experiments made in this project. Detailed information about the experimental set-up is given in Section 4.3.6.

## 4.2 The Fundamentals of Parameter Estimation in PDE

Two main algorithmic strategies in optimization algorithms are well established, *line search* and *trust region* methods. Their basic concept is to compute a new parameter set of the objective function in each iteration starting with a chosen (or calculated) initial parameter set. A successful algorithm defines an iteration that converges towards a set of parameters that optimizes the objective function. Giving a general introduction into these main strategies, they can then be modified for the application to PDE-based optimization problems.
The trust region methods operate similarly to line search methods in that they calculate a stepping direction and a step length in the algorithm. The objective function in both cases can be approximated by a quadratic function. For this quadratic approximation a trust region is chosen, dependent on this region a minimizer function is calculated which delivers a step direction and step length simultaneously. Each time the trust region changes, so does step direction and length. Line search methods choose a stepping direction according to some algorithm and then calculate ideal stepping lengths afterwards.
The latter method proved to be ideal for the purpose presented here. For an elaborate description of trust region methods, see [50]. In the following, a brief introduction to line search methods is given, which will be used for the estimation of the diffusion coefficient for nuclear calcium.

**Line search method:**

In order to minimize a function $F$

$$\min_{p\in\mathbb{R}^n} F(p) \tag{4.2}$$

it is substituted by an iterative process

$$\min_{\alpha>0} F(D_k + \alpha\Delta D_k) \tag{4.3}$$

with a line search direction $\Delta D_k$ and a given step length $\alpha > 0$. Minimizing (4.3) in respect to $\alpha$ yields an optimal step length in the direction $\Delta D_k$. It depends on the current problem how precise the step length should be calculated. Precision comes at the cost of computational time. This needs to be put in relation to the effect of the accuracy on the convergence of the algorithm. It might be sufficient to calculate a rough estimate of $\alpha$ for the effective convergence of the algorithm.
In [50] conditions are defined for which a step length $\alpha$ will resolve in a convergent algorithm. Given a search direction $\Delta D_k$, ideally the steepest descent direction $-\nabla F_k$, the *Wolfe conditions* can be stated as

**Definition 5** Wolfe Conditions

$$F(D_k + \alpha_k \Delta D_k) \leq F(D_k) + c_1 \alpha_k \nabla F_k^T \Delta D_k \tag{4.4}$$
$$\nabla F(D_k + \alpha_k \Delta D_k)^T \Delta D_k \geq c_2 \nabla F_k^T \Delta D_k \tag{4.5}$$

*with* $0 < c_1 < c_2 < 1$.

The first condition demands, that the reduction of $F$ be proportional to $\alpha_k$ as well as $\nabla F_k^T \Delta D_k$. The second condition, the curvature condition, couples the value of the slope of $F(D_k + \alpha \Delta D_k)$ with the step length $\alpha$. This condition indicates whether a great change in $F$ can be expected or not, and adjust $\alpha$ accordingly. The following Lemma states the existence of a step length satisfying the Wolfe condition.

**Lemma 1** *Suppose that* $F : \mathbb{R}^n \longrightarrow \mathbb{R}$ *is continuously differentiable. Let* $\Delta D_k$ *be a descent direction at* $p_k$*, and assume that* $F$ *is bounded below along the ray* $\{p_k + \alpha \Delta D_k | \alpha > 0\}$*. Then if* $0 < c_1 < c_2 < 1$*, there exist intervals of step lengths satisfying the Wolfe condition.*

For proof of this lemma, consult [50]. A similar set of conditions are the *Goldstein conditions*, which state

$$F(p_k) + (1-c)\alpha_k \nabla F_k^T \Delta D_k \leq F(p_k + \alpha_k \Delta D_k) \leq F(p_k) + c\alpha_k \nabla F_k^T \Delta D_k \tag{4.6}$$

with $0 < c < 1/2$. A *Backtracking Line Search* is proposed in [50], which eliminates the second of the Wolfe condition (4.5) and can be formulated as

**Algorithm 2** (Backtracking Line Search)
*Initial step length* $\tilde{\alpha} > 0, \rho \in (0,1), c \in (0,1)$; *Set* $\alpha \leftarrow \tilde{\alpha}$;
***repeat*** *until* $F(p_k + \alpha \Delta D_k) \leq F(p_k) + c\alpha \nabla F_k^T \Delta D_k$
$\alpha \leftarrow \rho\alpha$;
***end (repeat)***
*Terminate with* $\alpha_k = \alpha$.

This backtracking approach enforces the sufficient decrease condition, choosing $\alpha$ as a fixed value or dynamically to fulfill condition (4.4), and shows to be applicable with Newton methods with steepest descent conditions (of which will be made use of later). An optimization algorithm is ideally globally convergent. For it to be so, certain demands must be made for the search direction $\Delta D_k$ of the line search objective function. It can be shown, that the steepest descent method, i.e. choosing the step direction $-\nabla F_k$, is globally convergent and also all search directions that do not deviate far from the steepest descent direction. This can be put into a geometric argument, when considering the angle $\vartheta_k$ between $\Delta D_k$ and $-\nabla F_k$

$$\cos \vartheta_k = \frac{-\nabla F_k^T \Delta D_k}{\|\nabla F_k\| \|\Delta D_k\|}. \tag{4.7}$$

The following theorem guarantees that, as long as the search direction does not converge towards the normal of the gradient of $F$, global convergence is given.

**Theorem 1** *Consider any iteration of the form $p_{k+1} = p_k + \alpha_k \Delta D_k$, where $\Delta D_k$ is a descent direction and $\alpha_k$ satisfies the Wolfe conditions (4.4) and (4.5). Suppose that $F$ is bounded below in $\mathbb{R}^n$ and that $F$ is continuously differentiable in an open set $\mathcal{N}$ containing the level set $\mathcal{L} := \{p : F(p) \leq F(x_0)\}$, where $p_0$ is the starting point of the iteration. Assume furthermore that the gradient $\nabla F$ is Lipschitz continuous on $\mathcal{N}$, i.e.*

$$\exists L > 0 \quad \|\nabla F(p) - \nabla F(\tilde{p})\| \leq L\|p - \tilde{p}\|, \quad \forall p, \tilde{p} \in \mathcal{N}. \tag{4.8}$$

*Then*

$$\sum_{k \geq 0} \cos^2 \vartheta_k \|\nabla F_k\|^2 < \infty. \tag{4.9}$$

A proof for the above theorem is given in [50]. From the theorem one can immediately follow, that

$$\cos^2 \vartheta_k \|\nabla F_k\|^2 \to 0. \tag{4.10}$$

Now, under the assumption that the search direction does not converge towards the normal of $\nabla F_k$, which means

$$\exists \vartheta > 0 \quad \cos \vartheta_k \geq \vartheta, \quad \forall k, \tag{4.11}$$

it follows, that

$$\lim_{k \to \infty} \|\nabla F_k\| = 0. \tag{4.12}$$

In particular it follows that the method of steepest descent, $\Delta D_k := \nabla F_k$, is globally convergent.

## 4.3 The Least Squares Problem

In this section an optimization model will be defined, that brings together a mathematical model for calcium diffusion and experimental data from calcium propagation in hippocampal neurons. The goal is to identify the diffusion coefficient of calcium in this cell type using an inverse modeling approach with a *Least Squares Problem*.

### 4.3.1 Introduction

The objective function of a least squares problem defines the distance (the deviation) between the mathematical model and experiment. With deviation $d_j$, the objective function can be written as

$$F(D) := \sum_{j=1}^{m} d_j^2(p). \tag{4.13}$$

The least squares problem can be a constrained or unconstrained, linear or nonlinear problem, depending on the origin of the underlying objective. Figure 4.1 illustrates the concept of solving a least squares problem, i.e. finding a function that minimizes the squared distance between simulation data and experimental data.

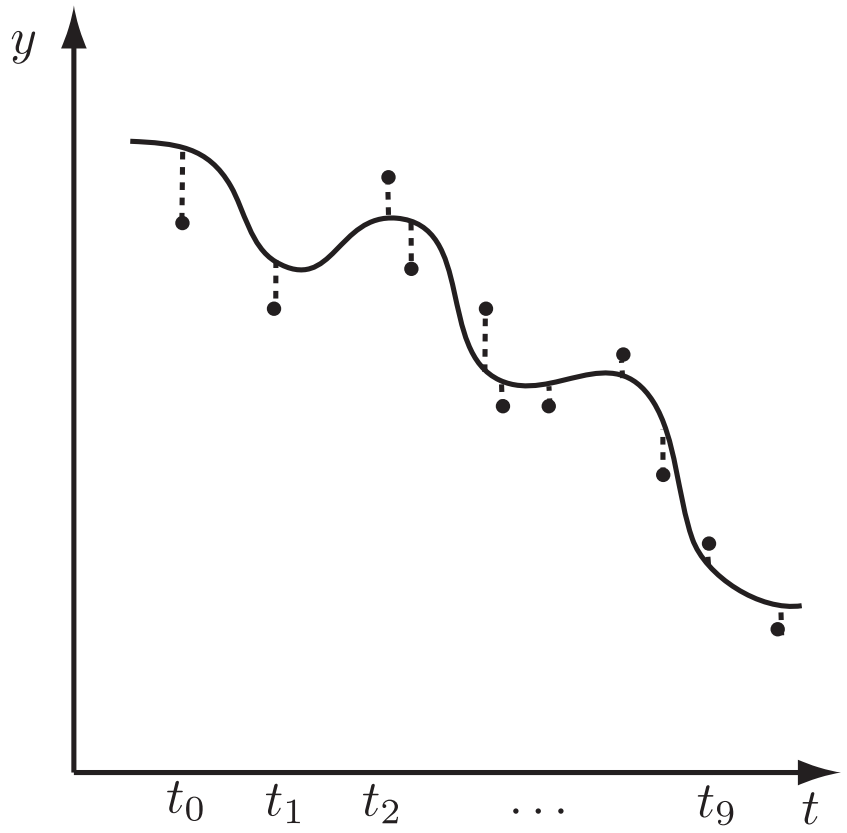


Figure 4.1: *Example of a least squares solution for experimental data points.*

### 4.3.2 Definition of the Optimization Objective

The optimization objective of identifying the diffusion coefficient of calcium ions can be formulated as a least squares problem. The objective function of this problem is defined by

$$F(D) := \sum_{i=1}^{N} \sum_{j=1}^{M} (u_{ij}(D) - \hat{c}_{ij})^2 \tag{4.14}$$

with $\hat{c}_{ij}$ being the experimentally measured values at $N$ different locations in the neuron and at $M$ different times. $u_{ij}$ are the values of the simulation at the same locations and times as the experimental data (Figure 4.2). As the simulation data is derived from the model from Chapter 3, the constraint to the minimization problem

$$\min F(D) \tag{4.15}$$

is, that the minimizing function is a solution of the diffusion model for calcium. This yields a constrained minimization problem

$$\min F(D), \quad \text{s.t. } u(D) \in \mathcal{B}, \tag{4.16}$$

with $\mathcal{B}$ the solutions of the diffusion model.

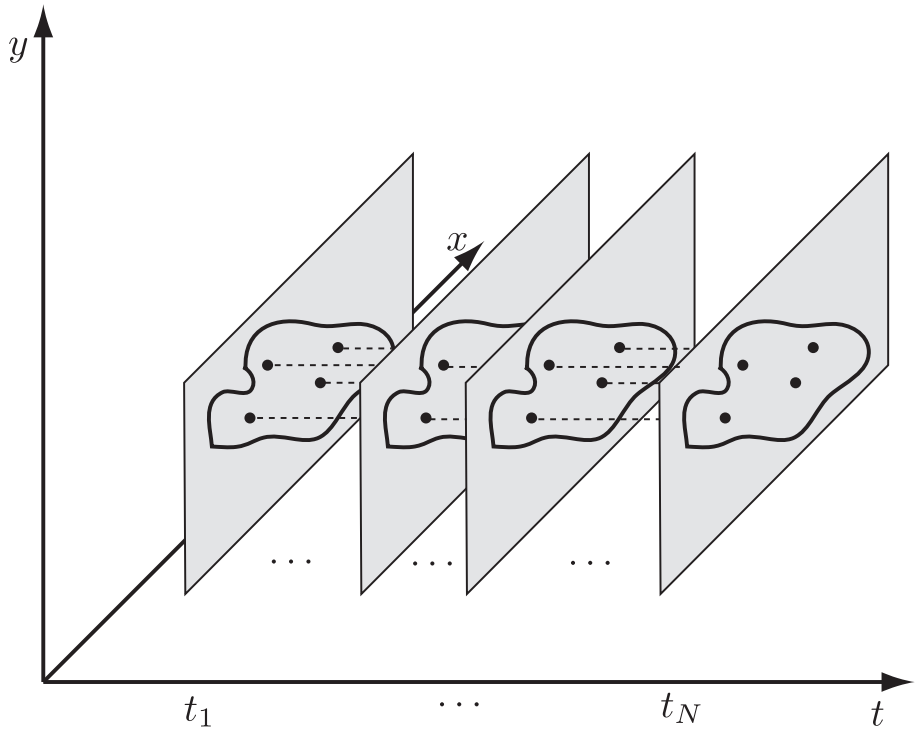


Figure 4.2: *Time course of data recording: At times $t_1 \dots t_N$ the calcium data at different points are recorded.*

### 4.3.3 A Steepest Descent Algorithm for the Diffusion Model

One way to solve the minimization problem

$$\min_D F(D) := \sum_{i=1}^{N} \sum_{j=1}^{M} (u_{ij}(D) - \hat{c}_{ij})^2 \tag{4.17}$$

is applying a steepest descent iteration. This process requires the calculation of a step direction $\Delta D$ in each iteration, which is defined as the derivative of the objective function $F(D)$. In Section 4.2 the backtracking line search algorithm shows to be convergent for the steepest descent direction. In Figure 4.3 the concept of the

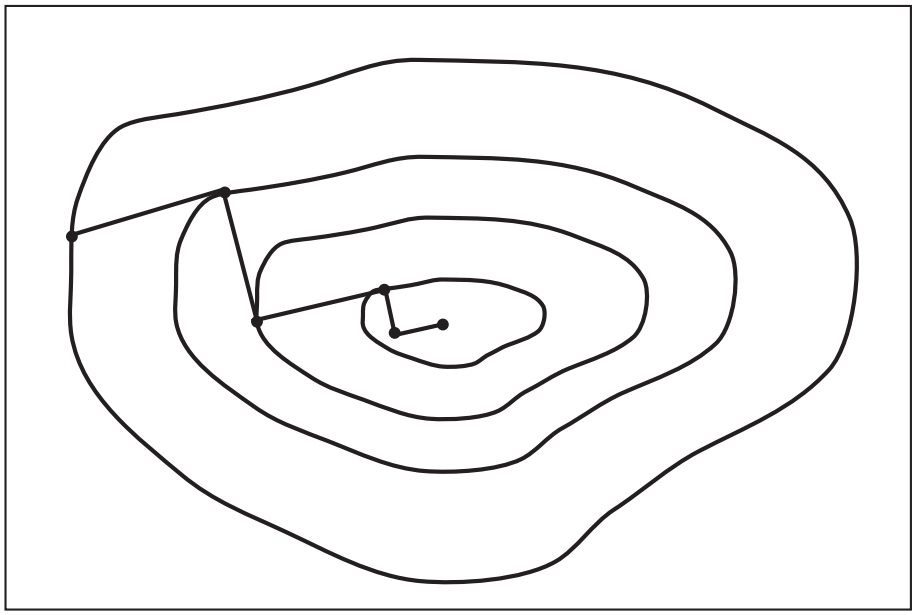

Figure 4.3: *Contour lines of objective functions and steepest descent direction.*

steepest descent method is outlined.
An approximation that can be made is the *local* representation of the nonlinear objective function $F(D)$ by a linearized quadratic function. With the iteration parameter $q$ the linearized minimization problem yields

$$\min \sum_{i,j} \left( u_{ij}(D^{(q)}) - \hat{c}_{ij} + \frac{\partial u_{ij}}{\partial D^{(q)}} \cdot \Delta D^{(q)} \right)^2 \tag{4.18}$$

by applying a Taylor-development to equation (4.14) and including the first order term and

$$\Delta D^{(q)} := D^{(q+1)} - D^{(q)}. \tag{4.19}$$

Deriving the linearized quadratic objective function, and calculating $\Delta D$ for which the derivative is zero, defines the steepest descent direction for each iteration step in

the algorithm which converges towards a minimal deviation between modeled and experimental data. The linearized quadratic objective function is defined as

$$\tilde{F} := \sum_{i,j} \left( u_{ij}(D^{(q)}) - \hat{c}_{ij} + \frac{\partial u_{ij}}{\partial D^{(q)}} \cdot \Delta D^{(q)} \right)^2 . \tag{4.20}$$

The derivative of $\tilde{F}$ in respect to $D$ is

$$\frac{\partial \tilde{F}}{\partial D^{(q)}} = \sum_{i,j} \left( 4 \left( u_{ij}(D^{(q)}) - \hat{c}_{ij} + \frac{\partial u_{ij}}{\partial D^{(q)}} \cdot \Delta D^{(q)} \right) \cdot \frac{\partial u_{ij}}{\partial D^{(q)}} \right) . \tag{4.21}$$

When setting $\frac{\partial \tilde{F}}{\partial D^{(q)}}$ to zero, one can retrieve the steepest local descent direction $\Delta D^{(q)}$:

$$\begin{aligned}
& \tfrac{\partial \tilde{F}}{\partial D^{(q)}} = 0 \\
\Leftrightarrow \quad & \textstyle\sum_{i,j} \left( 4 \left( u_{ij}(D^{(q)}) - \hat{c}_{ij} + \frac{\partial u_{ij}}{\partial D^{(q)}} \cdot \Delta D^{(q)} \right) \cdot \frac{\partial u_{ij}}{\partial D^{(q)}} \right) = 0 \\
\Leftrightarrow \quad & 4 \cdot \textstyle\sum_{i,j} \left( \frac{\partial u_{ij}}{\partial D^{(q)}} \cdot u_{ij}(D^{(q)}) - \hat{c}_{ij} \right) + \sum_{i,j} \left( \left( \frac{\partial u_{ij}}{\partial D^{(q)}} \right)^2 \cdot \Delta D^{(q)} \right) = 0 \\
\Leftrightarrow \quad & \textstyle\sum_{i,j} \left( \hat{c}_{ij} - \frac{\partial u_{ij}}{\partial D^{(q)}} \cdot u_{ij}(D^{(q)}) \right) \left( \left( \frac{\partial u_{ij}}{\partial D^{(q)}} \right)^2 \cdot \Delta D^{(q)} \right) = \\
& \qquad = \textstyle\sum_{i,j} \left( \hat{c}_{ij} - \frac{\partial u_{ij}}{\partial D^{(q)}} \cdot u_{ij}(D^{(q)}) \right) \\
& \qquad \vdots \\
\Leftrightarrow \quad & \Delta D^{(q)} = \frac{\sum_{i,j} \left( \hat{c}_{ij} - \frac{\partial u_{ij}}{\partial D^{(q)}} \cdot u_{ij}(D^{(q)}) \right)}{\sum_{i,j} \left( \frac{\partial u_{ij}}{\partial D^{(q)}} \right)^2}
\end{aligned} \tag{4.22}$$

With the step direction

$$\Delta D^{(q)} = \frac{\sum_{i,j} \left( \hat{c}_{ij} - \frac{\partial u_{ij}}{\partial D^{(q)}} \cdot u_{ij}(D^{(q)}) \right)}{\sum_{i,j} \left( \frac{\partial u_{ij}}{\partial D^{(q)}} \right)^2} \tag{4.23}$$

the new diffusion coefficient $D^{(q+1)}$ can be calculated exploiting the following equation:

$$\begin{aligned}
& \Delta D^{(q)} = D^{(q+1)} - D^{(q)} \\
\Leftrightarrow \quad & D^{(q+1)} = D^{(q)} + \Delta D^{(q)}
\end{aligned} \tag{4.24}$$

With the step direction defined, a parameter estimation algorithm for the diffusion coefficient $D$ of calcium can be formulated.

**Algorithm 3** (Parameter estimation for calcium ions)
*Initialize $D^{(0)}$ as the initial diffusion coefficient according to Section 3.1.2.*
*Import experimental data.*
***repeat** until $F(D^{(n)}) \leq \varepsilon$, $\varepsilon > 0$ and $q = 0 \dots n$*
*Calculate $\Delta D^{(q)}$ according to Eq. (4.23).*
$D^{(q+1)} \leftarrow D^{(q)} + \Delta D^{(q)}$;
***end (repeat)***
*Terminate with $D := D^{(n)}$.*

Several components need to be calculated for the above algorithm. In the following sections, the calculation of these components within the framework of *UG* is addressed.

### 4.3.4 Calculating Simulation Data

In order to calculate equation (4.23), the solutions $u_{ij}$ of the diffusion model are required. This means, that in each iteration of the parameter estimation the simulation of calcium diffusion needs to be carried out until time $M$ is reached. At each measuring time the values at the $N$ measuring points need to be stored. The PDE that is solved for the simulation of calcium diffusion in nuclei is

$$\frac{\partial u(x,t)}{\partial t} = div(D \cdot \nabla u(x,t)). \tag{4.25}$$

In its weak formulation (see Section 3.2.2)

$$\int_\Omega \frac{\partial u}{\partial t} = \int_\Omega div(D \cdot \nabla u), \tag{4.26}$$

the diffusion equation is discretized using flux conserving finite volume discretization and implicit Euler time discretization. With numerical integral approximation this yields the equation (see Section 3.2.2):

$$|b_i| \left(u(t_{k+1}, x_i) - u(t_k, x_i)\right) = \Delta t \sum_{j,l} |\partial b_i \cap \partial b_l| \, u(t_{k+1}, x_j) D \cdot \nabla \xi_j(x_{il}) \cdot \vec{n}_{il}. \tag{4.27}$$

The solution $u(t_{k+1}, x_j)$ is calculated on each finite volume element and stored at each grid node of the volumetric grid. If $t_{k+1}$ and $x_j$ happen to be a parameter estimation time and location, the solution $u_{ij}$ of (4.27) is needed in the algorithm.

### 4.3.5 Calculating Derivatives

The second component that needs to be calculated is the derivative $\frac{\partial u}{\partial D}$. Instead of deriving the analytical diffusion equation (4.25) one can derive the discretization

(4.27) of the PDE and transform it back to an analytical expression.
Deriving (4.27) with respect to $D$ yields:

$$\begin{aligned} &|b_i| \tfrac{\partial}{\partial D^{(q)}} \left(u(t_{k+1}, x_i) - u(t_k, x_i)\right) = \\ &\Delta t \textstyle\sum_{j,l} |\partial b_i \cap \partial b_l| \left(\tfrac{\partial}{\partial D^{(q)}} u(t_{k+1}, x_j) \cdot D^{(q)} + u(t_{k+1}, x_j)\right) \nabla \xi_j(x_{il}) \vec{n}_i \\ \Leftrightarrow \\ &|b_i| \left(\tfrac{\partial}{\partial D^{(q)}} u(t_{k+1}, x_i) - \tfrac{\partial}{\partial D^{(q)}} u(t_k, x_i)\right) = \\ &\Delta t \textstyle\sum_{j,l} |\partial b_i \cap \partial b_l| \left(\tfrac{\partial}{\partial D^{(q)}} u(t_{k+1}, x_j) \cdot D^{(q)} + u(t_{k+1}, x_j)\right) \nabla \xi_j(x_{il}) \vec{n}_i \end{aligned} \tag{4.28}$$

In an iterative manner the needed derivatives can be calculated by solving the above equation

$$\begin{aligned} &|b_i| \left(\tfrac{\partial}{\partial D^{(q)}} u(t_{k+1}, x_i) - \tfrac{\partial}{\partial D^{(q)}} u(t_k, x_i)\right) = \\ &\Delta t \textstyle\sum_{j,l} |\partial b_i \cap \partial b_l| \left(\tfrac{\partial}{\partial D^{(q)}} u(t_{k+1}, x_j) \cdot D^{(q)} + u(t_{k+1}, x_j)\right) \nabla \xi_j(x_{il}) \vec{n}_i. \end{aligned} \tag{4.29}$$

Defining $g(t_k, x_j, D^{(q)}) := \frac{\partial}{\partial D^{(q)}} u(t_{k+1}, x_j)$ yields

$$\begin{aligned} &|b_i| \left(g(t_{k+1}, x_i, D^{(q)}) - g(t_k, x_i, D^{(q)})\right) = \\ &\Delta t \textstyle\sum_{j,l} |\partial b_i \cap \partial b_l| \left(g(t_{k+1}, x_j, D^{(q)}) \cdot D^{(q)} + u(t_{k+1}, x_j)\right) \nabla \xi_j(x_{il}) \vec{n}_i. \end{aligned} \tag{4.30}$$

The right hand side sum can be separated into two parts:

$$\begin{aligned} &\Delta t \textstyle\sum_{j,l} |\partial b_i \cap \partial b_l| \left(g(t_{k+1}, x_j, D^{(q)}) \cdot D^{(q)} + u(t_{k+1}, x_j)\right) \nabla \xi_j(x_{il}) \vec{n}_i = \\ &\Delta t \textstyle\sum_{j,l} |\partial b_i \cap \partial b_l| \, g(t_{k+1}, x_j, D^{(q)}) \cdot D^{(q)} \nabla \xi_j(x_{il}) \vec{n}_i \\ &+ \Delta t \textstyle\sum_{j,l} |\partial b_i \cap \partial b_l| \, u(t_{k+1}, x_j) \nabla \xi_j(x_{il}) \vec{n}_i. \end{aligned} \tag{4.31}$$

The first sum on the right hand side now defines $div(D \nabla g(x,t))$ and the second yields $div(\nabla u(x,t))$. Equation (4.29) is therefore a discretization of

$$\frac{\partial g(x,t)}{\partial t} = div(D \nabla g(x,t)) + div(\nabla u(x,t)) \tag{4.32}$$

Solving the discretization of (4.32) at parameter estimation time and locations concludes the computation of a single iteration of the parameter estimation algorithm. A detailed description of how equation (4.29) is solved is given in Section 4.4.

### 4.3.6 Data from Local Uncaging Experiments

The parameter estimation problem requires experimentally measured data which suits the design of the proposed method. The idea is to simulate calcium diffusion inside nuclei from hippocampal neurons. The mathematical diffusion model demands initial conditions, as well as the time course of the calcium signal injected into the nucleus. Furthermore, the time courses of calcium concentration in several points in the nucleus are needed. To produce these sets of data, experiments with

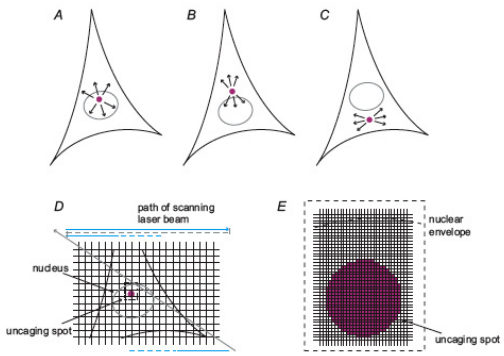


Figure 4.4: *Schematics of uncaging experiments (from A. Eder (IZN)).*

laser-assisted uncaging of calcium in a defined area and measurements at defined distances of the uncaging area were done by A. Eder (IZN, Heidelberg). These experiments are based on [20].
For this purpose, hippocampal neurons were loaded with calcium indicator Fluo-4, [48], and a calcium caging compound, [20]. By applying a UV-laser beam to the caging compound in a restricted area, calcium is released (uncaged) in the specific area where the UV-laser is directed. The experimental set-up was modified in respect to the above model requirements. Local calcium release needs to be recorded as input data for the inverse model. In the experiment, the uncaging parameters were set in such a way, to guarantee that the fluorescence signal actually comes from calcium release itself and not from the laser beam. Thus, calcium release dynamics can be recorded as input data for the developed model.
After uncaging calcium at a defined spot, Figure 4.4, the reaction can be measured at different locations within the nucleus. In order to use maximum recording speed for the concentration time course in specific locations, the calcium distribution was recorded in a three pixel wide band. Since it was verified, that calcium diffuses radially within the nucleus, [20], this experimental structure poses no problems.
Nuclei from hippocampal neurons were loaded with Fluo-4, the calcium indicator, with the caging compound NP-EGTA, AM, and with MitoTracker Deep Red 633 in order to distinguish cyto- and nucleoplasm, [20], and excited with a UV laser line. This method presents the needed structure:

- Calcium is released locally.
- The time course of this release can be recorded.
- Calcium concentration can be measured at known distances along a band in time.

This data can be included in the parameter estimation model presented in this chapter.

## 4.4 Implementing the Least Squares Problem in *UG*

### 4.4.1 General Definition of a Boundary Value Problem in *UG*

*UG* makes use of a class-oriented implementation of numerical procedures for solving mathematical boundary value problems. Such a class, which in C-based *UG* is called a numproc (numerical procedure), is designed in such a way to carry out all methods for computing each component of a PDE. For example, a flow and transport problem which can consist of parts such as viscosity, density, permeability etc. has each of these components described in a numproc and passed on to the discretization.
All numprocs in *UG* are derived from a general class `NP_BASE`, which adds three main methods to each class, `Init()`, `Display()` and `Execute`, see Figure 4.5. These meth-

```
struct np_base {
/* data */
ENVVAR v;                                   /* is an environment variable        */
MULTIGRID *mg;                              /* associated multigrid              */
INT status;                                 /* has a status, NO type and size...  */

/* functions */
INT (*Init) (struct np_base *,INT,char **);     /* initializing routine          */
INT (*Display) (struct np_base *);              /* Display routine               */
INT (*Execute) (struct np_base *,INT,char **);  /* Execute routine               */
};
typedef struct np_base NP_BASE;
```

Figure 4.5: *UG-implementation of struct np_base.*

ods initialize a set of parameters, display the current status and execute a defined action. When all components of the model are described in a numerical procedure, derived from `NP_BASE`, they are calculated and assembled within a discretization and assembly routine, which needs to be implemented separately. In that way, the central multi-grid structure is linked to boundary value problem data and numerical routines. The connectivity between different *UG*-components is graphed in Figure 4.6.

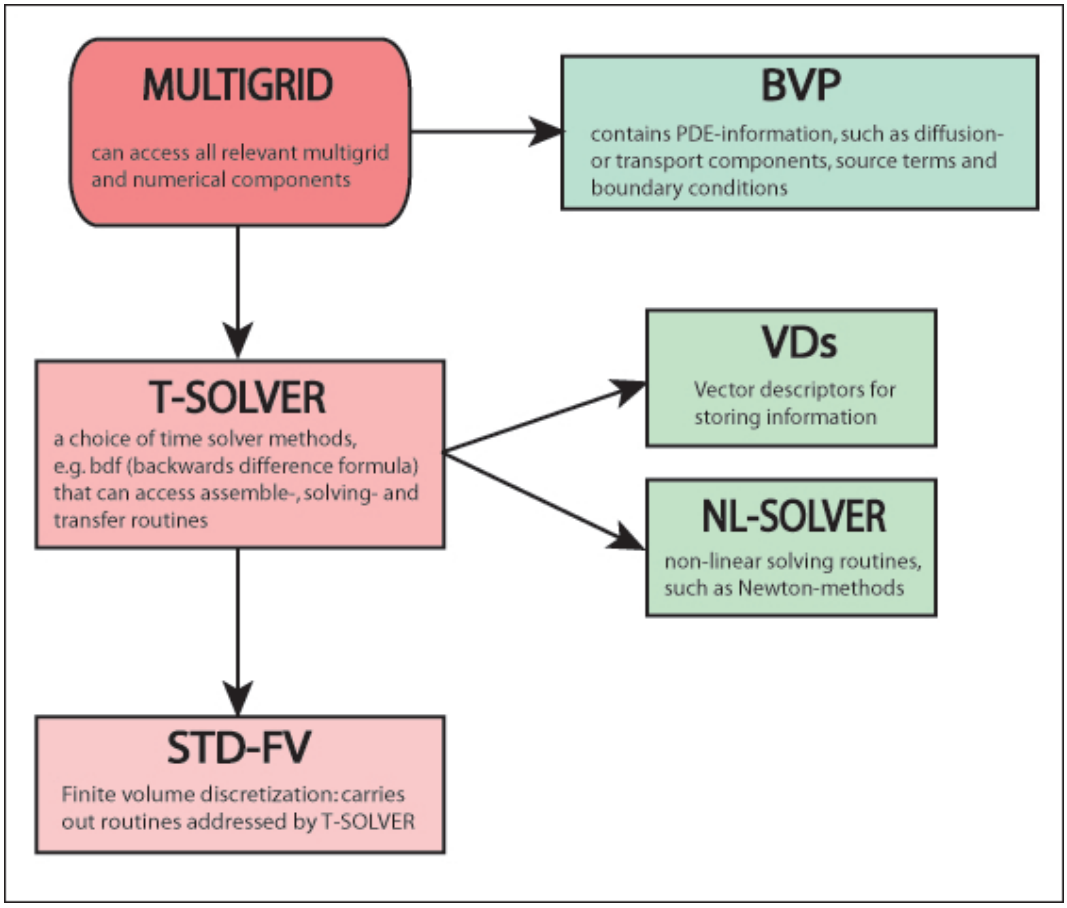


Figure 4.6: *Connectivity graph of components for the standard implementation of a boundary value problem in UG.*

### 4.4.2 Setting up a One-Way Coupled System

In preprocessing steps, the components for the calculation of the steepest descent step direction in the least-squares problem are derived. Recall equation (4.23)

$$\Delta D^{(q)} = \frac{\sum_{i,j} \left( \hat{c}_{ij} - \frac{\partial u_{ij}}{\partial D^{(q)}} \cdot u_{ij}(D^{(q)}) \right)}{\sum_{i,j} \left( \frac{\partial u_{ij}}{\partial D^{(q)}} \right)^2}$$

for which components $u_{ij}(D^{(q)})$ and $\frac{\partial u_{ij}}{\partial D^{(q)}}$ need to be calculated. This is done by solving a system of partial differential equations within a defined geometry and

defined boundary and initial conditions:

$$\frac{\partial u(x,t)}{\partial t} = div(D \cdot \nabla u(x,t)) \qquad \text{on } \Omega \subset \mathbb{R}^{3\times 1} \tag{4.33}$$

$$\frac{\partial g(x,t)}{\partial t} = div(D\nabla g(x,t)) + div(\nabla u(x,t)) \qquad \text{on } \Omega \subset \mathbb{R}^{3\times 1} \tag{4.34}$$

$$u(x,t_0) = g(x,t_0) = u_0(x) \tag{4.35}$$

$$u(x,t) = g(x,t) = f(t) \qquad \text{on } \partial\Omega \subset \mathbb{R}^3 \tag{4.36}$$

In this system, equation (4.34) depends on the first equation (4.33). The solution of (4.33) defines the "known" side for the calculation of $g$. Since (4.33) does not depend on $g$, both equations are coupled only in direction (4.33) $\rightarrow$ (4.34). In their finite volume discretization both equations are identical, except for modified right hand sides. The discrete form for (4.33) can be stated as

$$|b_i|\,(u(t_{k+1},x_i) - u(t_k,x_i)) - \Delta t \sum_{j,l} |\partial b_i \cap \partial b_l|\, u(t_{k+1},x_j) D \cdot \nabla \xi_j(x_{il}) \cdot \vec{n}_{il} = 0, \tag{4.37}$$

with a zero right hand side. In the case of the discretization for the PDE defining $g$, the right hand side is modified by the solution $u$:

$$\begin{aligned} &|b_i|\left(g(t_{k+1},x_i,D^{(q)}) - g(t_k,x_i,D^{(q)})\right) - \\ \Delta t \sum_{j,l} &|\partial b_i \cap \partial b_l|\, g(t_{k+1},x_j,D^{(q)}) \cdot D^{(q)} \nabla \xi_j(x_{il}) \vec{n}_i \\ &= \Delta t \sum_{j,l} |\partial b_i \cap \partial b_l|\, u(t_{k+1},x_j) \nabla \xi_j(x_{il}) \vec{n}_i \end{aligned} \tag{4.38}$$

Gradients of Ansatzfunctions $\xi_j$ and normal vectors $\vec{n}_i$ are retrievable within the finite volume routines and need to be multiplied with solution $u_{ij}$ in each grid node $ij$.
In ordinary schemes of *UG* for solving boundary value problems, the existence and definition of PDE-components are defined in BVP (see Figure 4.6). In a "one-way" coupled system of equations of the sort described above, the right hand side of two PDE vary and need to be redefined in each parameter estimation time step in order to solve both equations. For the purpose of solving one-way coupled equations systems a new method was developed in *UG*. The implementation of (4.33)–(4.35) in *UG* is outlined in the following section.

### 4.4.3 Modifying the BVP Structure for the Least Squares Problem

Recalling Figure 4.6 the PDE-components are defined in BVP, vector descriptors are at disposal to store the solution of the partial differential equation, and boundary conditions are also passed on to BVP. When more than one equation is solved

in each time step, static definitions of PDE-components in BVP need to be made dynamic.
Therefore, the connection to the boundary value problem initiation which the multigrid structure of *UG* retrieves from BVP is cut and a new, dynamic storage possibility of BVP-data is included. To that end, a number of modifications are made in the general structure of implementing a boundary value problem in *UG*.

1. Define a numerical procedure, that contains the necessary components for realizing dynamic PDE-components. This is done in `format.h` of the parameter estimation package located in `UG/scPE`.

2. Modifications to the finite volume discretization component `stdfv.c` located in `UG/scPE/pclib`.

3. Changes in `UG/scPE/nucleus3d/nucleus3d.c` where the parameter estimation problem is generated.

4. Adaption of the script file to carry out a "double time step" in each time step of the parameter estimation algorithm with changing right hand sides of the PDE, located in
`UG/scPE/nucleus3d/scripts`.

5. Implementing the parameter estimation algorithm in `disctools.c`.

The numproc `np_nucleus_param` in `format.h` is shown in Figure 4.7. The flag `computeDerivative` that is defined in this numproc is used to switch between a zero right hand side and the right hand side $\Delta t \sum_{j,l} |\partial b_i \cap \partial b_l| \, u(t_{k+1}, x_j) \nabla \xi_j(x_{il}) \vec{n}_i$ which is used to calculate the derivative $\frac{\partial u}{\partial D}$. Figure 4.8 shows an excerpt of how a switch with `computeDerivative` is integrated in `stdfv.c`.
In `nucleus3d.c` all specifications of the parameter estimation problem are made and the problem is initiated. Figure 4.9 shows the constructor function in `nucleus3d.c` which retrieves all necessary components from the numproc `np_nucleus_param`, defines base functions and parameter estimation problem functions (nucleus problem functions).
Finally, the steepest descent algorithm is implemented in `disctools.c`, located at `UG/ug/np/udm`, and can be executed by the script with the command

```
DiffusionCoefficient $sol sol $deriv deriv $t @TIME;
```

in a post-processing step of the time stepping routine.

**Modifications to the general structure:**

The above changes and implementations offer a new structure for implementing the underlying coupled system of equations. The connection to BVP is substituted by a numproc that dynamically defines PDE-components. The changes in the general structure are shown in Figure 4.10.

```
struct np_nucleus_param {

    NP_BASE base;                       /* inherits base class            */

    /* attributes */
    DOUBLE K_nucleus;                   /* diffusion coefficient          */
    DOUBLE_VECTOR V_nucleus;            /* velocity coefficient           */
    DOUBLE S_nucleus;                   /* source coefficient             */
    DOUBLE I_nucleus;                   /* initial value coefficient      */


    VECDATA_DESC *diffsol;              /* vecdesc for solution           */
    VECDATA_DESC *derivsol;             /* vecdesc for diffusion storage  */
    INT computeDerivative;              /* switch between sol and deriv   */

    /* functions */
    INT (*Diffusion)
         (/*struct np_nucleus_param *,*/    /* pointer to (derived) object    */
          DOUBLE *,                     /* position vector                */
          /*INT ,*/                       /* subdomain id                   */
          DOUBLE *);                    /* result tensor                  */
    INT (*VelocityVector)
         (/*struct np_nucleus_param *,*/    /* pointer to (derived) object    */
          DOUBLE *,                     /* position vector                */
          /*INT ,*/                       /* subdomain id                   */
          DOUBLE *);                    /* result vector                  */
    INT (*Source)
         (/*struct np_nucleus_param *,*/    /* pointer to (derived) object    */
          DOUBLE *,                     /* position vector                */
          /*INT ,*/                       /* subdomain id                   */
          DOUBLE *);                    /* result vector                  */
    INT (*InitialValues)
         (/*struct np_nucleus_param *,*/    /* pointer to (derived) object    */
          DOUBLE *,                     /* position vector                */
          /*INT ,*/                       /* subdomain id                   */
          DOUBLE *);                    /* result vector                  */
    INT (*BndCndFunction)
         (/*struct np_nucleus_param *,*/    /* pointer to (derived) object    */
          DOUBLE *,                         /* space time vector              */
          DOUBLE *,                         /* result vector (bnd vals)       */
          INT *);                           /* result vector (bnd types)      */
    INT (*InnerBndCndFunction)
         (/*struct np_nucleus_param *,*/    /* pointer to (derived) object    */
          DOUBLE *,                         /* space time vector              */
          DOUBLE *,                         /* result vector (bnd vals)       */
          INT *);                           /* result vector (bnd types)      */
};
```

Figure 4.7: *Implementation of the numerical procedure for the parameter estimation program.*

```
/*Added for calculation of rhs in derivative of Nucleus diffusion model*/
/*here rhs is the solution of diffusion model, multiplied with Ansatzfunction and normal*/
/*NEW -- Gillian 16072007 */
if ((bflow->params->computeDerivative)==1)
{
    //DOUBLE gradNk[DIM];
    DOUBLE gradDiffU[DIM];
    //DOUBLE fluxV2;
    for (k=0; k<geo->n_scvf; k++)
    {
        /* get connection (Daten des aktuellen scvf)*/
        scvf = geo->scvf+k;

        /* get end points of connection (Nummern der Endpunkte)*/
        i = scvf->i; j = scvf->j;

        /* product of sol*normal*gradNk: */

        GradientFEFunction(DIM, geo->tag, scvf->ip_local, scvf->sdv.Jinv, dat->co_diffu, gradDiffU); /* grad u at ip k */
        V_DIM_SCALAR_PRODUCT(gradDiffU, scvf->normal, flux);

        LocalRhs[i] += flux; LocalRhs[j] -= flux;
    }
}
```

Figure 4.8: *Modification in the definition of the "right hand side" in the discretization of the parameter estimation model.*

```
/* constructor function */

static INT NucleusParamsConstruct (NP_BASE *theNP)
{
    NP_NUCLEUS_PARAM *np;

    np = (NP_NUCLEUS_PARAM *) theNP;

    /* base functions */
    theNP->Init             = NPNucleusParamsInit;
    theNP->Display          = NPNucleusParamsDisplay;
    theNP->Execute          = NPNucleusParamsExecute;

    /* nucleus problem functions */
    np->Diffusion           = NucleusDiffusion;
    np->VelocityVector      = NucleusVelocityVector;
    np->Source              = NucleusSource;
    np->InitialValues       = NucleusInitialValues;
    np->BndCndFunction      = NucleusBoundary;
    np->InnerBndCndFunction = NULL;

    return(0);
}
```

Figure 4.9: *Constructing the parameter estimation problem in UG.*

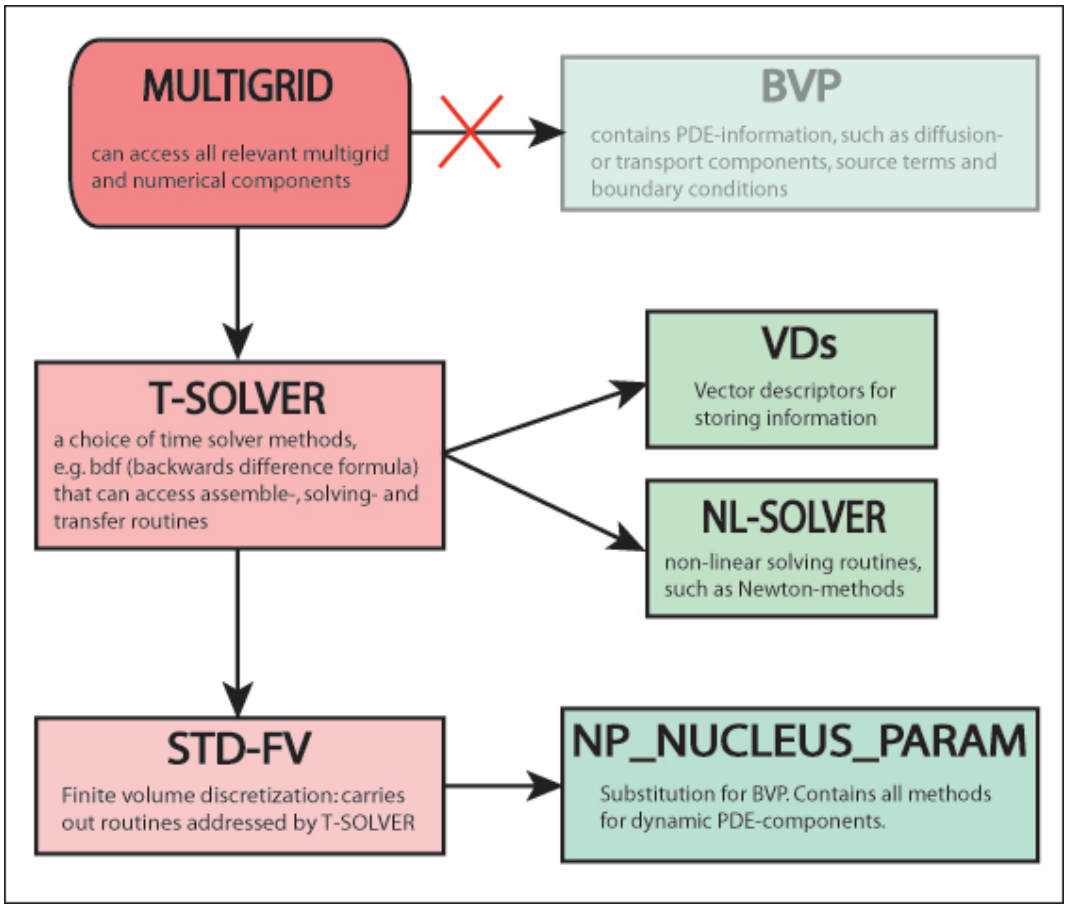


Figure 4.10: *The modified structure of the parameter estimation function.*

# Chapter 5

# Results

## 5.1 Biological Results

In close collaboration with the Bading lab of the IZN (Heidelberg) the morphologies of hippocampal neuron nuclei were investigated. In microscopy experiments, both fluorescence microscopy (M. Wittmann) and electron microscopy (A. Hellwig),membrane structures inside many nuclei became visible, Figure 5.1. The elec-

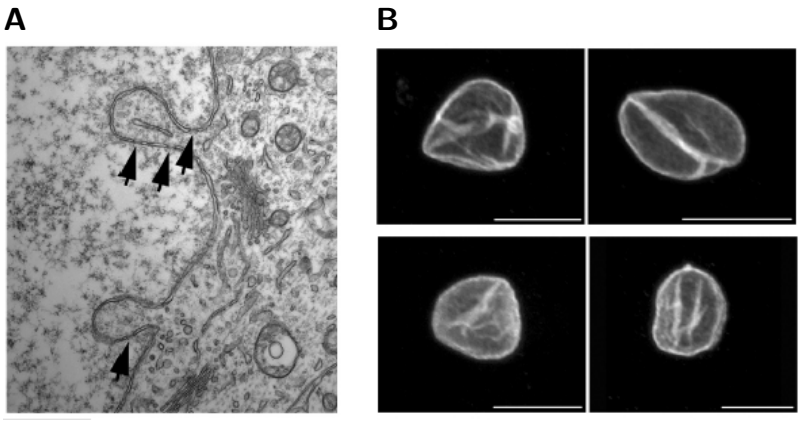


Figure 5.1: *Hippocampal neuron nuclei contain infoldings of the nuclear envelope: (A) Electron microscopy evidence for nuclear pore complexes on the infolded envelope (A. Hellwig). (B) Confocal recordings of laminB marked infolded nuclei (M. Wittmann).*

tron microscopy data shows, that not only the inner membrane, but also the outer membrane of the nucleus follows these infoldings. This contradicts theories of the existence of a "nucleoplasmic reticulum" which would be formed if only the inner of both membranes was infolded.
After verification of the existence of infolded structures of the nuclear membrane, further experiments were carried out by M. Wittmann, [67]. The question of the underlying biological processes responsible for the formation (and possible decomposition) of nuclear infoldings was assessed. A first result was that toxic levels of extra-synaptic NMDA induced in the cell led to a rapid change in the morphology of nuclei. Within a time span of approximately 15 minutes, infolded nuclei lost their structure and swelled up to a spherical form, Figure 5.2.
The contrary effect was demonstrated when activating synaptic NMDA. The measured base-level of infolded nuclei lies at approximately 30% of cells. Activation of synaptic NMDA channels raised the count of infolded nuclei to about 70%, Figure 5.3. After eliminating other possibilities that might be responsible for the 40% increase, it could be verified that solely synaptic NMDA receptors were responsible for the new formation of infoldings, [67].
Next, the stability of the 40% newly formed infoldings was assessed. To that extent, two different experiments were done, first neurons were treated for 1h with bicuculline following immediate blockage of synaptic activity with TTX, in a second

experiment they were treated for 1h with bicuculline and, following a 40h pause, addition of TTX. Direct application of TTX led to rapid decay of infoldings within approximately one hour. Yet in the second case, the 40h pause before applying TTX stabilized the infoldings, that even after the application of TTX levels of infolded nuclei stayed high, [67]. These experiments show, that neuronal activity (only in the second experiment) is necessary for stabilizing newly formed nuclear morphologies. After a number of days, nuclear infoldings have stabilized entirely, which is an indication for a transcriptional or translational mechanism responsible for the stabilization. Therefore, a specific – still unknown – set of genes might be responsible for the morphological changes of the nucleus.
One of the changes due to reorganization of the nuclear geometry is a change in the number of pore complexes on the membrane. Nuclear Pore Complexes (NPCs) are the gateway for cytosolic molecules into the nucleus and are therefore a regulating parameter for processes taking place in the nucleus. To test whether the number of NPCs is affected by the formation of nuclear infoldings, densitometric analyses of NPCs were carried out with infolded and non-infolded nuclei, [67]. The result of these experiments was, that in unstimulated neurons the pore count of infolded nuclei was approximately 15% higher than in spherical nuclei. That the formation of NPCs takes a certain time is proven by the investigation of active neurons. Nuclei with newly formed infoldings showed only an increase of 1.4% of NPCs compared to spherical nuclei.
In [67] some cases were recorded where bicuculline-induced calcium spikes carried much higher amplitudes in infolded nuclei compared to spherical nuclei. Since both the nuclear and cytosolic signal were larger, this points to an increase of transcription due to nuclear infoldings. Furthermore, patch clamp studies showed a correlation between recurrent activity of neurons and their nucleus containing infoldings, as well as a correlation between randomly active neurons and their nucleus being spherical, Figure 5.4.

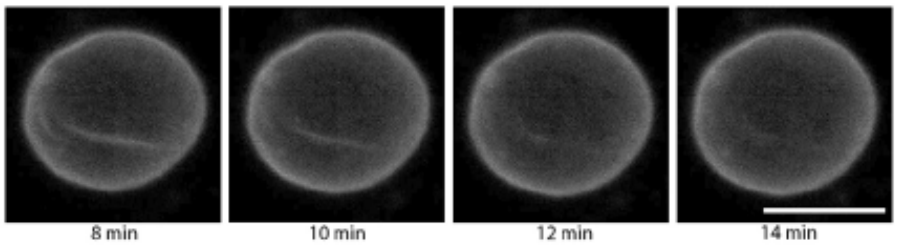


Figure 5.2: *Toxic levels of extra-synaptic NMDA lead to loss of nuclear structure, [67].*

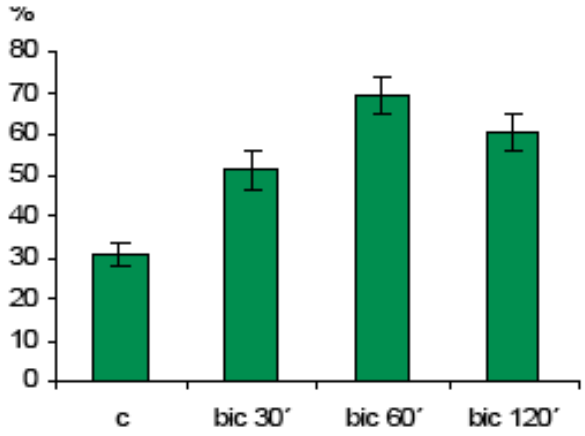


Figure 5.3: *Synaptic NMDA raises the infolded nuclei density from 30% to 70% after one hour of activation, [67].*

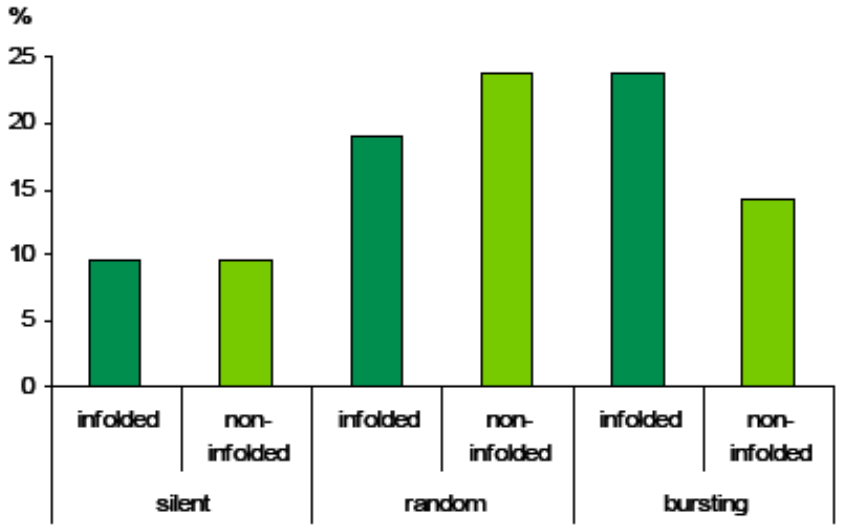


Figure 5.4: *Active, bursting cells are more likely to have an infolded nucleus than silent cells, [67].*

## 5.2 Image Reconstruction of Neuron Nuclei

Image processing is part of the preprocessing for simulating calcium signaling in nuclei. Novel concepts in image filtering based on a diffusion equation equipped with inertia-based directional filtering were developed in order to produce reconstructions of the three-dimensional structure of hippocampal neuron nuclei.
Next to linear filters, numerous techniques in nonlinear anisotropic diffusion filtering were developed, two prominent methods by Perona/Malik, [54], and J. Weickert,

[66]. This section will present results from the inertia-based diffusion filter on test geometries as well as applications on real microscopy data and comparisons with the above state of the art nonlinear filters.

### 5.2.1 Inertia-Based Filter Optimizes Parameters Simultaneously

The inertia-based filter described in Chapter 2 was developed to filter two-dimensional substructures, i.e. the nuclear envelope. By defining a diffusion filter with the equations

$$\frac{\partial u}{\partial t} = div(D(\nabla u_\sigma) \cdot \nabla u) \qquad \text{on } \mathbb{R}^+ \times \Omega \tag{5.1}$$

$$u(x,0) = \text{raw microscopy data} \qquad \text{on } \Omega \cup \partial\Omega \tag{5.2}$$

$$(D(\nabla u_\sigma) \cdot \nabla u) \cdot \vec{n} = 0 \qquad \text{on } \mathbb{R}^+ \times \partial\Omega \tag{5.3}$$

and adding structure detection with the inertia-based diffusion tensor

$$D(u) := T^t \begin{pmatrix} 1 & & 0 \\ & 1 & \\ 0 & & \varepsilon \end{pmatrix} T, \tag{5.4}$$

with $T$ a transformational matrix and $\varepsilon$ a near zero value, a nonlinear anisotropic diffusion filter was constructed, see [55, 56] and Chapter 2, to optimally deal with certain parameters.
The global objectives of closing gaps in the nuclear envelope, keeping the membrane diameters constant and eliminating background noise was achieved best by the inertia-based filter. To verify this statement, a number of tests were carried out. Recall the two anisotropic filters with their defined diffusivity functions $g(s)$:

1. Perona/Malik diffusivity: $g(s) := \frac{c_{PM}}{1+(s/\lambda)^{\alpha+1}}$
   and parameters $c_{PM}$, $\alpha > 0$ and $\lambda$.

2. Weickert diffusivity: $g(s) := 1 - \exp\left(\frac{c_W}{(s/\lambda)^m}\right)$
   and constants $c_W$, $m$ and $\lambda$.

In a first test, the capability of these two filters to close gaps in respect to keeping diameters constant and reducing background noise was tested. To this end, the free parameters $\lambda$, $m$, $c_{PM}$, $c_W$ and $\alpha$ were varied and the effect of the filters recorded in order to find ideal settings for the Perona/Malik and Weickert filter.
For parameter tuning of the Weickert filter, $\lambda$ was varied while keeping $m$ fixed, in a second series, $m$ was varied while $\lambda$ was fixed. The parameter $c_W$ was set to $-1$. Varying $c_W$ has the same effect as varying $\lambda$, so it can be neglected in the optimization series. Tables 5.1 and 5.2 show the results of varying the free parameters in the Weickert diffusivity function.

| $\lambda$ | $10^{-4}$ | 0.1 | 0.3 | 0.7 | 1.0 | Inertia-based |
|---|---|---|---|---|---|---|
| minimum gray value | 22 | 23 | 24 | 59 | 51 | 80 |
| diameter [pixels] | 9 | 9 | 8 | 15 | 20 | 9 |

Table 5.1: *Results of Weickert filter on gap and membrane diameters in respect to the choice of $\lambda$, while $m = 4$ and $c_W = -1$ compared to inertia-based filter. The minimum gray value was measured in the gap to be closed by the filter. Best results are achieved when maximizing the minimum gray value.*

| $m$ | 1 | 2 | 4 | Inertia-based |
|---|---|---|---|---|
| minimum gray value | 49 | 47 | 24 | 80 |
| diameter [pixels] | 13 | 10 | 8 | 9 |

Table 5.2: *Results of Weickert filter when changing the parameter $m$; $\lambda$ is set to 1 and $c_W = -1$.*

The minimum gray value refers to the minimum when plotting the profile of the gap. The diameter is measured by a profile plot perpendicular to the membrane of the test geometry, which is a sphere with holes of different diameters.
What becomes clear from Tables 5.1 and 5.2 is that the inertia-based filter is capable of optimizing minimum gray value and diameter at the same time, while the Weickert filter can be tuned is such a way to optimize the minimum gray value, but only at the cost of broadening the diameter. This effect occurs when tuning the parameter $\lambda$ (Table 5.1) as well as when tuning $m$ (Table 5.2).

Similar results are observed for the Perona/Malik filter. When fixing $\alpha = 1$ and $c_{PM} = 1$, the parameter $\lambda$ can be varied and the different results of the filter recorded. Table 5.3 shows these results. Also when manipulating $\alpha$, the Perona/Malik filter cannot be tuned to optimize the given two parameters simultaneously, Table 5.4.
The values of the inertia-based filter in the above tables show, that structure detection using the inertial tensor is a powerful method when trying to identify a substructure in a 3D-image set that needs to be filtered without affecting its original structure. Figure 5.5 (A) shows, that the three nonlinear anisotropic filters tested have a similar effect on the membrane diameter and perform better than linear filters, such as the Gaussian blur.
Setting the Perona/Malik and Weickert filter, so they do not broaden the membrane

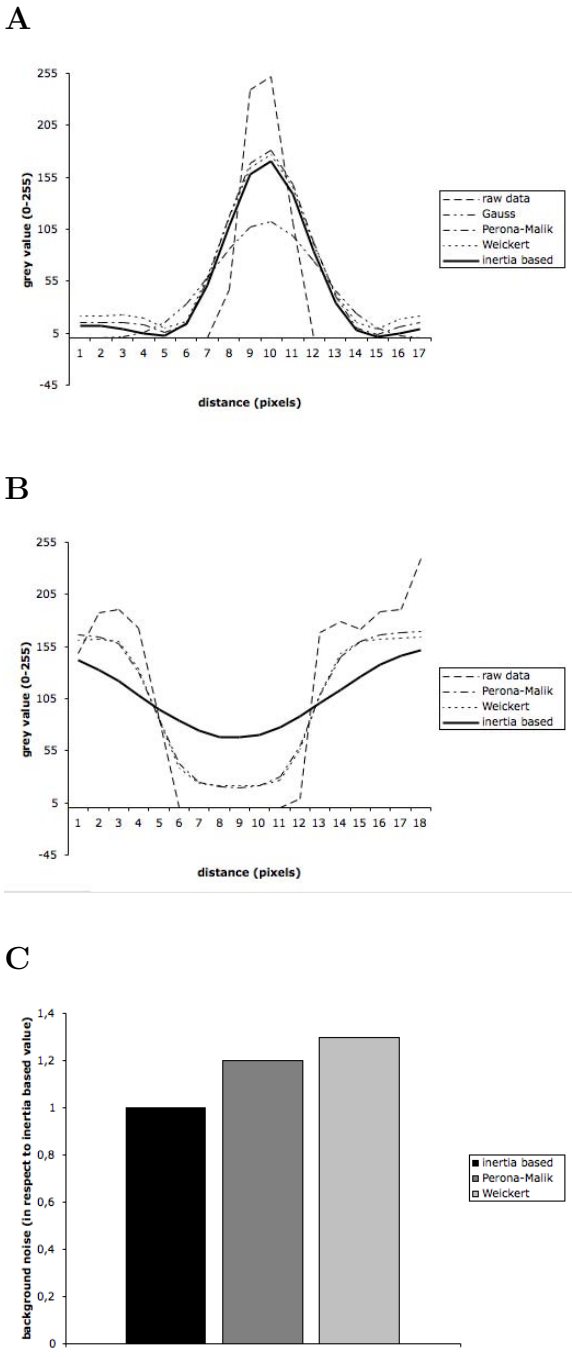


Figure 5.5: *(A) Line plot perpendicular to membrane of the gray value profile of raw data and after application of different filters. (B) Line plot parallel and across a hole in the membrane of the gray value profile of raw data and after application of three different nonlinear diffusion filters. (C) Background noise levels of different nonlinear diffusion filters in respect to inertia-based diffusion filter.*

| $\lambda$ | $10^{-4}$ | 0.1 | 0.3 | 0.7 | 1.0 | Inertia-based |
|---|---|---|---|---|---|---|
| minimum gray value | 33 | 18 | 30 | 51 | 52 | 80 |
| diameter [pixels] | 7 | 7 | 9 | 15 | 18 | 9 |

Table 5.3: *Effect of Perona-Malik filter on gap and membrane diameters in respect to variation of the parameter $\lambda$ while $\alpha = 1$ and $c_{PM} = 1$.*

| $\alpha$ | 1 | 3 | 5 | Inertia-based |
|---|---|---|---|---|
| minimum gray value | 30 | 33 | 31 | 80 |
| diameter [pixels] | 9 | 9 | 9 | 9 |

Table 5.4: *Perona-Malik filter results in respect to changes in the parameter $\alpha$ with $\lambda = 0.3$ and $c_{PM} = 1$.*

diameter more than the inertia-based filter, Figure 5.5 (B) demonstrates that the inertia-based filter performs significantly better than the compared two nonlinear filters.
Furthermore, one can witness a better reduction of noise when applying the inertia-based filter, see Figure 5.5 (C). Both Perona/Malik and Weickert filters lack differentiation between gaps and background noise. This means, that the minimum gray values reached by those filters are in the range of the background intensity, therefore make gaps indistinguishable from background for further segmentation processing.
A full image processing cycle consists of filtering the nucleus image stack with optimal settings, see [55], segmentation of the data with an Otsu segmentation algorithm and following that, a second pass of the filter. The results of these steps are seen in Figure 5.6.
Starting with a raw data set, (A), the image is filtered with the inertia-based filter, (B). For comparison, (C) shows the same image filtered with the Gaussian blur. After filtering, the image is turned into a black and white image, therefore eliminating background noise, (D). Finally, a second pass of the filter produced the final image (E) which can be used to create surface and volume grids.

### 5.2.2 3D Image Reconstructions and Measuring Nuclei

After closing gaps in the membrane of the nuclei by applying filter and segmentation techniques, an isosurface can be generated from the preprocessed image stack. As

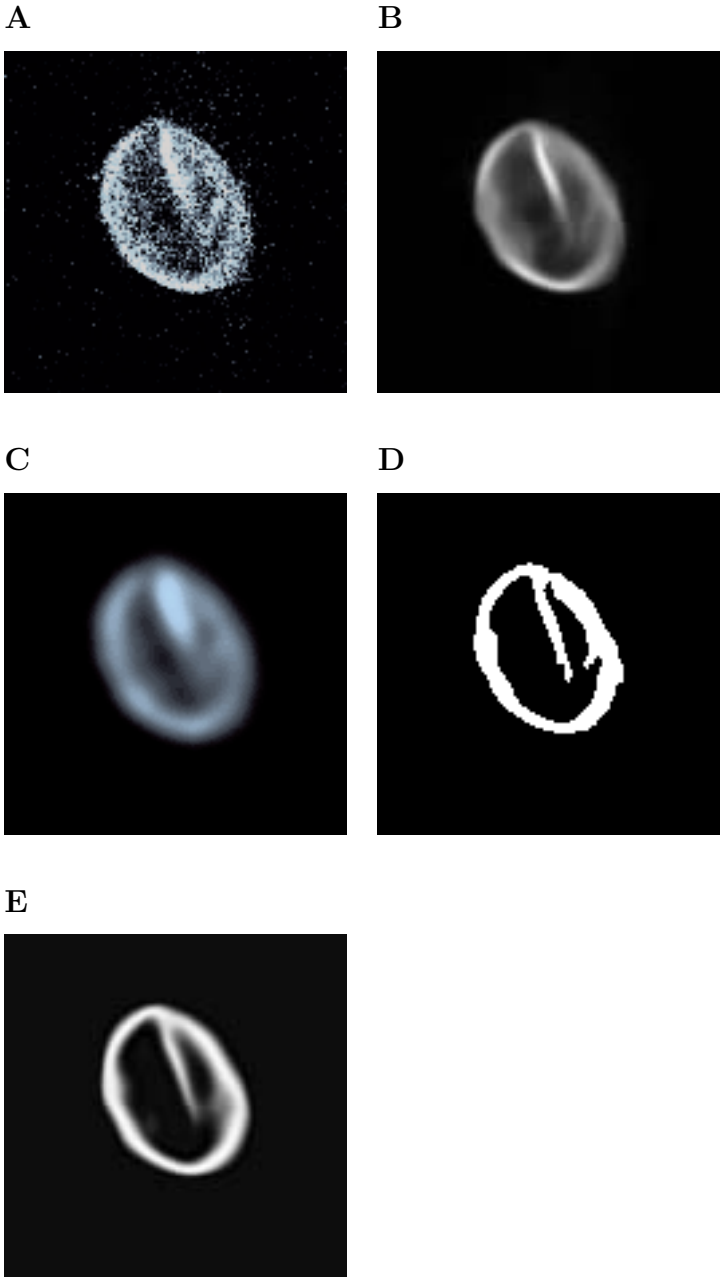


Figure 5.6: *(A) Raw image. (B) Filtered with nonlinear anisotropic filter for surfaces. (C) Filtered with gaussian blur. (D) Filtered with anisotropic filter and segmented with Otsu-method. (E) Second pass of anisotropic filter on segmented image.*

described in Chapter 2 a surface grid is generated from a defined gray value and by meshing the calculated surface nodes to a closed triangulation. Examples of reconstructed nuclei are shown in Figure 5.7 and the Appendix.
By applying grid optimization tools, the density of the surface grid can be manipulated and grid nodes repositioned to produce minimal deviance between triangle sizes. Zooming in on a nucleus mesh, Figure 5.8 demonstrates the effect of surface grid optimization.
Having reconstructed a large data set of infolded and spherical nuclei, the surface area and volume of these nuclei can be measured. This gives answer to the question posed in Chapter 2, whether surface area increases when infoldings of the membrane are formed, or if the nucleus reacts like a deflating ball, where surface area stays constant and the volume of the nucleus decreases. Figure 5.9 shows the results of 52 nuclei measured, 31 infolded and 21 control (spherical) nuclei.
Where the volumes of infolded and control nuclei are about the same, significant increase in the surface area of infolded nuclei can be measured. This indicates, that during the process of forming new infoldings, the cell generates or recruits new membranes for a morphological change.

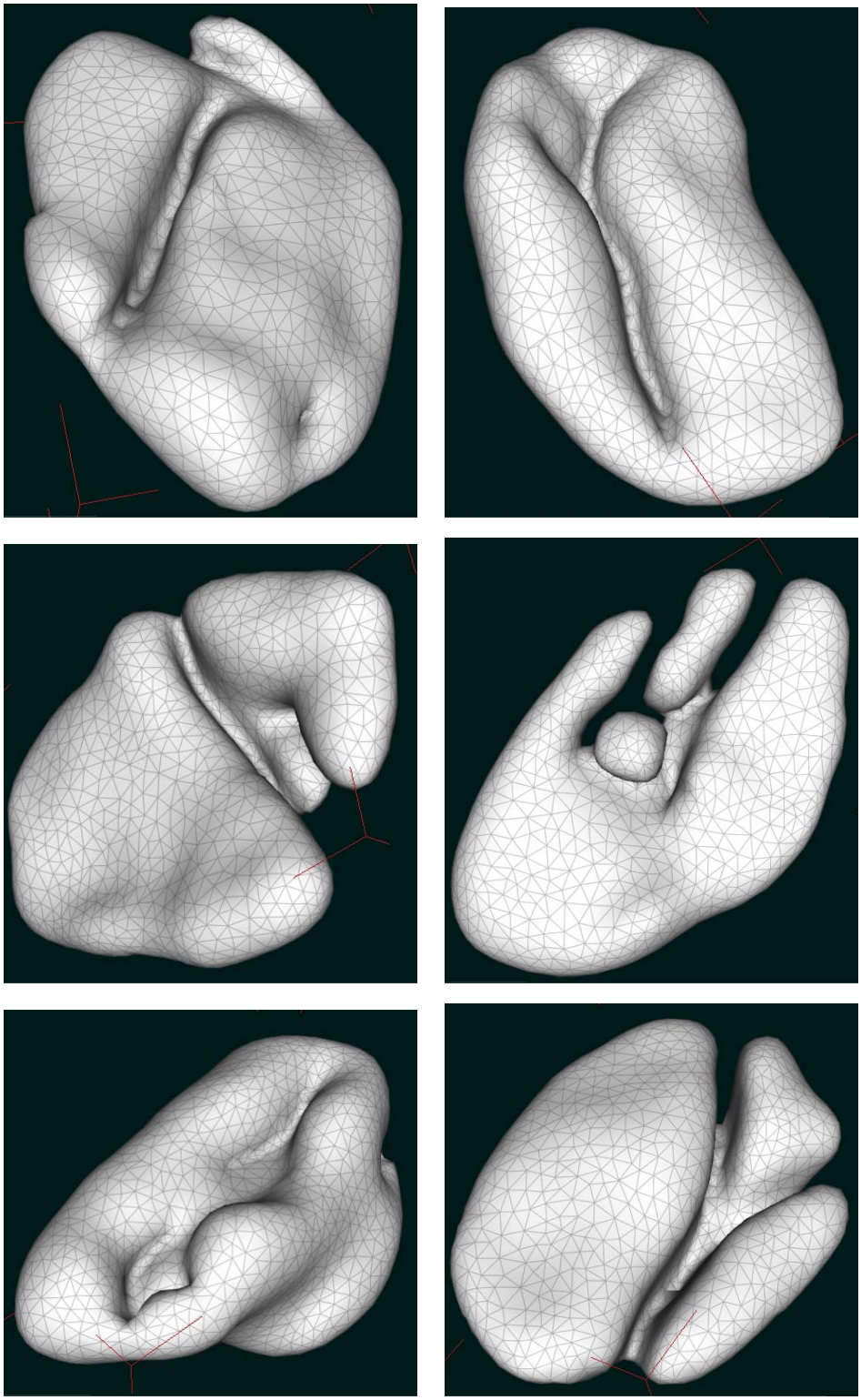

Figure 5.7: *Examples of three dimensional reconstructions of nuclei from hippocampal neurons.*

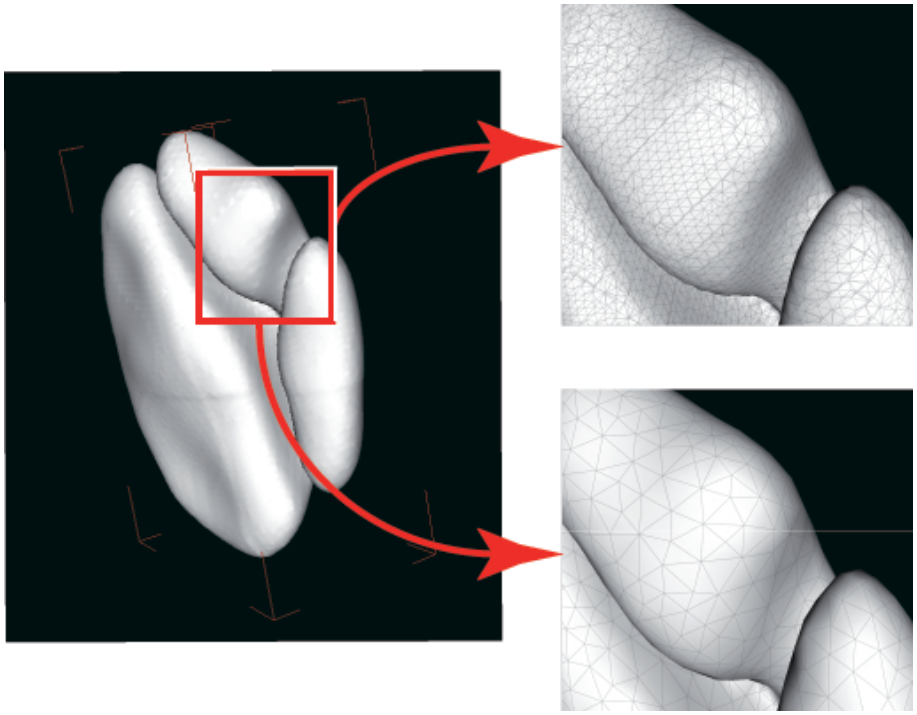

Figure 5.8: *Results of grid optimization. Top right: before grid optimization, Bottom right: after grid optimization.*

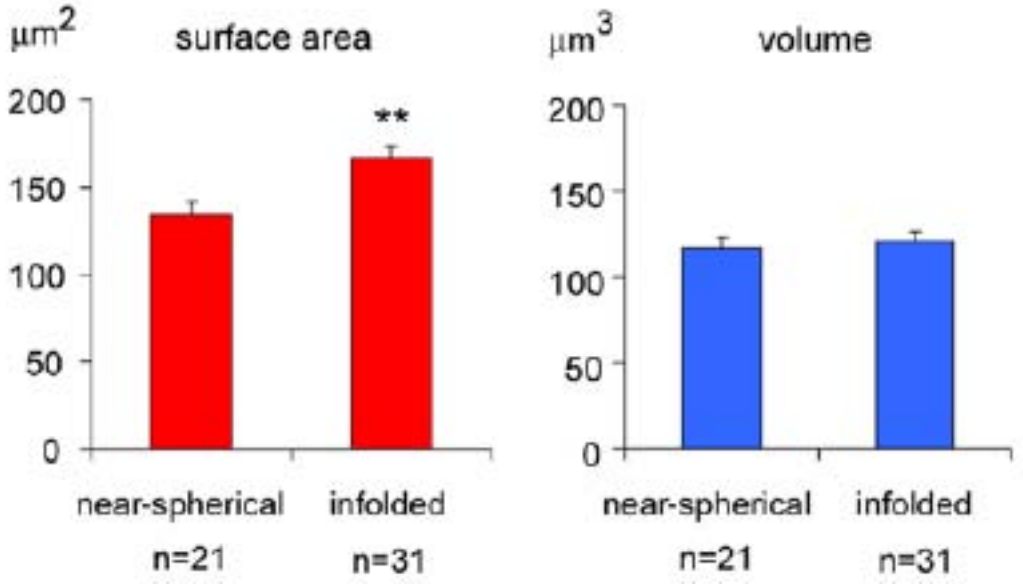


Figure 5.9: *Changes in surface and volume: Infolded nuclei have a larger surface, while the volume of infolded and spherical nuclei is approximately the same.*

## 5.3 Calcium Signaling Depends on Characteristic Morphologies

In this and the following section the core question of this thesis will be addressed, namely the question of the influence of the nuclear morphology on calcium signaling in the nucleus. The simulations and tests done in this section are based on test geometries, infolded and spherical, where radius and infolding depth are adjustable. This allows systematic investigation of the effect of different infolding depths, increasing or decreasing number of nuclear pores, and the evaluation of calcium load dynamics as well as threshold dependent processes under varying parameters. As the title of this section states, the interpretation of a calcium signal in a nucleus is dependent on the form and structure of the nuclear envelope.

### 5.3.1 Infoldings Minimize Diffusion Distances

As stated in Chapter 3, the propagation of a calcium signal in hippocampal neurons is driven by the concentration gradient of calcium. Therefore, one vital parameter for this process is the diffusion distance. The stronger the calcium signal is buffered for instance, the shorter the active range for calcium becomes, thus possibly excluding biologically relevant sites. In order for the cell to regulate where a calcium signal should be directed and also to be able to influence the amplitude and frequency of the signal, it can regulate the diffusion distances that a calcium signal needs to travel.
Intuitively, the nuclear membrane infoldings affect diffusion distances in the nucleus. As a measure for the effect of a nuclear infolding, one can measure calcium concentration at the most distal location from the membrane inside the nucleus. Figure 5.10 illustrates the dislocation of the measuring point depending on the size of the nuclear infolding. It also becomes visible, that an infolding automatically changes the entire characteristic of a nucleus. In a spherical nucleus, there exists only one "most distal" point, whereas the moment the envelope is infolded, the nucleus contains a "band" of distal points where the minimum calcium load values are reached.

**Simulation set-up and results:**

Simulations are carried out on varying geometries, Figure 5.10. As boundary settings for the model, the continuous mode is activated, see Section 3.6 and a single CCT burst is chosen as the cytosolic signal. Following the time course after triggering the CCT, the calcium concentration is measured at the most distal point in the nucleus. In order to compare the effects of different nuclear morphologies on the calcium dynamics in the nucleus, the calcium load at a specific time point is plotted against the infolding length. In a second simulation the maximum value of calcium concentration in the distal point is plotted against the infolding length, Figure 5.11.

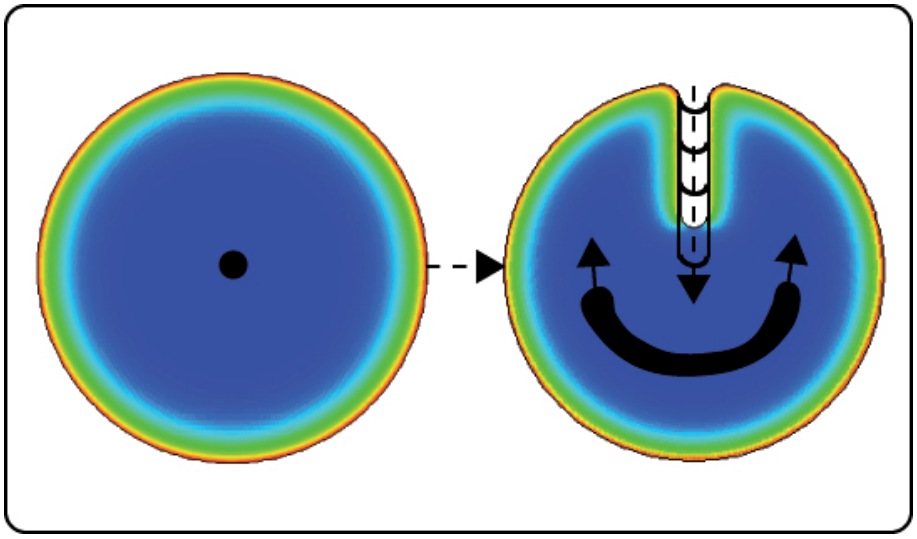

Figure 5.10: *Change of maximum distance from the membrane due to the existence and size of an infolding.*

Next, the infolding length was decreased in each simulation, until reaching a spherical structure. A last simulation was done on a spherical geometry with its radius set to a value that produces the same circumference as the infolded geometry, with a 25% surface increase compared to Circle 1, Figure 5.12.

The results show, that with increasing infolding size, the amplitude of the calcium signal at the distal point (which is the minimum of the calcium concentration) increases, Figure 5.11. Once the center of the nucleus is reached by the infolding, the amplitude of the calcium load minimum stays constant because the maximal diffusion distances in the nucleus are not changed by the infolding from that point on.
In Figure 5.12 the time course of a calcium signal in the nucleus at the distal nuclear site as a result of a CCT burst is depicted. Depending on the size of the infolding of the envelope, the amplitude of the signal is affected with a more that threefold difference between a spherical structure and one with an infolding extending to the center of the nucleus.
In conclusion to this study, infoldings of the nuclear envelope decrease diffusion distances in the nucleus, therefore raising the overall amplitude of the calcium signal and optimizing the nuclear structure so that distal regions of the nucleus are stronger addressed by an incoming calcium signal.

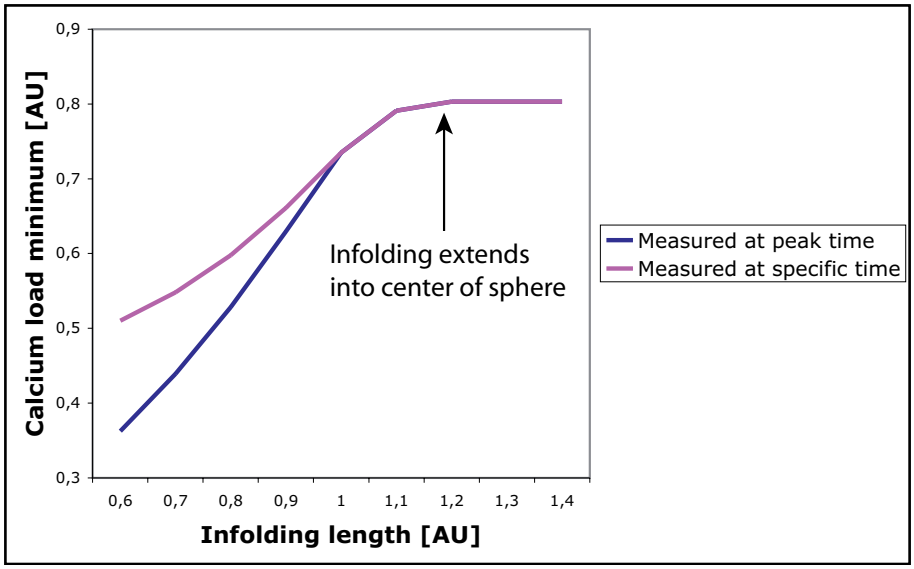


Figure 5.11: *Changes in the length of an infolding affect the calcium load minimum. Once the infolding extends into the center of the nucleus, a steady state is reached.*

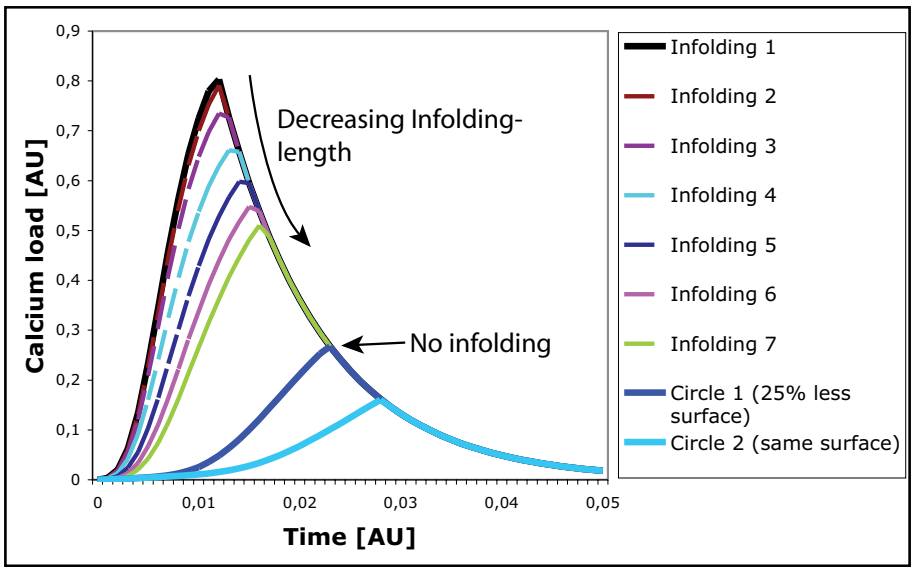


Figure 5.12: *Depending on the size or existence of an infolding, the amplitude of calcium load is changed. Large infoldings produce high calcium load amplitudes, while spherical structures have low load amplitudes.*

### 5.3.2 Calcium Load Dynamics Controlled by Number of NPCs

Calcium load is defined as the global amount of calcium in the nucleus at a given time. In the model, calcium load can be measured by summing the concentration values of calcium in each grid node, and norming it with the volume of the nucleus. Measuring calcium load along a given time course gives a measure for the dynamics of different nuclear morphologies.
In this simulation experiment, an infolded geometry is compared to a spherical one, stimulated with different single CCT bursts. The calcium load is then measured in each time step of the simulation and plotted in time, Figure 5.13. The figure indicates, that due to an infolded structure the nucleus shows faster rise and decay times than a spherical one. This also results in a change of the signal-amplitude. Variation of the primary stimuli shows, that the ratio between the calcium load amplitude of infolded and spherical geometry is largely independent of the length of CCT-input, Figure 5.14. The different types of single CCTs are specified in Section 3.4.
Yet when considering the definition of calcium load, the presented results are not directly linked to the actual morphology of different neurons, but are rather an indication of varying numbers of NPCs due to varying nuclear surface size. A spherical structure with the same surface size and number of NPCs as an infolded structure elicits similar calcium load dynamics.
Derived from these simulation results, the calcium load dynamics of a nucleus are independent from actual morphological features of the nucleus, but directly coupled to the surface size and number of NPCs located on the nuclear envelope. Infolded nuclei will offer faster passage for calcium into the nucleus.

### 5.3.3 Threshold-Dependent Biological Processes are Greatly Influenced by the Nuclear Morphology

Biochemical processes are activated fully or only to a certain degree if the inducing signal is above or below a certain threshold. In the case of nuclear calcium signaling, calmodulin plays a vital role in the relay of information. The degree of calmodulin activation in the presence of target enzymes depends on the strength of the calcium inducing signal. A calcium transient in the micro-molar range fully activates calmodulin, signals in the nano-molar range ($\approx$ 200 nm above basal level) hardly affect the state of calmodulin.
Taking this information into consideration, threshold-dependent calcium activity was simulated in the test environment. Calcium activity is determined by means of Section 3.5.3, where a threshold is defined manually and the volume above threshold is normed with the entire nuclear volume. As it is unknown where this biological threshold actually lies, the threshold was varied within a broad range, in order to estimate the tendencies of the morphological influence on calcium activity. Further

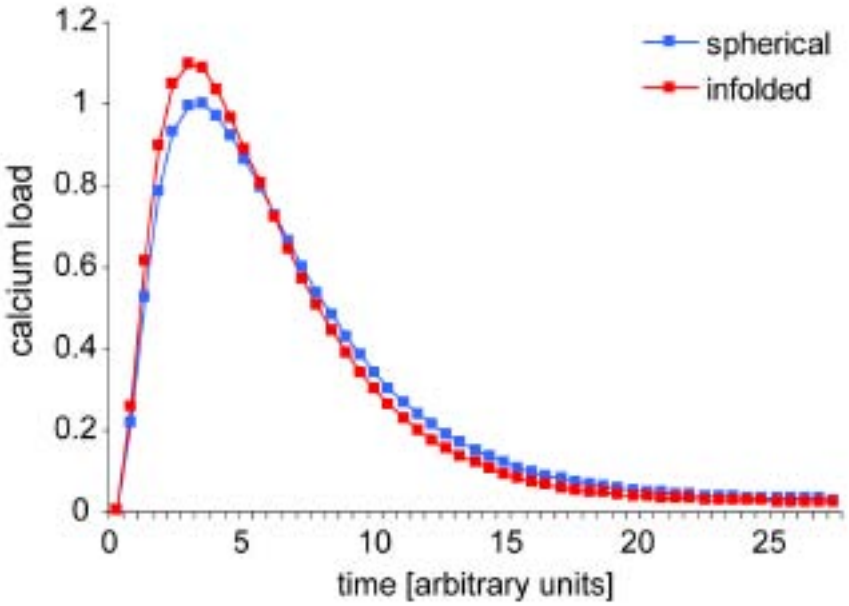


Figure 5.13: *Simulation of calcium diffusion on two dimensional models: Infolded nuclei show higher amplitudes of calcium load than spherical ones.*

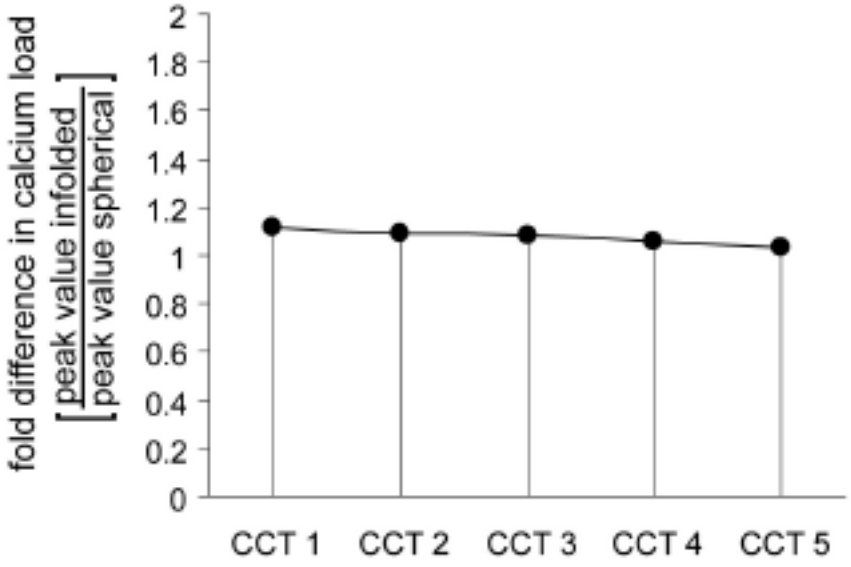


Figure 5.14: *Independent of the length of the cytosolic calcium transient the fold difference between the load amplitudes of infolded and spherical nuclei stay nearly constant.*

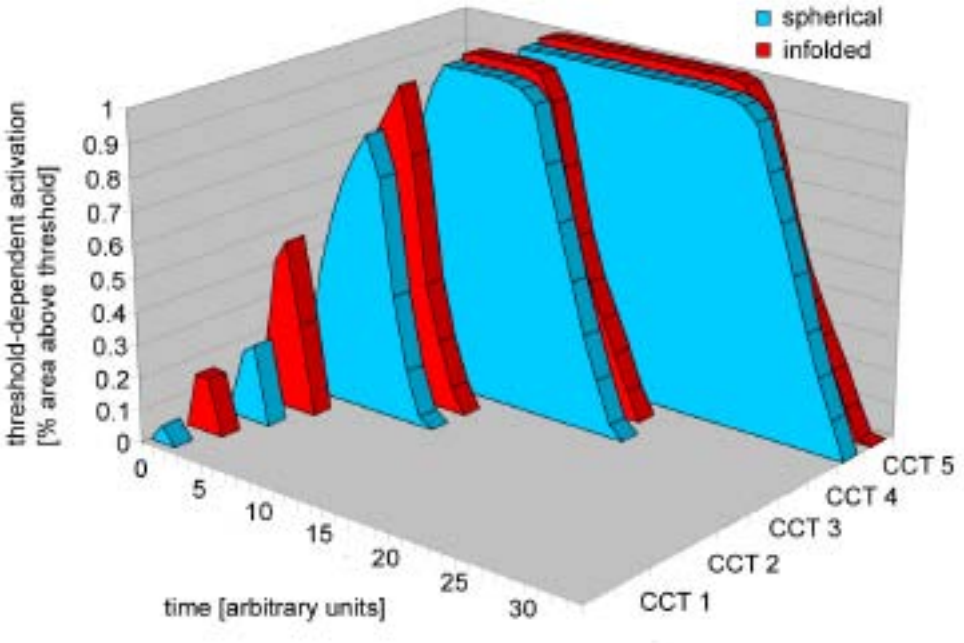


Figure 5.15: *Activity measurements in 2D simulations: Depending on the duration of the CCT infolded nuclei show up to six fold activity differences (for short CCTs).*

interest lay in the effect of different lengths of single CCT bursts on the calcium activity. Therefore, simulations were carried out with varying thresholds and varying CCTs, depicted in Figures 5.15–5.17.

Figure 5.15 shows the activity curves over a given time course with a fixed threshold (90% of the concentration maximum) and varying CCTs. When comparing spherical and infolded structures, one notices a larger difference in the nuclear activity when input signals are short (CCT 1 is the shortest input and length is doubled for each following CCT, see Section 3.4). Figure 5.16 plots the fold difference of threshold-dependent activity as a quotient of infolded activity and spherical activity in respect to the triggering CCT. This activity value is defined as the integral of the activity graphs, e.g. the ones seen in Figure 5.15.

A six fold difference in activity can be witnessed for CCT 1, and decreasing to CCT 5 the curve shows, that both geometries elicit nearly identical nuclear activity for longer CCTs. This shows, that the infolding of nuclear membrane enhances nuclear activity, especially for shortly induced signals.

Next, the input signal was fixed and the threshold varied in order to determine whether the effect of the nuclear geometry on activity-related processes is dependent on the biological threshold. Figure 5.17 contains information about the effect of a varying threshold on the fold difference in activity, again as a quotient of infolded and spherical activity. Simulation results are shown for the application of CCT 1, 2 and 3 where the threshold was varied in the range of 50% to 90% of the

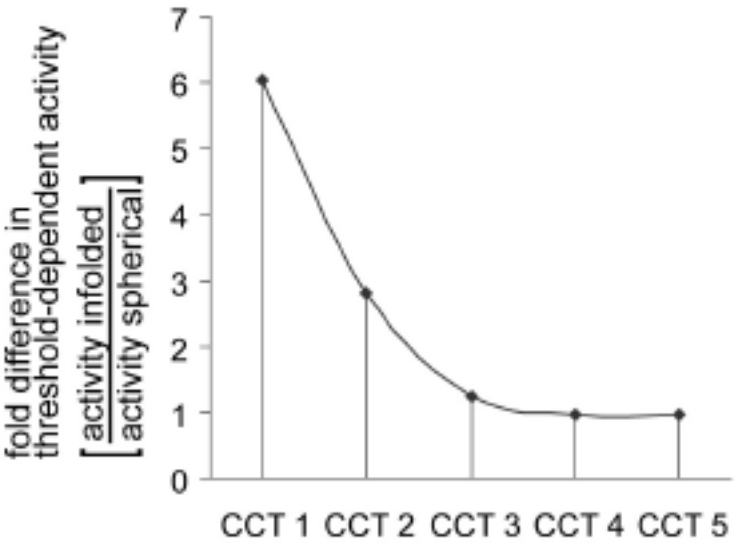


Figure 5.16: *Fold difference in threshold dependent activity: When the cytosolic stimulation is short (CCT1) infolded nuclei show 6 fold higher activity than spherical ones. The longer the stimulation is, the more infolded and spherical nuclei react the same.*

concentration maximum in the nucleus.
Nuclear activity for longer ranging CCTs (e.g. CCT 3) is nearly independent of the nuclear morphology yet the shorter the input signals are, the greater the influence of the morphology is. For low thresholds and induction of CCT 1, the infolded geometry shows twice the activity as the spherical structure. This ratio rises to a six fold difference when setting the threshold to 90%. In conclusion to these simulations, infolded nuclei are activity optimized for high thresholds, or to turn the argument around (a high threshold symbolizes a weak input), infolded nuclei are activity optimized for weak inductions.

The measure of nuclear activity, because it includes a threshold, is coupled directly to the morphology of a nucleus. This was not the case for calcium load simulations in the section above. An infolding, aside from changing the membrane surface size and number of NPCs, shortens diffusion distances and allows calcium entrance into the nucleus from opposing and closer sites which cause faster rise of the local calcium concentration, therefore increasing above-threshold nuclear volume.
Simulations of activity-dependent processes on test geometries yield the following results:

1. Nuclear calcium load is independent of the actual nuclear morphology, but is regulated by the membrane surface size and the number of NPCs located on the

membrane. Infolded nuclei, having an increased number of NPCs compared to spherical ones, show faster rise and decay times, as well as larger amplitudes in the calcium signal.

2. Nuclear activity, a measure of threshold dependent biochemical processes, is linked directly to the morphology of a nucleus.

3. Infolded nuclei optimize nuclear activity, especially for shortly induced signals as well as for high thresholds.

4. Infolded nuclei have an optimal form for activating biochemical processes with weak signals.

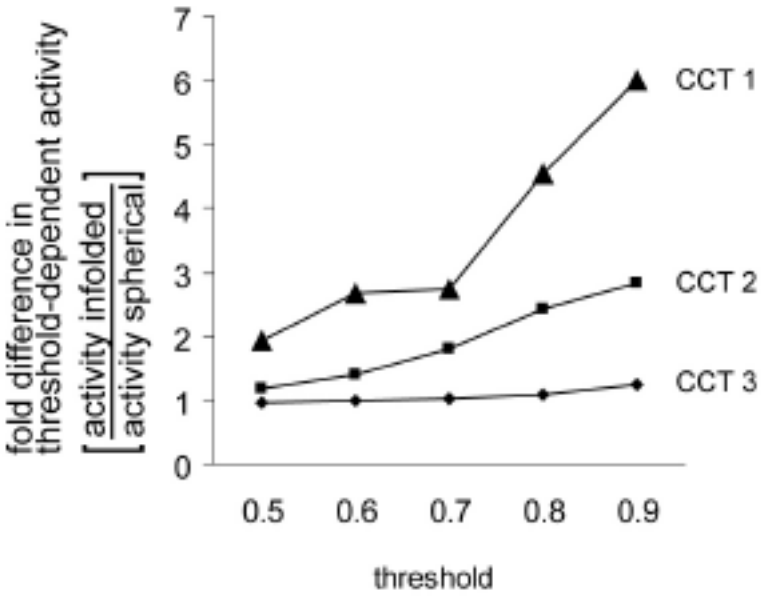


Figure 5.17: *Changing the threshold affect the activity of nuclei. Infolded nuclei are more adept at activating biochemical processes at high thresholds.*

## 5.4 Simulations on Real 3D Geometries

The core novelty of this thesis lies in the inclusion of three dimensional geometry data into a simulation framework. This geometry data is not generated artificially, but retrieved from confocal microscopy of nuclei from hippocampal neurons. Therefore it is possible to directly evaluate different morphologies of nuclei in respect to the decoding of information within these nuclei.
More abstract studies on two dimensional test geometries in the section above showed, that the form of the nucleus influences the dynamics of calcium load and nuclear activity, depending on the signal intensity or biological thresholds. This section will demonstrate, that these statements hold for simulations on real three-dimensional morphologies. The differences between spherical and infolded nuclei will be investigated systematically. In addition, 3D simulations showed, that not only spherical and infolded nuclei function differently, but also that infoldings of the nuclear membrane separate the nuclear interior, forming subunits of the nucleus that can function more or less independently from one another, depending on the degree of membrane infolding.
A further point for understanding signal processing in nuclei is the morphological effect on signal frequency decoding. The last part of this section will address this aspect.

### 5.4.1 The Difference Between Spherical and Infolded Nuclei

General differences between spherical and infolded nuclei are found in surface size and surface/volume ratios. These parameters affect calcium load dynamics and activity dependent processes. Infoldings of the membrane shorten diffusion distances, which affects the inertia of the nucleus and can be represented by calcium dynamics in the center of a nucleus. Multiple simulations were carried out by stimulating different nuclei, infolded and spherical, with varying single CCT bursts and varying thresholds. These simulations followed the objectives:

1. Differences in calcium load dynamics due to varying nuclear morphologies.
2. Morphological effects on threshold dependent processes.
3. Morphological effects on inertia of the nucleus.

In this section infolded and spherical nuclei are compared, see Figure 5.18. Signal transduction is recorded in time, Figure 5.19, and evaluated in respect to the above points.

**Differences in calcium load dynamics:**

Different CCTs activated the model, resulting in different calcium load behavior. Yet the differences between the tested geometries show to be independent of the

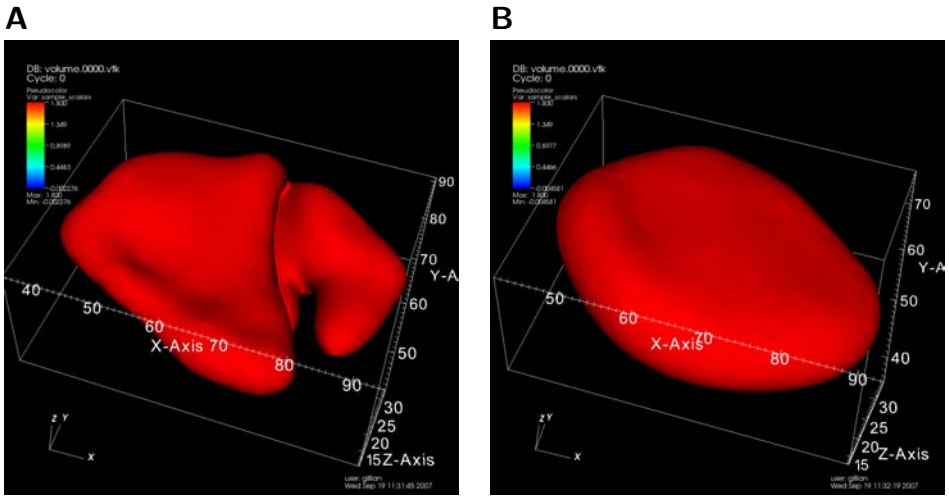


Figure 5.18: *Example of three dimensional test geometries of (A) an infolded nucleus and (B) a spherical nucleus.*

duration of the CCT, Figure 5.20. As seen on two dimensional test geometries, there is a visible shift in the amplitude of the calcium concentration together with faster rise and decay times of the signal. As mentioned earlier, this observation is correlated with an increase in surface size and a change of the surface/volume ratio.

**Threshold dependent processes:**

For the purpose of seeing what the influence of an infolded nuclear membrane has on threshold-dependent biological processes, these being biochemical reactions that take place or are only fully activated when the local calcium level exceeds a certain concentration, the nuclear activities of infolded and spherical nuclei were recorded with varying thresholds, Figure 5.21. Stimulating nuclei with a single CCT (CCT 1), infolded nuclei show higher levels of activity. This is the case for any chosen threshold. The higher the threshold is, the bigger the activity ratio between infolded and spherical becomes.
Figure 5.21 demonstrates, that for low thresholds (or strong signals) both spherical and infolded nuclei are capable of transforming the incoming signal into nuclear activity. The higher the threshold is (or the weaker the signal is), infolded nuclei are more adept at activating the nucleus. Spherical nuclei show nearly no activity at high threshold levels (or for weak signals), Figure 5.21 (D,E).

### 5.4.2 Infoldings Form Nuclear Micro-Domains

The three dimensional reconstructions of nuclei showed, that each nucleus has an individual form, somewhat like a nuclear "fingerprint". In all of the cases where

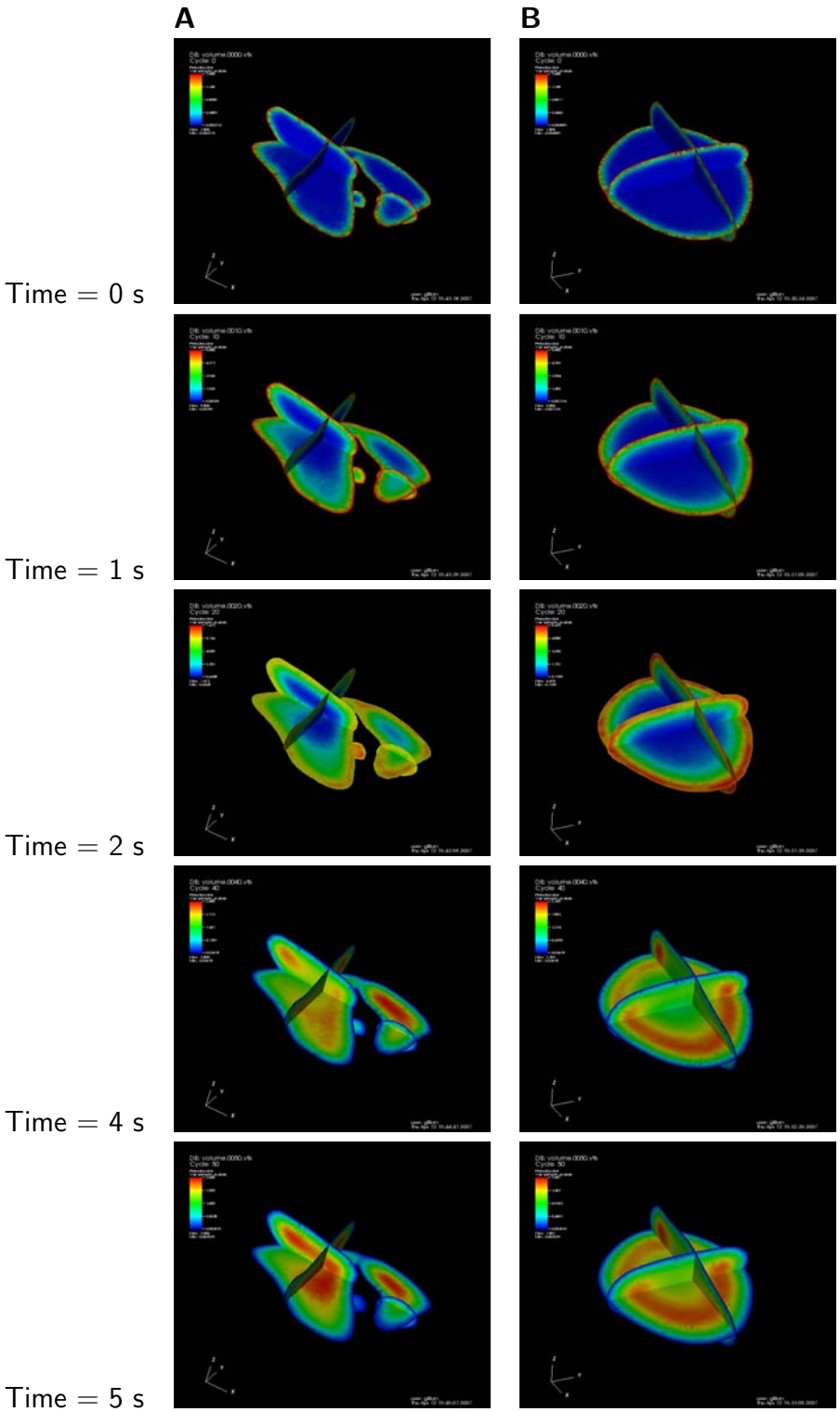


Figure 5.19: *Example of a time course of calcium diffusion into (A) an infolded nucleus and (B) a spherical nucleus.*

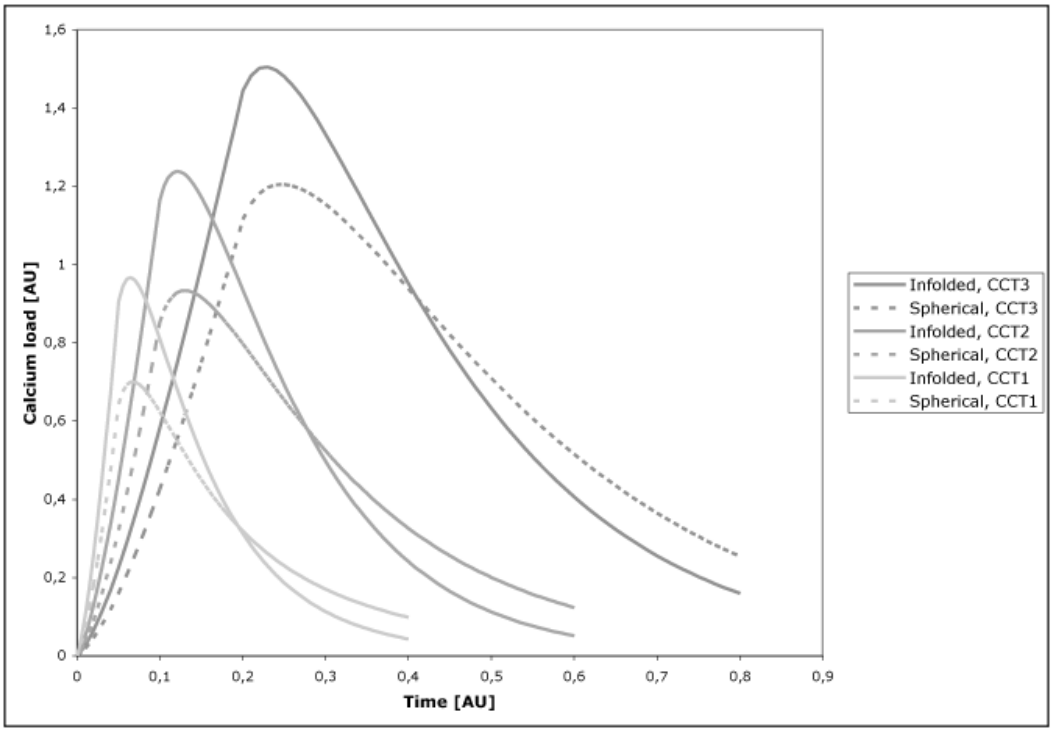


Figure 5.20: *Simulation of calcium diffusion in infolded and spherical nucleus: Calcium load amplitudes are higher in infolded nuclei, independent of the CCT (reaction to CCT1-CCT3 depicted).*

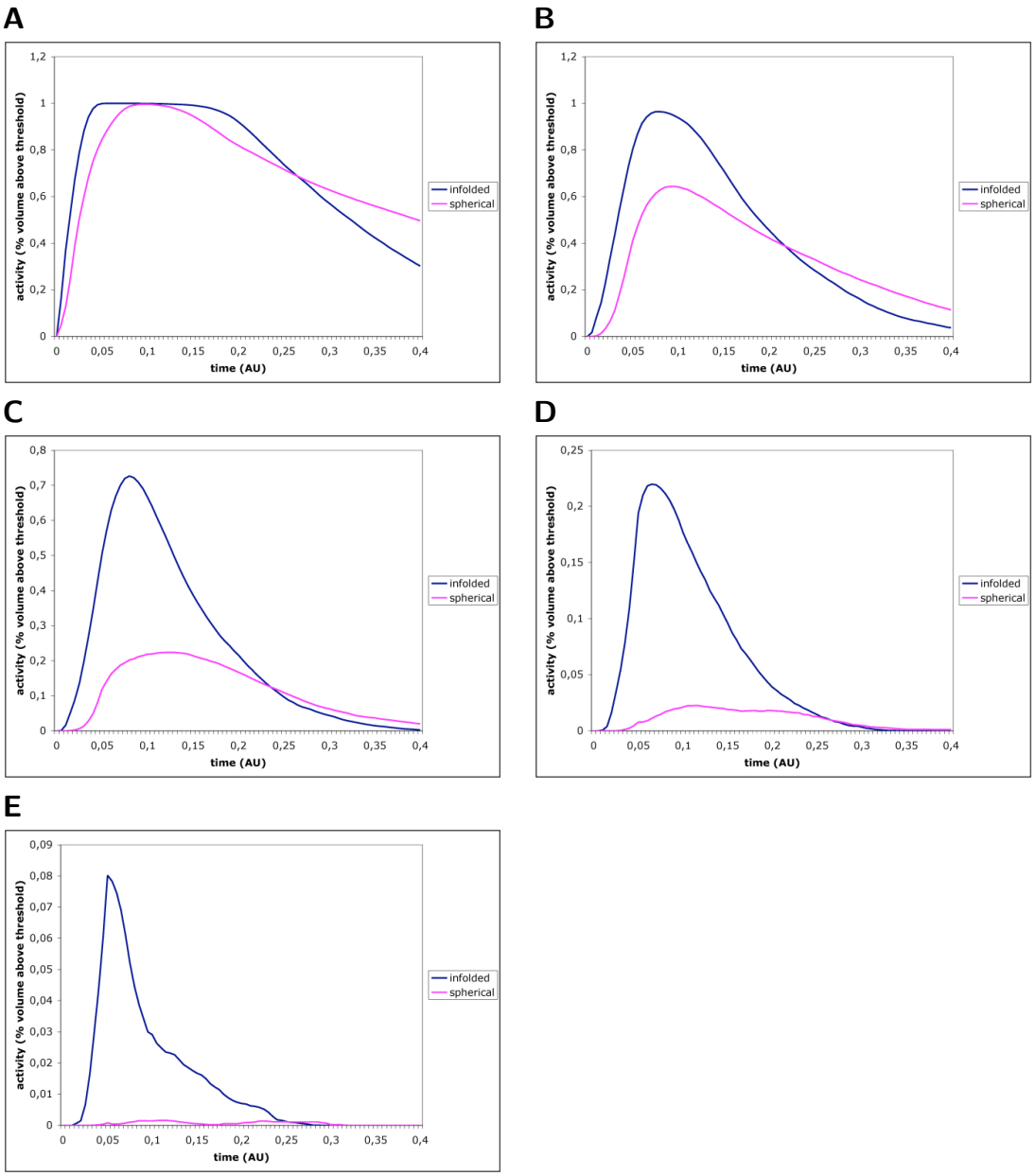


Figure 5.21: *Infolded nuclei are more active than spherical ones: With increasing threshold (from (A) to (E)) the difference between infolded and spherical nuclei becomes more prominent.*

the nuclear membrane is infolded, one can witness the formation of two or more compartments, meaning areas of the nucleus that are spatially separated by the deformation of the nuclear membrane.
Depending on the depth of the infolding, these "micro-domains" are more or less isolated from one another. This observation gave rise to the idea, that the deformation of the nuclear membrane might form not only spacial micro-domains, but also functional micro-domains. This would enable the nucleus to decode a signal coming from the cytosol in different ways.
The focus lies on investigating signal processing within infolded nuclei, and rather than comparing them to spherical nuclei, comparison takes place between spatially separated micro-domains. In order to make the model as realistic as possible, the nuclei were stimulated with calcium transients that were recorded experimentally. This model option is discussed in Section 3.4.

### 5.4.3 Activity of Micro-Domains

When activating infolded nuclei with CCTs from experimental data, micro-domain activity of the nucleus can be evaluated according to Section 3.5. Regulated by the size of the micro-domains and their surface to volume ratios, each domain expresses different responses to a CCT which enters the nucleus.
This was observed in experiments, Figure 5.22, as well as in simulations, Figure 5.23. Small micro-domains show higher activity than large ones, especially when the incoming signals are weak or the activation threshold is high. In all reconstructed nuclei, only one nucleus could be found that had symmetrical infoldings, meaning nearly every infolded nucleus is made up of different sized micro-domains, Figure 5.24.

### 5.4.4 Morphology Influences Signal Frequency Decoding

Since the frequency of calcium signals is an information encoding parameter for the cell, it is important to know how well the frequency in the nucleus is resolved. Simulations with different input frequencies show, that infolded nuclei are better at resolving higher frequency signals than spherical ones. Furthermore, the frequency of an incoming signal is better resolved in small compartments within single infolded nuclei. Figure 5.25 shows the effect of CCTs with different frequencies.
The faster cytosolic calcium transients oscillate, the harder it becomes for spherical nuclei and large compartments to resolve the full amplitude of the signal. Changing the frequency of an experimental CCT demonstrates, that at approximately 1 Hz spherical nuclei fully lose their ability of resolving single bursts and go into a plateau state, whereas infolded nuclei still resolve each burst within a calcium transient, Figure 5.26.

To address the point of frequency decoding, simulations and experiments were car-

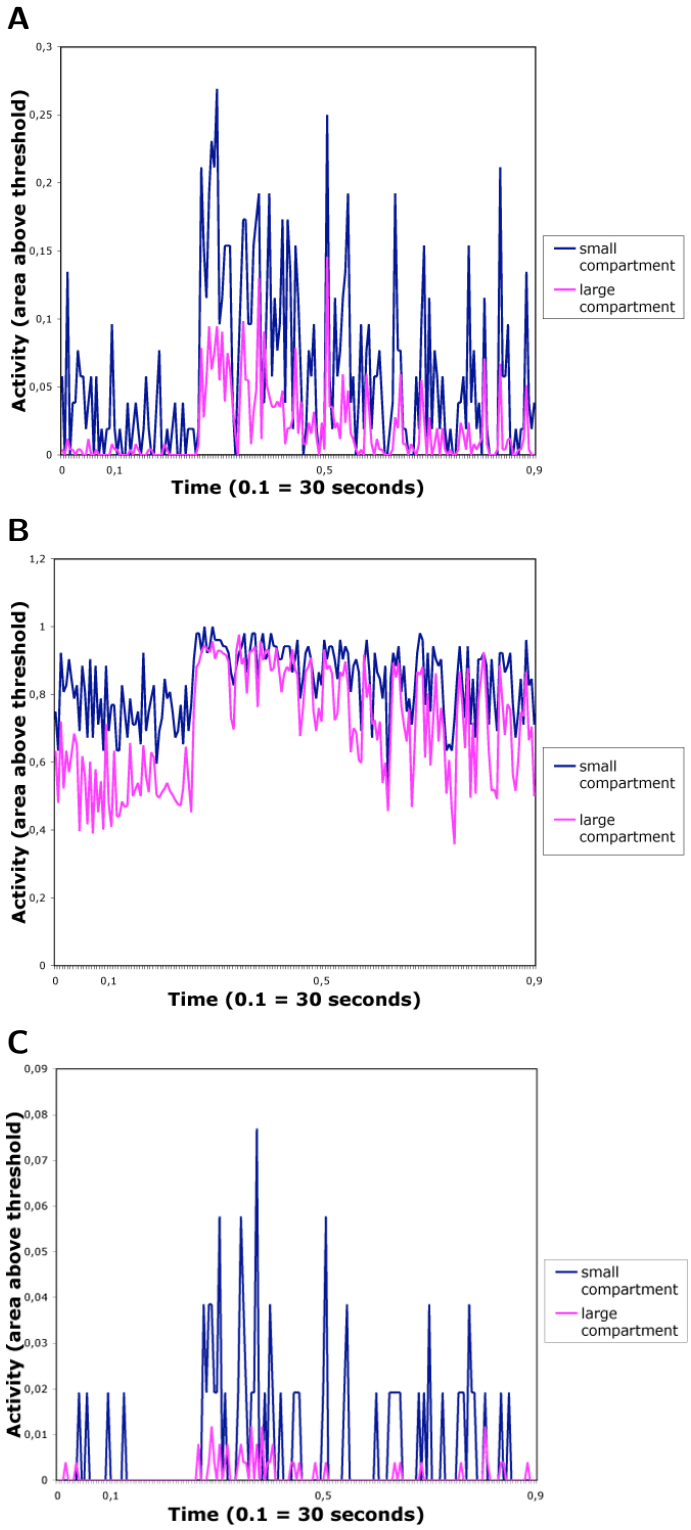


Figure 5.22: *Experimentally measured nuclear activity: With increasing threshold (from (A) to (C)) small compartments show higher activity than large ones.*

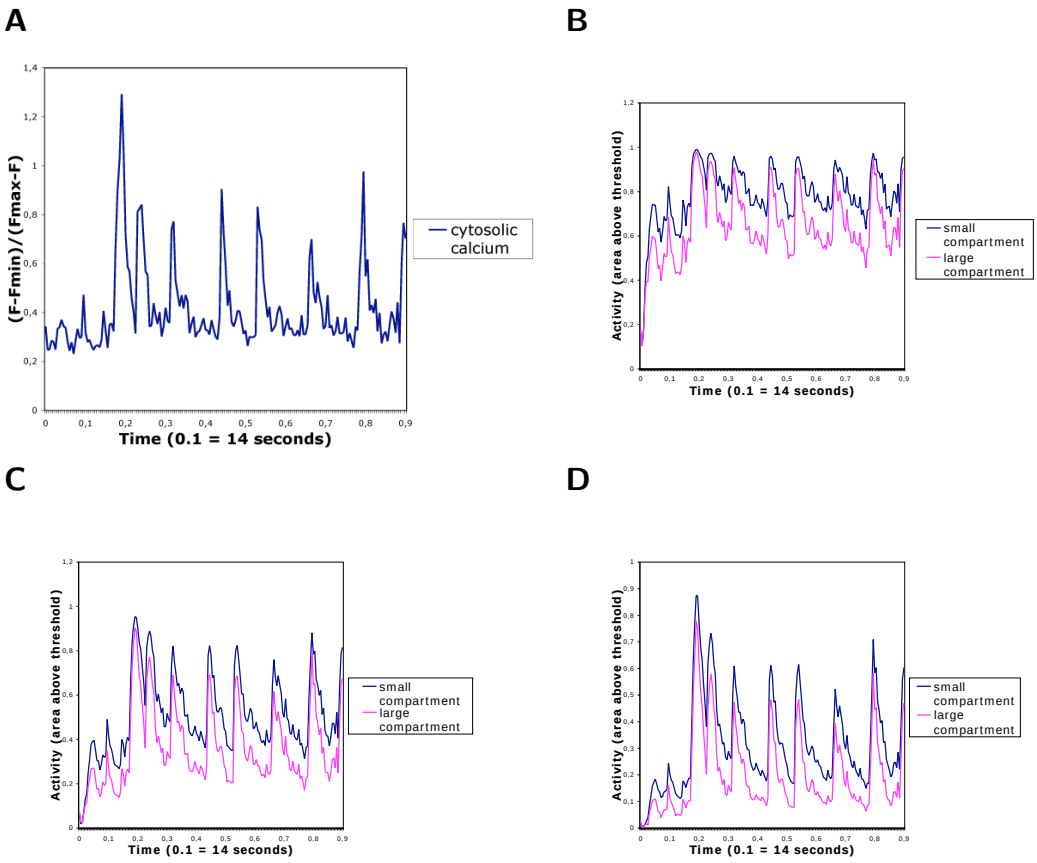


Figure 5.23: *Simulation of nuclear activity: (A) Time course of cytosolic calcium transient. (B) - (C) Activity of small and large compartment with increasing threshold.*

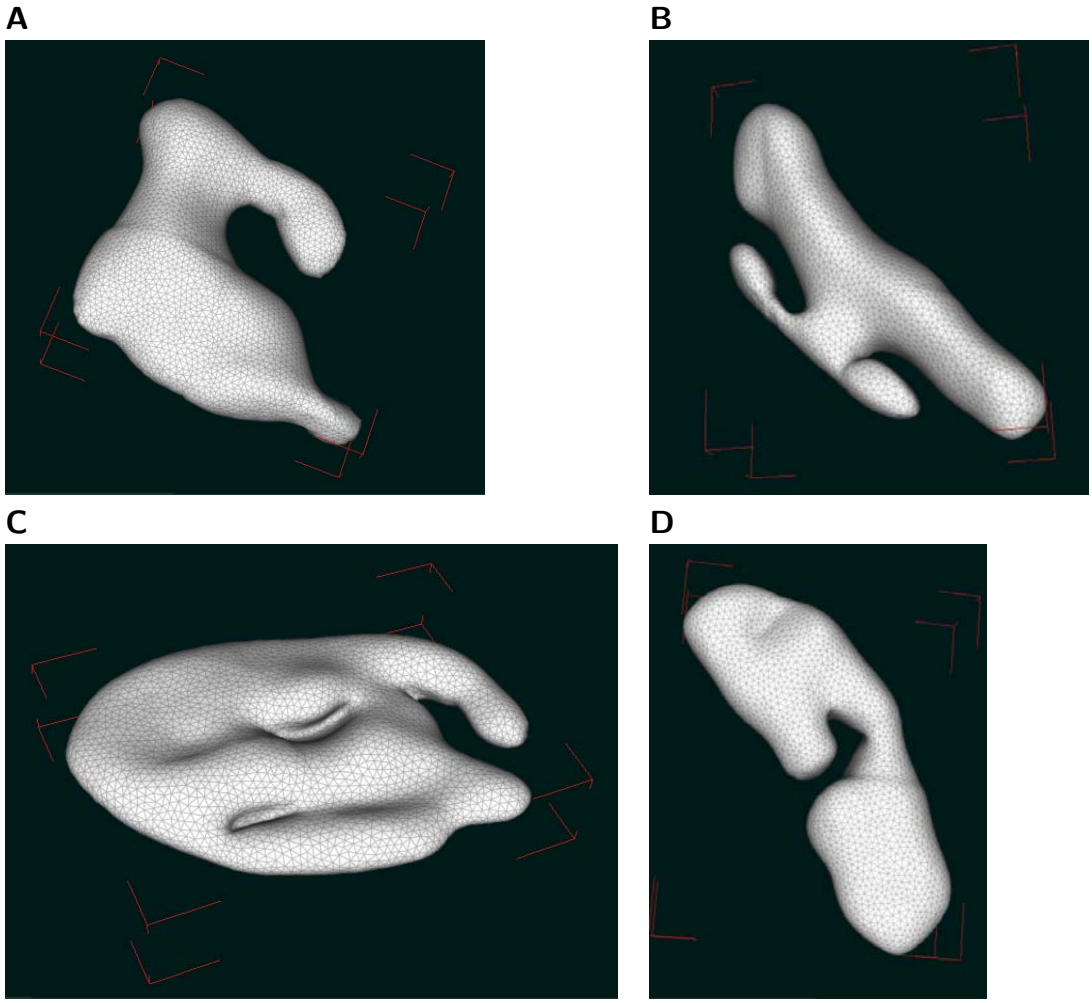


Figure 5.24: *Examples of the diversity of infolding structures.*

ried out at 5 Hz input signals. P. Bengston (IZN) stimulated cells electrically and recorded nuclear reaction with a CCD-camera. Simulations were also done at 5 Hz stimulation, Figure 5.27. In simulations as well as experiments, smaller micro-domains are more adept at resolving high frequencies (Figure 5.27 (A) and (B)). In a power spectral density plot (C), both model and experiment show a distinct amplitude at 5 Hz and the small nuclear compartment produces a higher peak at 5 Hz. This might play an important role for the cell, when relaying information to biochemical processes involved in gene transcription.
Simulations and experiments finally show, that the form of a nucleus from hippocampal neurons influences the dynamics of calcium signals. The amplitude of signals, the frequency and activity level depends on the way the nuclear envelope is folded. Deep infoldings form structural and functional micro-domains, and enable the nucleus to interpret cytosolic calcium signals differently, depending on the location.
Spherical nuclei, or large compartments of infolded nuclei, are more able to integrate incoming signals, while small compartments are tuned to resolve small changes in calcium concentration and fast oscillating signals.

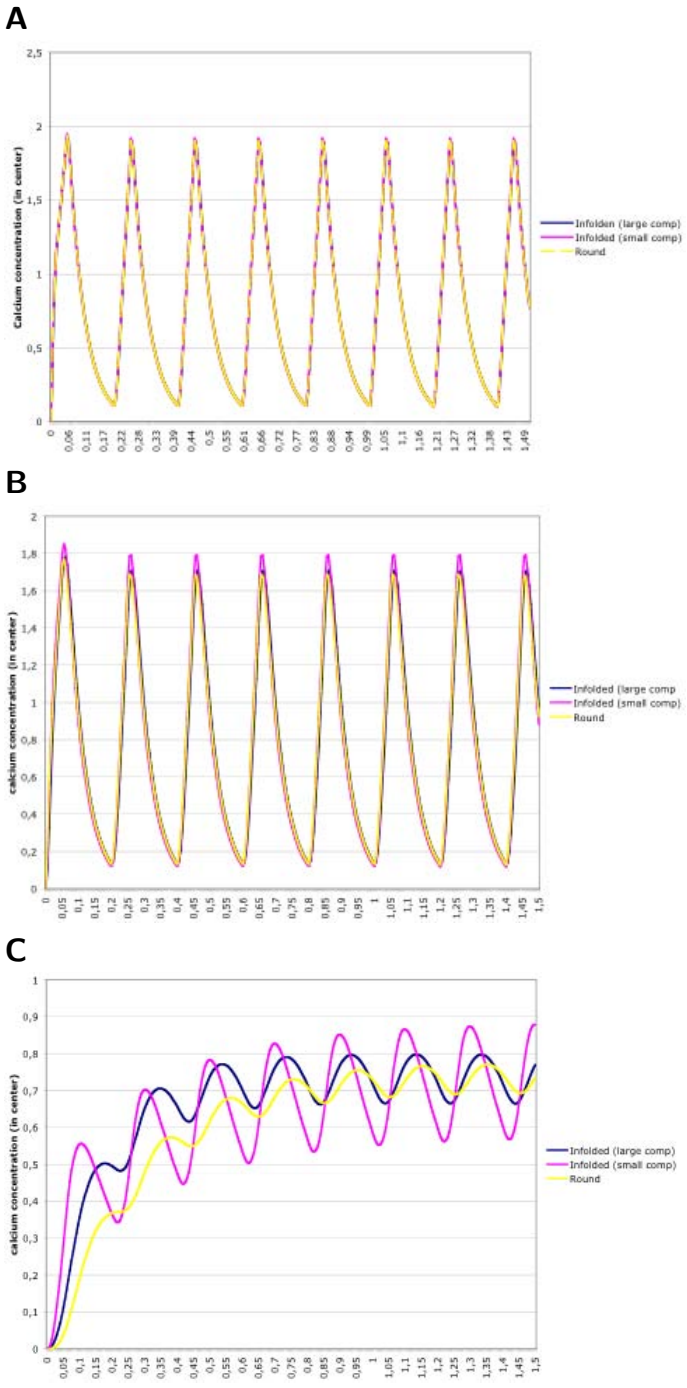


Figure 5.25: *High frequencies are better resolved by infolded nuclei, especially in small compartments of infolded nuclei: (A) 0.1 Hz frequency. (B) 0.2 Hz frequency. (C) 0.5 Hz frequency.*

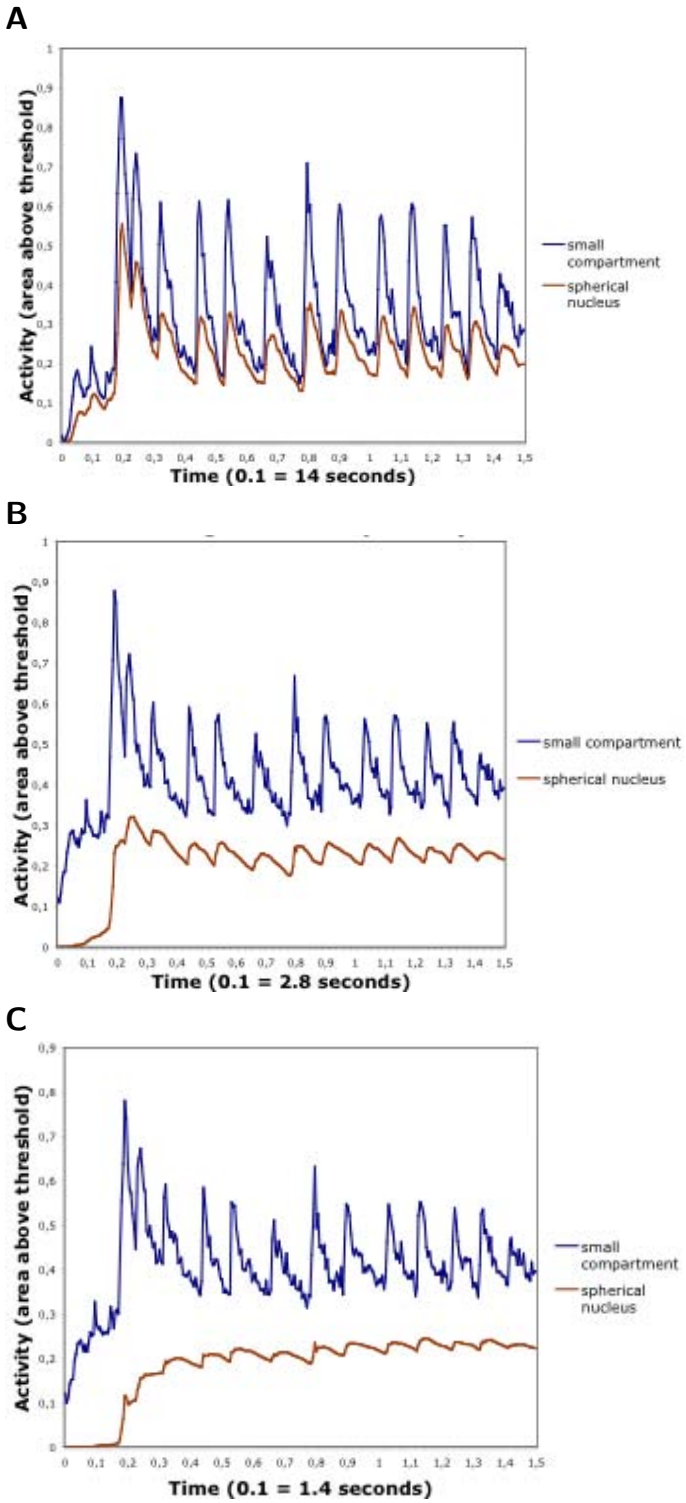


Figure 5.26: *Starting at 1 Hz, spherical nuclei begin to go into a plateau state, incapable of resolving high frequencies: (A) 0.1 Hz (B) 0.5 Hz (C) 1.0 Hz*

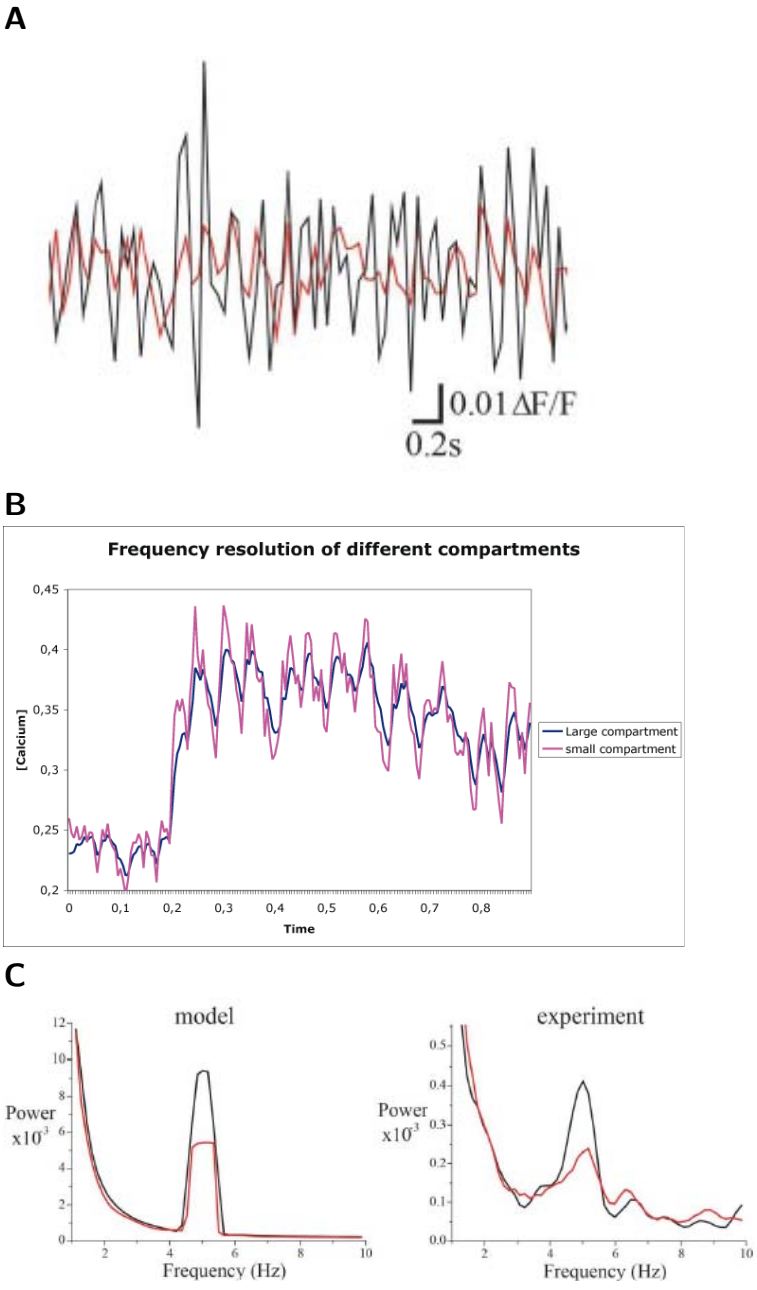


Figure 5.27: *Small nuclear micro-domains are tuned to resolve fast oscillating signals: (A) Experimental data of calcium-load time course after stimulation of the cell at 5 Hz. (B) Simulation of 5 Hz stimulation of the cell. (C) Power spectral density plot, which shows a defined peak at 5 Hz in both model and experiment. Small compartments express a higher amplitude.*

## 5.5 Estimating the Diffusion Coefficient of Nuclear Calcium

Until now, literature does not provide expermental data for the diffusion coefficient of calcium-ions in hippocampal neurons except for [1]. These experiments were carried out under laboratory conditions. Diffusion coefficients were measured in a cytosolic extract from *Xenopus laevis oocytes*, [1], which serves as an experimental model for neuron nuclei. As mentioned in Chapter 3 the interval for buffered calcium ions measured in [1] was stated as $D_{calcium} \in \{13 - 65\mu m^2/s\}$.
In order to validate the data of diffusion coefficients under those laboratory conditions, laser-assisted uncaging experiments were carried out in hippocampal neurons (Chapter 4).
Together with a parameter estimation algorithm developed for calculating the diffusion coefficient of nuclear calcium based on experimental data, see Chapter 4, the data stated in literature could be validated for nuclei from hippocampal neurons.
The program was implemented in the simulation environment *UG*, tested on artificial data and finally applied to experimental data.

### 5.5.1 Testing the Parameter Estimation Program

Estimating the diffusion coefficient for nuclear calcium could be reduced to minimizing the objective function

$$F(D) := \sum_{i=1}^{N} \sum_{j=1}^{M} (u_{ij}(D) - \hat{c}_{ij})^2 \tag{5.5}$$

where $\hat{c}_{ij}$ denotes the experimental data, $u_{ij}(D)$ the simulation data for a specific coefficient $D$, at location/time-points $ij$. The proposed algorithm for solving this minimization problem is stated in Chapter 4. In order to test whether and how fast the solution converges, tests on artificial data were carried out.
In order to generate "experimental data", the diffusion coefficient was set to the dimensionless value 100.0 and the calcium concentration at different points recorded when simulating nuclear diffusion. This data was then used as $\hat{c}_{ij}$ in the parameter estimation program. The program was then executed with varying initial coefficient values $D^{(0)}$, Figure 5.28.
The results show quadratic convergence to the expected value, independent of the initial value $D^{(0)}$. The program can be carried out until a defined number of iterations were carried out, or until the change in the coefficient value $\Delta D$ is smaller than a value $\varepsilon$.

### 5.5.2 Adding Experimental Data

Experimental data was recorded according to Chapter 4 and Figure 5.29. Since the uncaging location and measurement points are all located inside the nucleus,

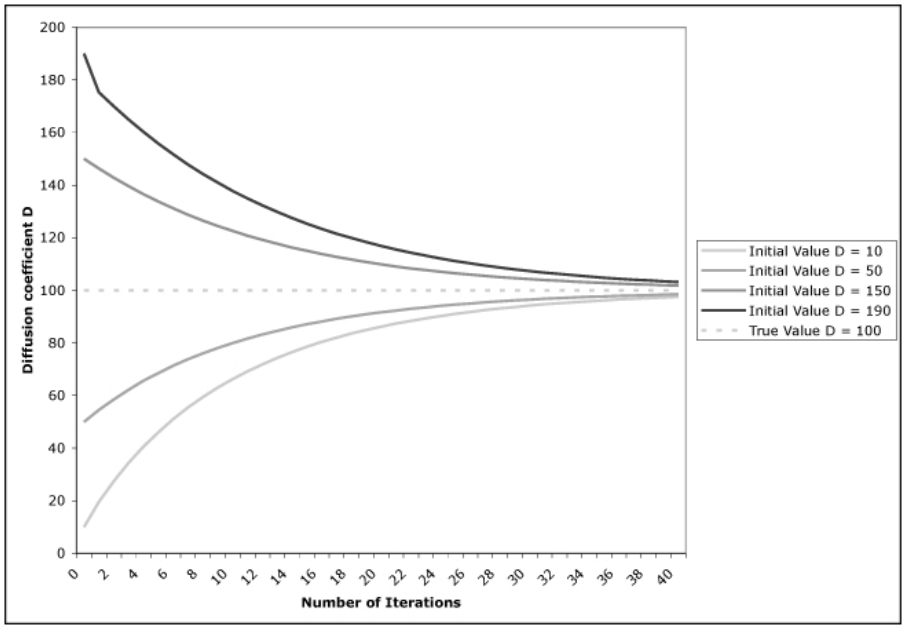


Figure 5.28: *Convergence tests of parameter estimation algorithm*

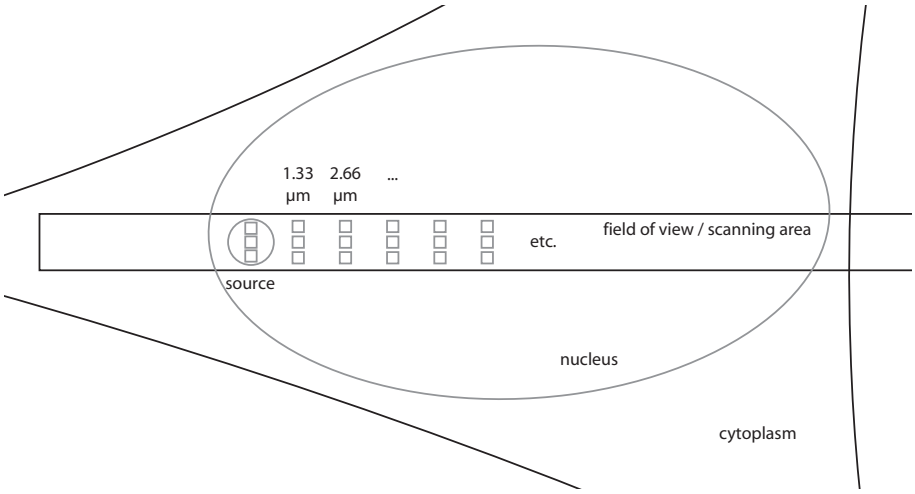


Figure 5.29: *Experimental set-up for retrieving data for parameter estimation: The uncaging source is recorded as well as points with varying distances from the source. Each point is recorded three times in order to retrieve a mean value.*

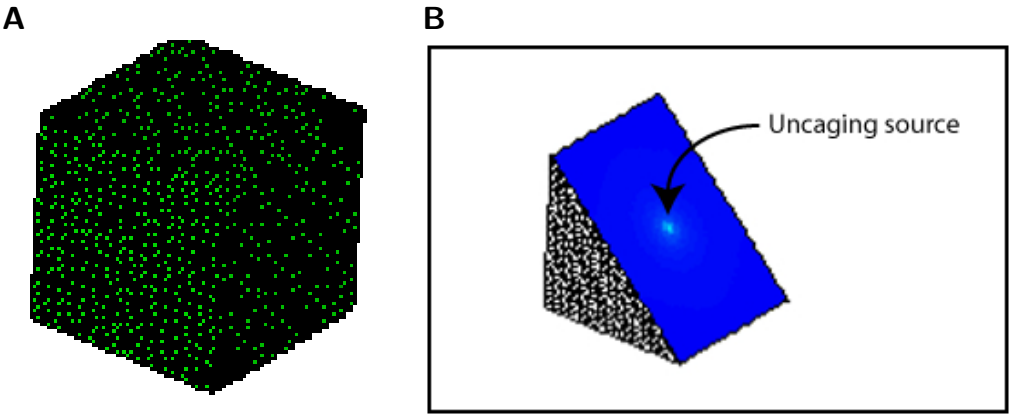


Figure 5.30: *(A) Cube in which the parameter estimation problem is defined. (B) The uncaging source is located at the center of the cube.*

the actual morphology of the nucleus plays no role. It is sufficient to carry out simulations on an abstract geometry. The mathematical setting is the following:

**Mathematical model:** Defined on a cube, discretized with a triangular surface grid and a tetrahedral volume grid, is the diffusion equation with a source term in the center of the cube, representing the uncaging location of calcium ions, Figure 5.30:

$$\frac{\partial u(x,t)}{\partial t} = div(D \cdot \nabla u(x,t)) \qquad \text{on } \Omega \subset \mathbb{R}^{3\times 1} \tag{5.6}$$

$$u(x,t_0) = u_0(x) \tag{5.7}$$

$$u(x,t) = f(t) \qquad \text{on } \partial\Omega \subset \mathbb{R}^3 \tag{5.8}$$

$$u(x_{Center}) = S(t) \tag{5.9}$$

with $S(t)$ denoting the experimentally measured uncaging source data and the first three equations the model derived in Chapter 3. Designated grid nodes that have the same distance to the uncaging site as the experimental measurement sites are calculated and the time course of calcium concentration in these points forwarded to the parameter estimation algorithm.

Results of the data-driven parameter estimation are shown in Figure 5.31. For the experiments and simulations which were carried out, the diffusion coefficient for nuclear calcium lies in the previously measured interval for buffered calcium in [1] at approximately $36\mu m^2/s$.

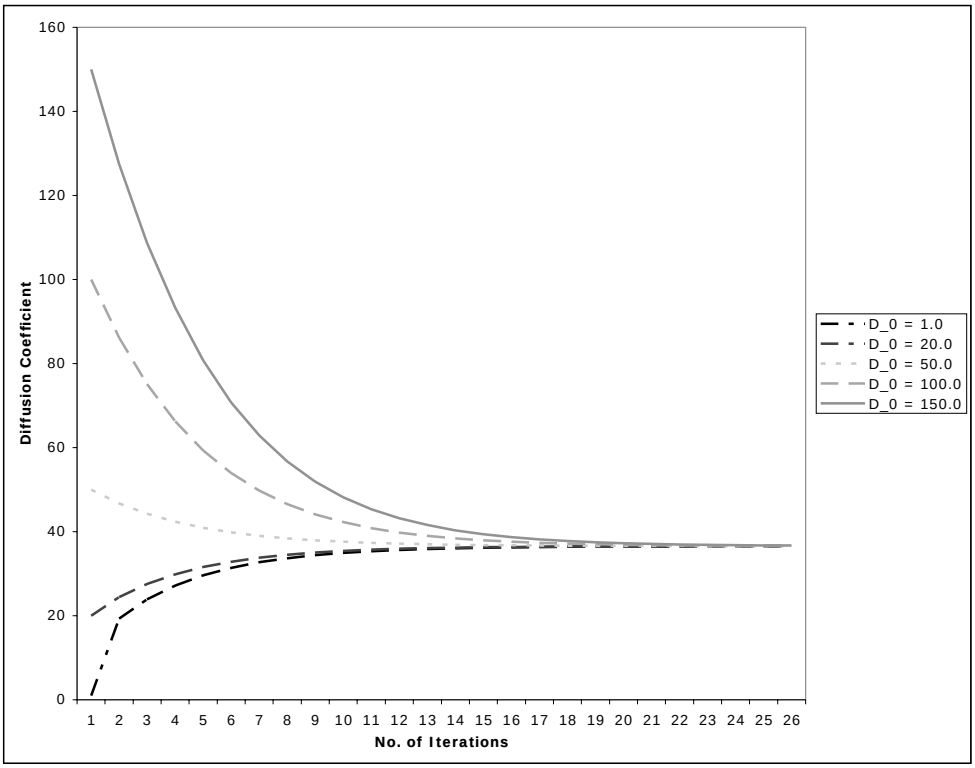


Figure 5.31: *Data-driven parameter estimation of the nuclear calcium diffusion coefficient with different starting points: Independent of the starting point the solution converges towards the value* $D \approx 36\mu m^2/s$.

# Chapter 6

# Conclusion

## 6.1 The Role of Calcium Signals

Electrical signals generated by synaptic activity propagate to the cell body, where calcium influx through NMDA receptors triggers vital biochemical processes. Phosphorylation of CREB in the cell's nucleus is one reaction triggered by a calcium signal diffusing into the nucleus and results in the expression of certain genes.
Work done by M. Wittmann (IZN) showed, that the origin of a calcium wave is vital for the cells fate, [67]. Where extrasynaptic NMDA-gated calcium influx led to cell death, synaptic NMDA-gated calcium propagates cell survival. Furthermore, during these studies, novel morphological features of the nucleus were discovered, [68]. Validated by electron microscopy (EM) studies by A. Hellwig (IZN), confocal fluorescence microscopy revealed fine structures traversing the nucleus. EM-data proved these structures to be the infolding of both the inner and outer membrane of the nuclear envelope, thus creating higher connectivity between the cytoplasm and the nucleus.
The cell nucleus contains the DNA and is the location where protein synthesis in the cell takes place. Biochemical processes that are responsible for learning, memory and survival are regulated by calcium signals, [6, 33, 5]. The observed structures in the nucleus cause a change in the connectivity between cytosol and the nucleus, which affects the central signaling pathway between synapse and nucleus.

## 6.2 The Inertia-Based Filter

To investigate the morphological novelties, a method to reconstruct the nuclear geometry from raw microscopy data was developed. Data was recorded by M. Wittmann (IZN) and A. Eder (IZN) by staining the nuclear lamina with a fluorescence indicator. Since raw data contains noise and discontinuities in the structure, a surface generating algorithm would not succeed in constructing a continuous isosurface of the nuclear envelope. An adequate image filter should process raw data in such a way that it dampens background noise, enhances the signal to noise ratio, and closes gaps in the membrane. State of the art nonlinear anisotropic diffusion filters, [54, 66], produced the wanted directional diffusivity for filtering two-dimensional substructures, yet this takes place at the cost of losing nuclear features.
In this project, a novel inertia-based, nonlinear, anisotropic diffusion filter was designed to filter two-dimensional substructures with minimal effect on important nuclear features. Directional diffusivity is regulated by calculating eigenvalues of the inertia tensor, [56]. Fine structures of the nucleus can therefore be preserved, while background noise is reduced and the membrane structure of the nucleus is made continuous. When setting parameters of the Perona-Malik and Weickert filter, [54, 66], to ideal values for keeping membrane diameters constant and closing gaps in the membrane, tests showed, that the inertia-based filter, developed for reconstructing nuclei, performed nearly three times better when comparing minimum gray values in membrane gaps.

Combined with the segmentation algorithm by Otsu, [52], the inertia-based filter produces image stacks that can be used for three-dimensional reconstructions. In order to create a surface triangulation of the nuclear envelope, a software toolbox developed by A. Heusel (SiT) and S. Reiter (SiT) was used. Three-dimensional reconstructions (see Appendix for an excerpt of reconstruction images) of hippocampal neuron nuclei revealed, that the fine structures in the nucleus observed in confocal fluorescence microscopy and EM, are in fact infoldings into the nuclei. By generating a 3D-image data base (at the moment consisting of 102 nuclei), it can be stated, that the morphology is a nuclear fingerprint.

## 6.3 Functionality of the Nuclear Morphology

In an attempt to investigate the function of these morphological novelties, an interdisciplinary project of the department Simulation in Technology (SiT) of the Interdisciplinary Center for Scientific Computing (IWR) and the Interdisciplinary Center for Neurosciences (IZN) was brought to life. Detailed mathematical modeling and experimental work was to investigate calcium signaling in the nucleus in respect to its geometry. A data-driven model, based on the three-dimensionally reconstructed nuclei and experimental data, was developed and implemented into the simulation environment *UG*, [8]. Surface grids for mathematical modeling were retrieved with the developed method described above and presented in [56]. Volume grids, needed for multi-grid solving methods for partial differential equations used by *UG*, were constructed using the software *Ansys* ICEM CFD. In a first step of modeling calcium diffusion under morphological consideration, two dimensional models of spherical and infolded nuclei were designed for systematic investigations. This preliminary data indicated, that due to reduced diffusion distances, nuclear infoldings have a visible effect on the dynamics of nuclear calcium signals.
Three-dimensional modeling on real nuclear morphologies was guided by two main aspects. The comparison between infolded and spherical nuclei and the comparison of structural micro-domains within single infolded nuclei. Reconstructions of nuclei showed that diffusion distances are minimized by nuclear infoldings and deep invaginations into the nucleus form structural sub-compartments in nuclei. The parameters investigated were the dynamics of the nuclear calcium load, nuclear activity and signal frequency. Calcium load and activity are strongly influenced by the form of a nucleus. Infolded nuclei show higher calcium signals than spherical ones, they are more active and more adept at resolving high signal frequencies and weak signals. Spherical nuclei carry out a more integrating task. They are less sensitive to small and fast fluctuations in the calcium code.
The morphology of a nucleus might be a way for the cell to regulate the signal that reaches transcriptional sites in the nucleus. Infoldings not only form structural compartmentalization, but also functional micro-domains, see Chapter 5. This observation made from mathematical models could also be shown in the laboratory,

[68, 67]. Experimental data from M. Wittmann (IZN) was able to prove, that in the active state 70% of cells contain infolded nuclei. Furthermore the count for infolded nuclei was higher in cells that are more active than others. This leads to the assumption, that the form of the nucleus changes under active conditions, making the nucleus more dynamic in its response to repeated and perhaps weak and/or fast signals.
The conclusion that can be made at this point, is that the nuclear morphology in most cases is not spherical, but contains deep invaginations in the nuclear envelope. The form of the nucleus influences the calcium code that is generated at synaptic NMDA sites and changes, depending on the state of activity of the cell. The fine structures of the nucleus are an optimization method for the cell, in respect to synapse to nucleus communication.

## 6.4 Properties of Nuclear Calcium

The diffusion model based on real three dimensional geometries includes a diffusion coefficient for nuclear calcium. This parameter has been investigated under laboratory conditions, [1]. While data presented in literature seem to pin-point the behavior of buffered calcium well, an attempt at validating this data, by using a data-driven inverse model was made. Implementing techniques from numerical optimization for parameter estimation in *UG*, together with experimental calcium uncaging experiments, [20], by A. Eder (IZN) completed a mathematical model with which the diffusion coefficient for nuclear calcium could be estimated. Results of the parameter estimation state, that nuclear calcium is in fact in buffered form, and has an active diffusion coefficient of approximately 36 $\mu m^2/s$.

The developed methods from mathematical image processing deliver novel reconstructions of nuclei from hippocampal neurons, which show stunning morphological features. Numerical modeling and mathematical optimization techniques could identify the nuclear geometry as a regulating parameter for the cell's communication between synapse and nucleus, as well as validate, from a modeling perspective, the diffusion properties of nuclear calcium.

# Appendix

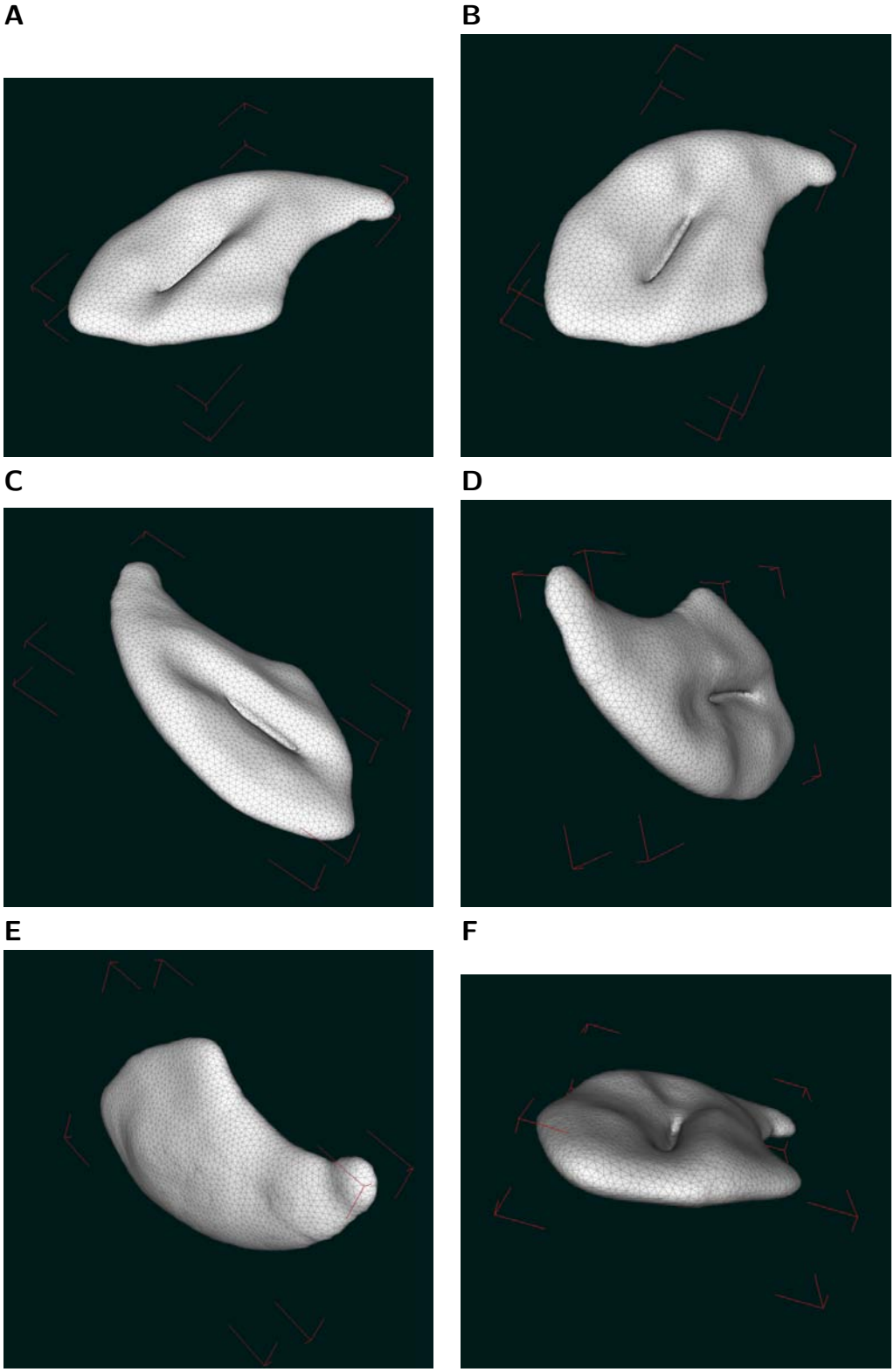


Figure 6.1: *Excerpt 1: Example of reconstructed nucleus*

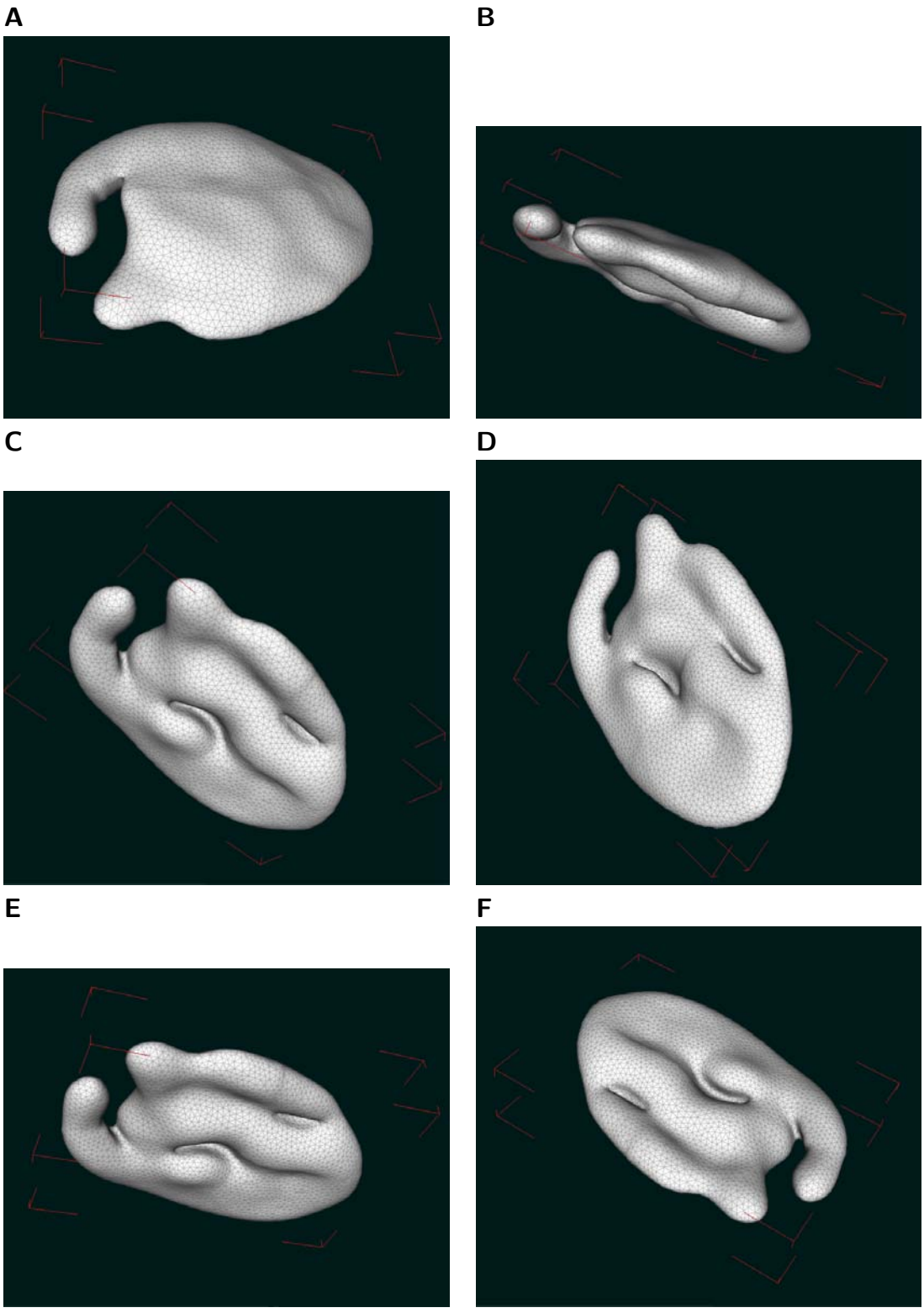


Figure 6.2: *Excerpt 2: Example of reconstructed nucleus*

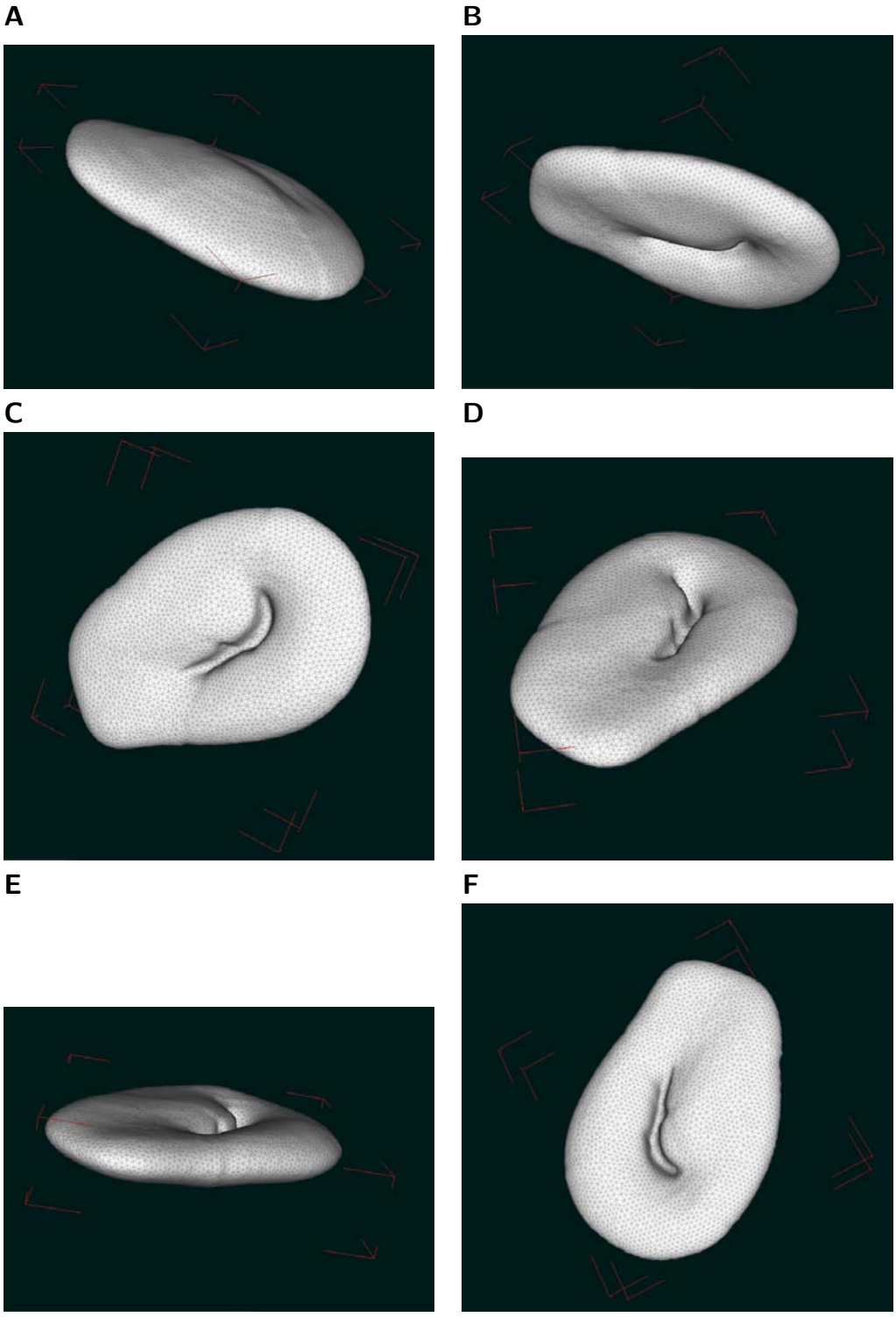


Figure 6.3: *Excerpt 3: Example of reconstructed nucleus*

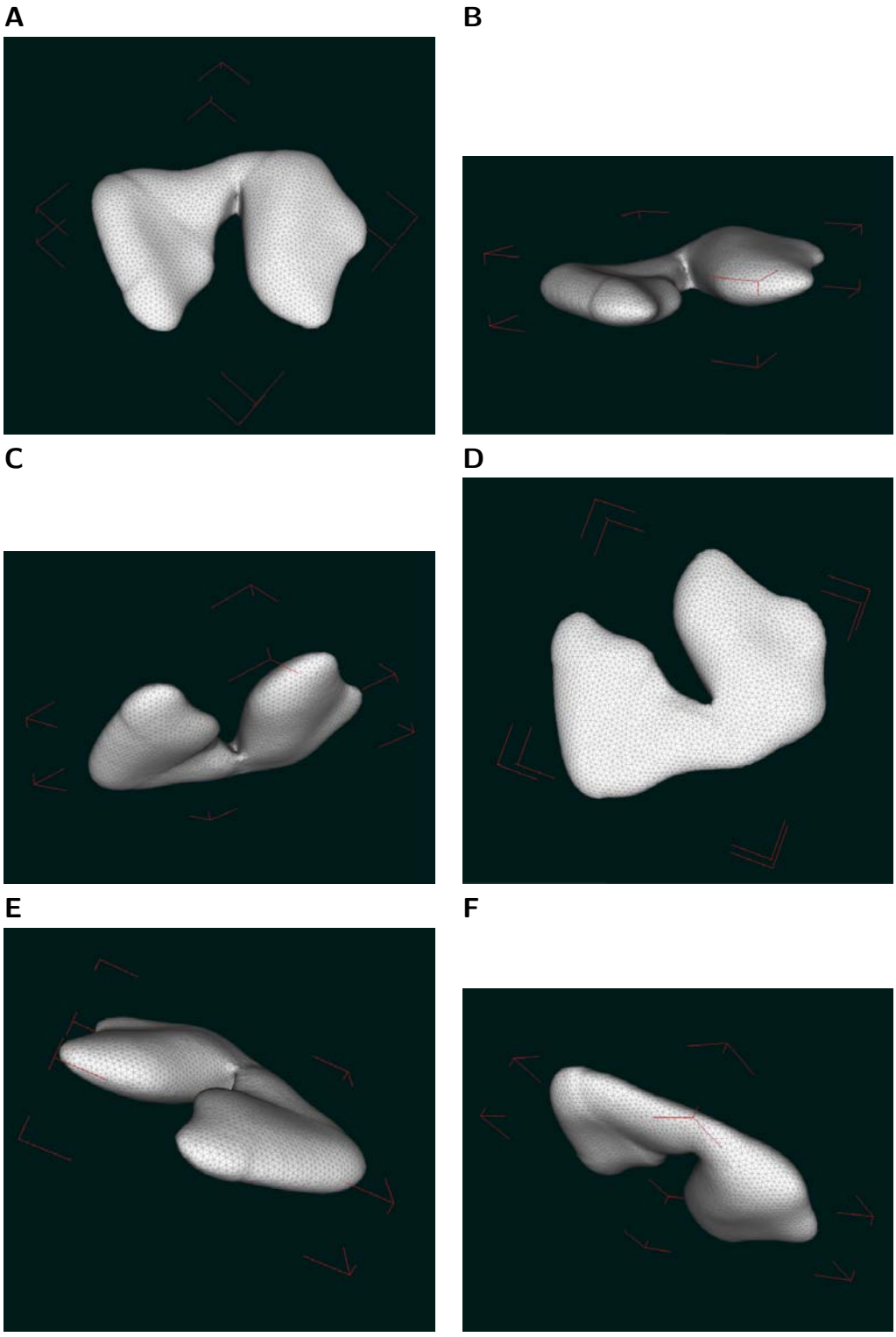


Figure 6.4: *Excerpt 4: Example of reconstructed nucleus*

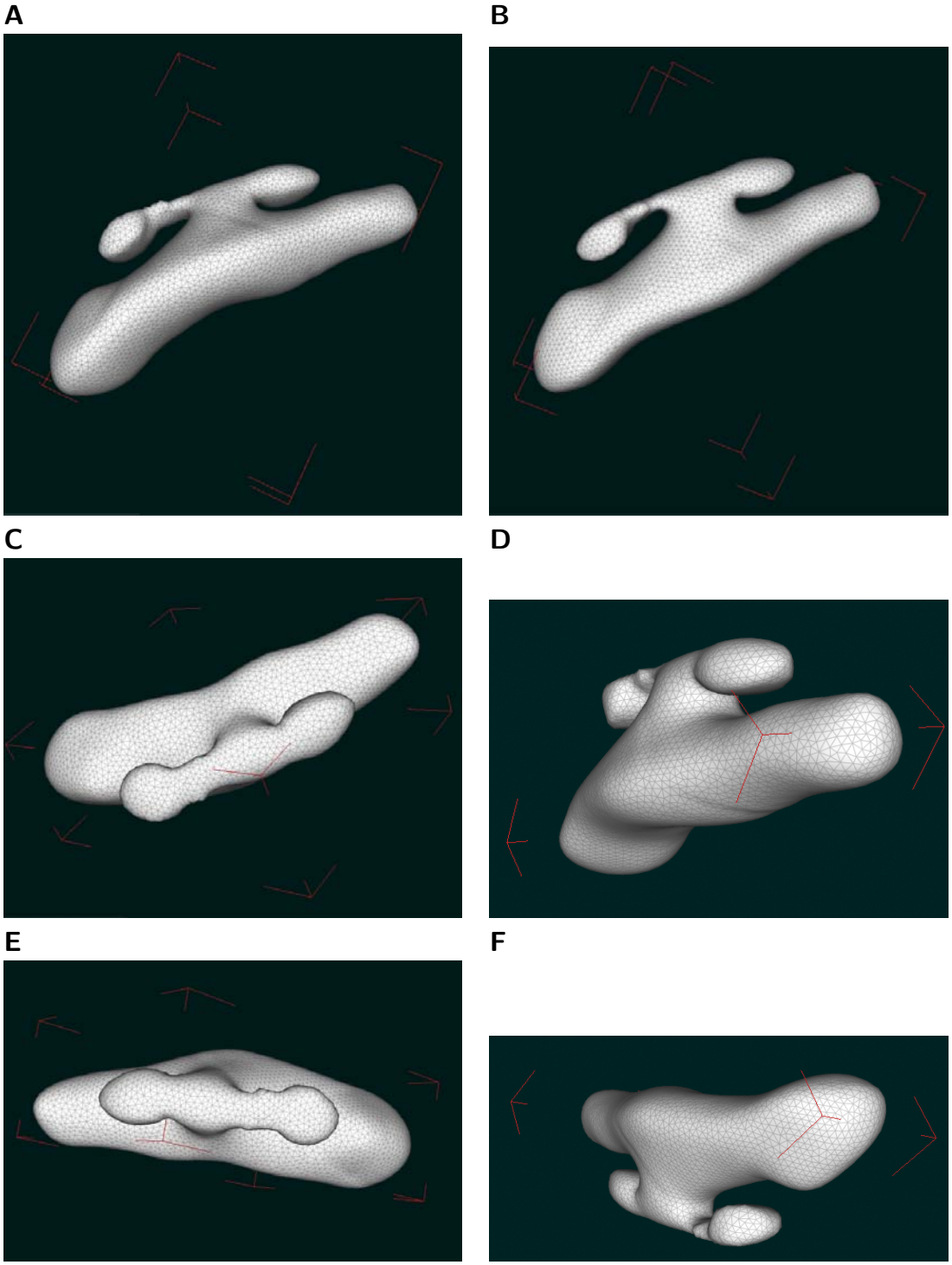


Figure 6.5: *Excerpt 5: Example of reconstructed nucleus*

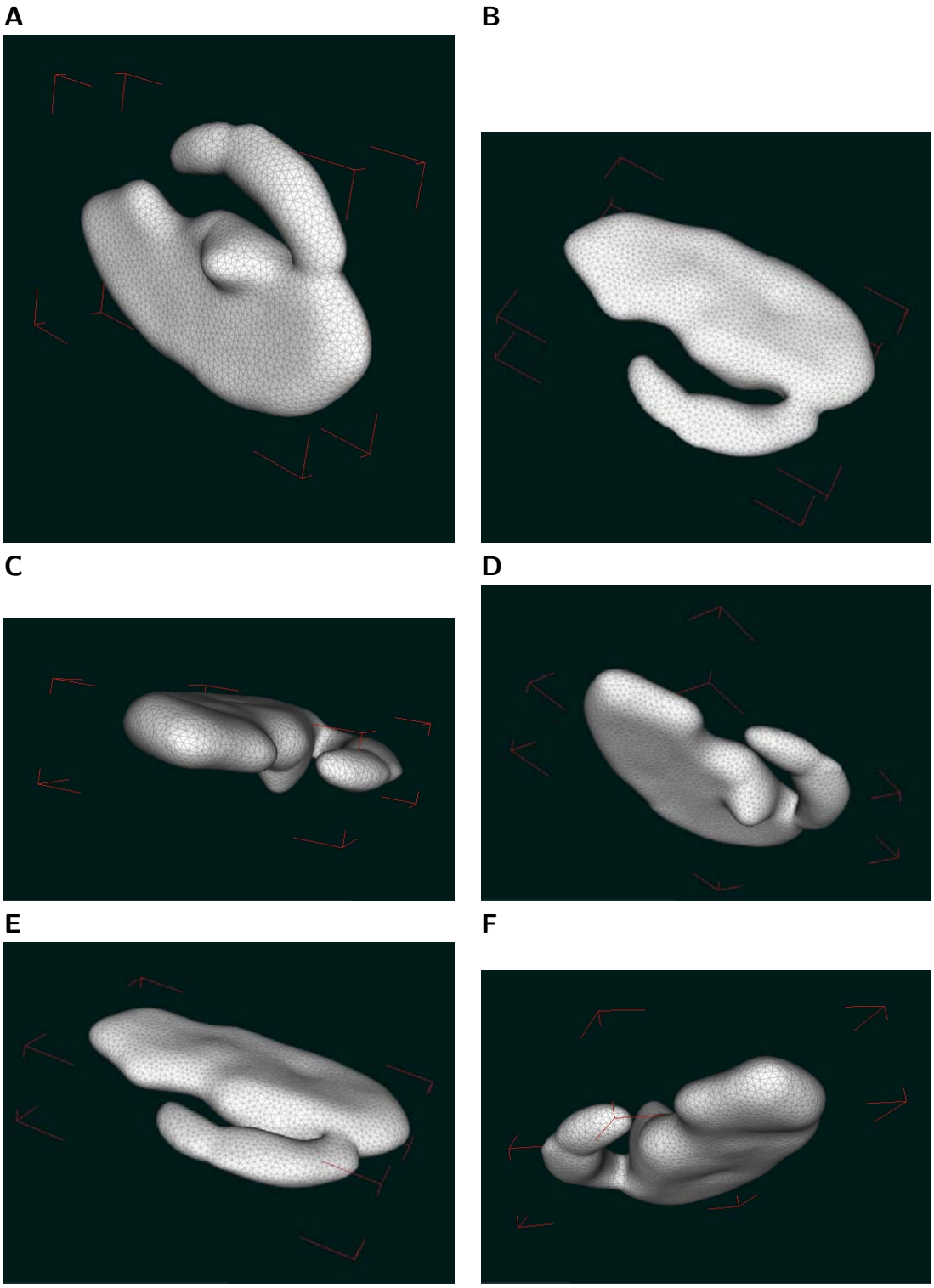


Figure 6.6: *Excerpt 6: Example of reconstructed nucleus*

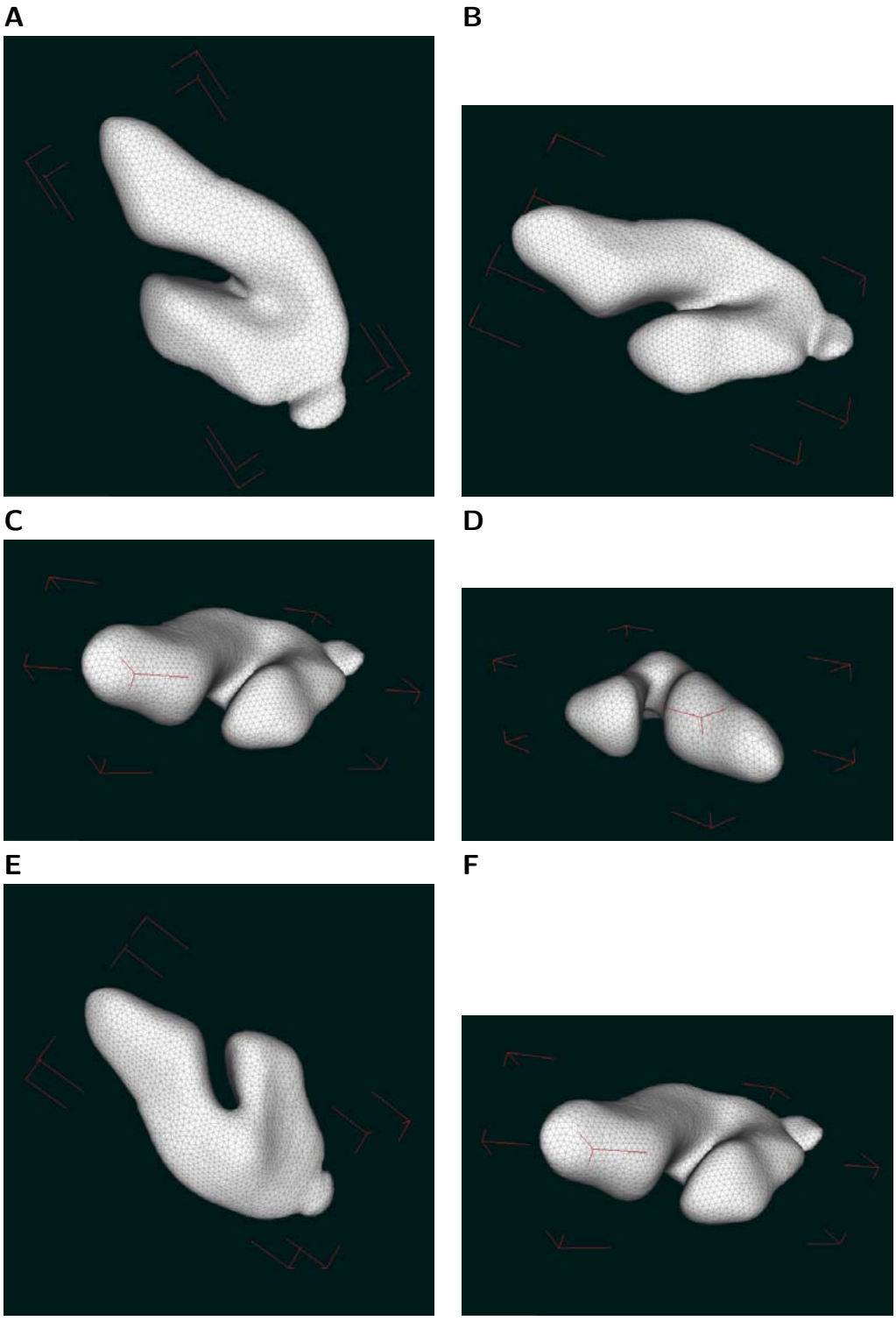


Figure 6.7: *Excerpt 7: Example of reconstructed nucleus*

# Bibliography


[1] N. L. Allbritton, T. Meyer, and L. Stryer. Range of Messenger Action of Calcium Ion and Inositol 1,4,5-Trisphosphate. *Science*, 258:1812 – 1815, 1992.

[2] http://www.ansys.com/products/icemcfd.asp.

[3] C. Aoki, C. Venkatesan, C. G. Go, J. A. Mong, and T. M. Dawson. Cellular and subcellular localization of NMDA-R1 subunit immunoreativity in the visual cortex of adult and neonatal rats. *J Neurosci*, 14:5202–5222, 1994.

[4] A. Bacci, C. Verderio, E. Pravettoni, and M. Matteoli. Synaptic and intrinsic mechanisms shape synchronous oscillations in hippocampal neurons in culture. *European Journal of Neuroscience*, 11:389–397, 1999.

[5] H. Bading. Transcription-dependent neuronal plasticity. *Eur. J. Biochem.*, 267:5280–5283, 2000.

[6] H. Bading, D.D. Ginty, and M.E. Greenberg. Regulation of gene expression in hippocampal neurons by distinct calcium signaling pathways. *Science*, 260:181–186, 1993.

[7] H. Bading and M. E. Greenberg. Stimulation of protein tyrosine phosphorylation by NMDA receptor activation. *Science*, 253:912–914, 1993.

[8] P. Bastian and G. Wittum. Adaptive Multigrid Methods: The UG Concept. In *Notes on Numerical Fluid Mechanics*. Vieweg, 1994.

[9] M.F. Bear and R. C. Malenka. Synaptic plasticity: LTP and LTD. *Curr Opin Neurobiol*, 4:389–399, 1994.

[10] P. J. Broser, R. Schulte, A. Roth, F. Helmchen, J. Waters, S. Lang, B. Sakmann, and G. Wittum. "Nonlinear anisotropic diffusion filtering of three-dimensional image data from 2-photon microscopy". *Journal of Biomedical Optics*, 9(6):1253–1264, 2004.

[11] V. Chan-Palay, S. L. Palay, and S. M. Billings-Gagliardi. Meynert cells in the primate visual cortex. *J Neurocytol*, 3:631–658, 1974.

[12] S. Chawla, G. E. Hardingham andD. R. Quinn, and H. Bading. CBP: a signal-regulated transcriptional coactivator controlled by nuclear calcium and CaM kinase IV. *Science*, 281:1505–1509, 1998.

[13] W. Cheung. Calmodulin plays a pivotal role in cellular regulation. *Science*, 207:19–27, 1980.

[14] D. Chin and A. R. Means. Calmodulin: A prototypical calcium sensor. *Trends Cell Biol.*, 10:322–328, 2000.

[15] Thomas M. Devlin. *Textbook of Biochemistry*. Wyley-Liss, 6th edition, 2006.

[16] E. W. Dijkstra. "A note on two problems in connexion with graphs". *Numerische Mathematik, vol.1. Mathematical Centre, Amsterdam, The Netherlands*, pages 269–271, 1959.

[17] R. E. Dolmetsch, R. S. Lewis, C. C. Goodnow, and J. I. Healy. Differential activation of trancription factors induced by $Ca^{2+}$ response amplitude and duration. *Nature*, 386:855–858, 1997.

[18] R. E. Dolmetsch, U. Pajvani, K. Fife, J. M. Spotts, and M. E. Greenberg. Signaling to the Nucleus by an L-type Calcium Channel-Calmodulin Complex Through the MAP Kinase Pathway. *Science*, 294:333–339, 2001.

[19] Ricardo E. Dolmetsch, Keli Xu, and Richard S. Lewis. Calcium oscillations increase the efficiency and specificity of gene expression. *Nature*, 392:933–936, 1998.

[20] A. Eder and H. Bading. Calcium signals can freely cross the nuclear envelope in hippocampal neurons: somatic calcium increases generate nuclear calcium transients. *BMC Neuroscience*, 8:57, 2007.

[21] L. C. Evans. *Partial Differential Equations*. American Mathematical Society, 2002.

[22] D. W. Fawcett. On the occurrence of a fibrous lamina on the inner aspect of the nuclear envelope in certain cells of vertebrates. *Am J Anat*, 119:129–145, 1966.

[23] M. Fricker, M. Hollinshead, N. White, and D. Vaux. Interphase nuclei of many mammalian cell types contain deep, dynamic, tubular membrane-bound invaginations of the nuclear envelope. *J Cell Biol*, 136:531–544, 1997.

[24] R. D. Goldman, Y. Gruenbaum, R. D. Moir, D. K. Shumaker, and T. P. Spann. Nuclear lamins: building blocks of nuclear architecture. *Genes & Dev.*, 16:533–547, 2002.

[25] D. Görlich and U. Kutay. Transport between the cell nucleus and the cytoplasm. *Annu Rev Cell Dev Biol.*, 15:607–660, 1999.

[26] G. Grynkiewicz, M. Poenie, and R. Y. Tsien. A New Generation of $Ca^{2+}$ Indicators with Greatly Improved Fluorescence Properties. *The Journal of Biological Chemistry*, 260:3440–3450, 1985.

[27] Arthur C. Guyton and John E. Hall. *Textbook of Medical Physiology.* W.B. Saunders Company, 9th edition, 1996.

[28] W. Hackbusch. *Multi-Grid Methods and Applications.* Springer-Verlag Berlin Heidelberg New York Tokyo, 1985.

[29] W. Hackbusch. *Theorie und Numerik elliptischer Differentialgleichungen.* Teubner Studienbücher: Mathematik, 1986.

[30] W. Hackbusch. *Iterative Lösung großer schwachbesetzter Gleichungssysteme.* Teubner Studienbücher: Mathematik, second edition, 1993.

[31] G. E. Hardingham, F. J. L. Arnold, and Hilmar Bading. Nuclear calcium signaling controls CREB-mediated gene expression triggered by synaptic activity. *Nature Neuroscience*, 4:261–267, 2001.

[32] G. E. Hardingham and Hilmar Bading. The Yin and Yang of NMDA receptor signalling. *TRENDS in Neuroscience*, 26:81–89, 2003.

[33] G. E. Hardingham, S. Chawla, C. M Johnson, and H. Bading. Distinct functions of nuclear and cytoplasmic calcium in the control of gene expression. *Nature*, 385:260–265, 1997.

[34] Giles E. Hardingham, Yuko Pukunaga, and Hilmar Bading. Extrasynaptic NMDARs oppose synaptic NMDARs by triggering Creb shut-off and cell death pathways. *Nature Neuroscience*, 5:405–414, 2002.

[35] S. B. Hazra, H. Class, R. Helmig, and V. Schulz. Forward and inverse problems in modeling of multiphase flow and transport through porous media. *Journal of Computational Physics*, 8:21–47, 2004.

[36] S. B. Hazra and V. Schulz. Numerical parameter identification in multiphase flow through porous media. *Comput Visual Sci*, 5:107–113, 2002.

[37] M. Hochstrasser and J. W. Sedat. Three-dimensional organization of Drosophila melanogaster interphase nuclei. II. Chromosome spatial organization and gene regulation. *J Cell Biol*, 104:1471–1483, 1987.

[38] http://rsb.info.nih.gov/ij/, May, 2007.

[39] J. Klingauf and E. Neher. Modeling Buffered $Ca^{2+}$ Diffusion Near the Membrane: Implications for Secretion in Neuroendocrine Cells. *Biophysical Journal*, 72:674–690, 1997.

[40] G. Kluge and G. Neugebauer. *Grundlagen der Thermodynamik.* Spektrum Lehrbuch, 1994.

[41] P. De Koninck and H. Schulman. Sensitivity of CaM Kinase II to the Frequency of $Ca^{2+}$ Oscillations. *Science*, 279:227–230, 1998.

[42] G. D. C. Kuiken. *Thermodynamics of irreversible processes.* Whiley, 1994.

[43] S. Kupzig, S. A. Walker, and P. J. Cullen. The frequencies of calcium oscillations are optimized for efficient calcium-mediated activation of Ras and the ERK/MAPK cascade. *PNAS*, 102:7577–7582, 2005.

[44] F. Lenzen. 3D-Rekonstruktion von DNA-Strukturen. Master's thesis, University at Bonn, 2001.

[45] J. M. McHugh and J. L. Kenyon. An Excel-based model of $Ca^{2+}$ diffusion and fura 2 measurements in a spherical cell. *Am J Physiol Cell Physiol*, 286:C342–C348, 2004.

[46] P. G. Mermelstein, K. Deisseroth, N. Dasgupta, A. L. Isaksen, and R. W. Tsien. Calmodulin priming: Nuclear translocation of a calmodulin complex and the memory of prior neuronal activity. *PNAS*, 98:15342–15347, 2001.

[47] M. Micheletti, A. Brioschi, R. Fesce, and F. Grohovaz. A novel pattern of fast calcium oscillations point to calcium and electrical activity cross-talk in rat chromaffin cells. *Cell Mol. Life Sci.*, 62:95–104, 2005.

[48] A. Minta, J. P. Y. Kao, and R. Y. Tsien. Fluorescent Indicators for Cytosolic Calcium Based on Rhodamine Fluorescein Chromophores. *The Journal of Biological Chemistry*, 264:8171–8178, 1989.

[49] http://www.neura.org.

[50] J. Nocedal and S. J. Wright. *Numerical Optimization.* Springer Science+Business Media, second edition, 2006.

[51] D. M. O'Malley. Calcium Permeability of the Neuronal Nuclear Envelope: Evaluation Using Confocal Volumes and Intracellular Perfusion. *The Journal of Neuroscience*, 14:5741–5758, 1994.

[52] N. Otsu. "A threshold selection method from gray-level histograms". *IEEE Trans. Systems, Man, and Cybernetics*, 9(1), 1979.

[53] C. Perez-Terzic, L. Stehno-Bittel, and D. E. Clapham. Nucleoplasmic and cytoplasmic differences in the fluorescence properties of the calcium indicator Fluo-3. *Cell Calcium*, 21:275–282, 1997.

[54] P. Perona and J. Malik. Scale-Space and Edge Detection Using Anisotropic Diffusion. *IEEE Trans. PAMI*, 12:629–639, 1990.

[55] G. Queisser. Rekonstruktion und Vermessung der Geometrie von Neuronen-Zellkernen. Master's thesis, University of Heidelberg, 2005.

[56] G. Queisser, M. Wittmann, H. Bading, and G. Wittum. Filtering, reconstruction and measurement of the geometry of nuclei from hippocampal neurons based on confocal microscopy . *Journal for Biomedical Optics, in press*, 2007.

[57] F. Sala and A. Hernández Cruz. Calcium diffusion modeling in a spherical neuron. *Biophys. J.*, 57:313–324, 1990.

[58] L. Santella. The Cell Nucleus: An Eldorado to Future Calcium Research? *J. Membrane Biol.*, 153:83–92, 1996.

[59] F.A. Stephenson. Structure and trafficking of NMDA and $GABA_A$ receptors. *Biochem Soc Trans.*, 34:877–81, 2006.

[60] J. Stoer. *Numerische Mathematik 1.* Springer Berlin Heidelberg New York, 9th edition, 2005.

[61] J. Stoer and R. Bulirsch. *Numerische Mathematik 2.* Springer Berlin Heidelberg New York, 4th edition, 2000.

[62] G. M. Thomas and R. L. Huganir. MAPK Cascade Signalling and Synaptic Plasticity. *Nature Reviews Neuroscience*, 5:173–183, 2004.

[63] http://www.llnl.gov/visit/.

[64] S. S.-H. Wang. Modeling the Apparent Diffusion Constant of Calcium Ions Emanating From a Channel: Implications for Calcium Wave Propagation. *Biol. Bull.*, 185:297–298, 1993.

[65] J. Weickert. Theoretical Foundations of Anisotropic Diffusion in Image Processing. *Computing*, Suppl. 11:221–236, 1996.

[66] J. Weickert. *Anisotropic Diffusion in Image Processing.* B.G. Teubner Stuttgart, 1998.

[67] M. Wittmann. *Synaptic and extrasynaptic NMDA receptors in hippocampal neurons: regulation of nuclear shape and cell fate.* PhD thesis, University of Heidelberg, 2006.

[68] M. Wittmann, G. Queisser, Andrea Hellwig, G. Wittum, and H. Bading. NMDA receptor signaling controls the geometry of the cell nucleus. *Nature Neuroscience, submitted*, 2007.

[69] Z. Xia and D. R. Storm. The role of calmodulin as a signal integrator for synaptic plasticity. *Nat Rev Neurosci.*, 6:267–276, 2005.