

Institut für Parallelle und Verteilte Systeme

Universität Stuttgart  
Universitätsstraße 38  
D-70569 Stuttgart

Bachelorarbeit

**Untersuchung von  
Modellordnungsreduktions-  
Methoden (MOR) für die  
Ausbreitung von  
Aktionspotentialen entlang von  
Muskelfasern**

Pascal Litty

**Studiengang:** Informatik

**Prüfer/in:** Prof. Dr. rer. nat. habil. Miriam Mehl

**Betreuer/in:** Dr.-Ing. Nehzat Emamy

**Beginn am:** 22. Juni 2017

**Beendet am:** 22. Dezember 2017

**CR-Nummer:** F.2.1, G.1.2, G.1.3, G.1.7, G.1.8, G.3, G.4, J.3



## **Kurzfassung**

Model Order Reduction (MOR) ist der Oberbegriff für Methoden, die den Berechnungsaufwand mathematischer Modelle verringern, indem deren Komplexität reduziert wird. Können diese Methoden Berechnung einer Kombination von Hodgkin-Huxley-Modell und Monodomain-Modell beschleunigen? Das Hodgkin-Huxley-Modell ist ein rechenaufwändiges Modell zur Berechnung der Ionenströme innerhalb von Muskelfasern, welches mit dem Monodomain-Modell kombiniert wird, das die Ausbreitung von Aktionspotentialen entlang der Muskelfasern beschreibt.

Untersucht werden in der Bachelorarbeit Fehler und Beschleunigung von MOR-Verfahren wie POD-Galerkin oder Discrete Empirical Interpolation Method (DEIM) mithilfe von KerMor, einem Framework, das Methoden der Model Order Reduction bereitstellt.



# Inhaltsverzeichnis

<b>1 Einleitung</b>	<b>15</b>
<b>2 Grundlagen</b>	<b>17</b>
2.1 Anatomische, physikalische und biochemische Grundlagen . . . . .	17
2.2 Nichtlineare dynamische Differentialgleichungen . . . . .	19
2.3 Hodgkin-Huxley-Modell . . . . .	20
2.4 Monodomain-Modell . . . . .	23
2.5 Löser . . . . .	23
2.6 Model Order Reduction . . . . .	24
<b>3 Ergebnisse</b>	<b>27</b>
3.1 Kombination der Modelle und Diskretisierung . . . . .	27
3.2 KerMor . . . . .	28
3.3 Singulärwerte . . . . .	28
3.4 Einzelnes Aktionspotential . . . . .	30
3.5 Mehrere Trajectories . . . . .	34
3.6 Vergleich der verschiedenen Diskretisierungen . . . . .	35
<b>4 Zusammenfassung und Ausblick</b>	<b>41</b>
<b>Literaturverzeichnis</b>	<b>43</b>



# Abbildungsverzeichnis

2.1	Motoneuron . . . . .	18
2.2	Skelettmuskel . . . . .	18
2.3	Aktionspotential . . . . .	19
2.4	Sarkomere . . . . .	20
2.5	Hodgkin-Huxley-Modell: Stromkreis . . . . .	21
3.1	Singulärwerte für Matrix mit V, m, n und h-Werten (absolut) . . . . .	29
3.2	Singulärwerte für Matrix mit V, m, n und h-Werten (relativ) . . . . .	30
3.3	Singulärwerte für Matrix mit V-Werten (absolut) . . . . .	30
3.4	Singulärwerte für Matrix mit V-Werten (relativ) . . . . .	31
3.5	POD-Galerkin Beschleunigung über verschiedene Zeitschritte . . . . .	31
3.6	POD-Galerkin Fehler über verschiedene Zeitschritte . . . . .	32
3.7	POD-Galerkin Beschleunigung (absolut) . . . . .	32
3.8	POD-Galerkin Beschleunigung (relativ) . . . . .	33
3.9	POD-Galerkin Fehler (absolut) . . . . .	33
3.10	POD-Galerkin Fehler (relativ) . . . . .	34
3.11	POD-Galerkin Fehler-Beschleunigungsdiagramm . . . . .	34
3.12	DEIM Beschleunigung (relativ) . . . . .	35
3.13	DEIM Fehler (relativ) . . . . .	35
3.14	DEIM Fehler 320 Elemente (relativ) . . . . .	36
3.15	DEIM und POD-Galerkin, DEIM fest, Beschleunigung (relativ) . . . . .	36
3.16	DEIM und POD-Galerkin, DEIM fest, Fehler (relativ) . . . . .	37
3.17	DEIM und POD-Galerkin, POD fest, Fehler (relativ) . . . . .	37
3.18	DEIM und POD-Galerkin, POD fest, Fehler (relativ) . . . . .	38
3.19	POD-Galerkin mit mehreren Input-Funktionen Beschleunigung . . . . .	38
3.20	POD-Galerkin mit mehreren Input-Funktionen Fehler . . . . .	39
3.21	POD-Galerkin mit V-Matrix, Fehler . . . . .	39
3.22	POD-Galerkin mit V-Matrix, Fehler . . . . .	40



## **Tabellenverzeichnis**



## **Verzeichnis der Listings**



# **Verzeichnis der Algorithmen**

2.1	DEIM . . . . .	26
-----	----------------	----



# 1 Einleitung

Die Simulation der Ausbreitung von Aktionspotentialen entlang von Muskelfasern kann Teil der Simulation in ganzen Muskeln sein. Diese Berechnungen sind sehr aufwändig, da ein Muskel eine große Anzahl im Bereich von mehreren Millionen von Muskelfasern verschiedener Größe enthalten kann. Das Modell, das in dieser Arbeit untersucht wird, ist ein nichtlineares dynamisches System gewöhnlicher Differentialgleichungen, die mit einer partiellen Differentialgleichung gekoppelt werden. Dieses Modell muss also auch noch mit einer ausreichenden Anzahl diskreter Elemente und mit ausreichend feinen Zeitschritten berechnet werden.

Diese Berechnungen können von Model-Order-Reduction-Methoden wie POD-Galerkin und DEIM beschleunigt werden. Model-Order-Reduction-Methoden sind eine Klasse mathematischer Verfahren, die das Ziel haben, Rechenzeit und Komplexität von dynamischen Systemen, die viel Rechenzeit benötigen und sehr komplex sind, zu reduzieren. Diese Verfahren arbeiten unter der Prämisse, dass sich der Mechanismus, der dem zu reduzierenden System zugrunde liegt, in einer niedrigeren Dimension effektiv approximiert werden kann. Daher wird in dieser Arbeit untersucht, wie dieses Modell am besten für verschiedene MOR-Methoden approximiert wird.

## Gliederung

Die Arbeit ist in folgender Weise gegliedert:

**Kapitel 2 – Grundlagen:** Hier werden die Grundlagen beschrieben, die für die Simulationen in dieser Arbeit benötigt werden.

**Kapitel 3 – Ergebnisse:** In diesem Kapitel werden die Ergebnisse für die Berechnung des vorstellten Modells mit MOR-Methoden beschrieben.

**Kapitel 4 – Zusammenfassung und Ausblick** bespricht die Ergebnisse der Arbeit sowie mögliche zukünftige Schritte in diesem Bereich.



## 2 Grundlagen

### 2.1 Anatomische, physikalische und biochemische Grundlagen

In diesem Kapitel werden kurz Aufbau und Funktionsweise von Motoneuronen und Skelettmuskeln sowie chemische Reaktionen im Skelettmuskel beschrieben [Hei]. Besonders wird in diesem Kapitel darauf eingegangen, wie Aktionspotentiale über Motoneuron und Muskelfasern übertragen werden und schließlich zu einer Kraftentzerrung in den Muskelfasern führen. Für weitere Details wird auf Heidlauf [Hei] verwiesen. Diese Quelle wird auch zur Beschreibung von Funktionsweise und Aufbau in diesem Kapitel verwendet und aus ihr sind Abbildung 2.1, Abbildung 2.2 und Abbildung 2.3. Wenn in Kapitel 2.1 zusätzliche oder andere Quellen verwendet werden, dann werden diese am Ende des Satzes oder Absatzes genannt.

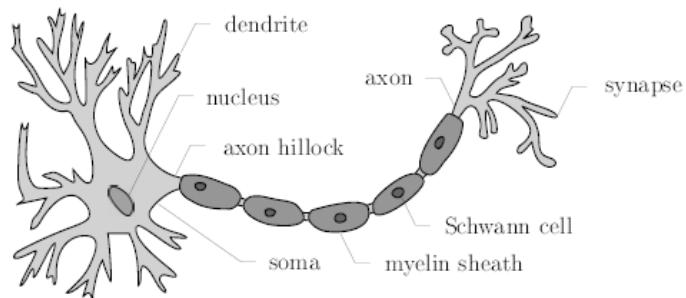
#### 2.1.1 Motoneuronen

Abbildung 2.1 zeigt den Aufbau eines Motoneurons. Der Zellkörper des Motoneurons, hier Soma genannt, liegt im Rückenmark. Es ist über das Axon mit einzelnen Muskelfasern eines Skelettmuskels verbunden. Das Motoneuron erhält elektrische Signale über die Dendriten von anderen Neuronen aus dem Rückenmark. Daraufhin leitet es dann unter bestimmten Bedingungen kurze elektrische Signale, sogenannte Aktionspotentiale, über das Axon und die neuromuskuläre Verbindung an die Muskelfasern weiter. Wenn ein Aktionspotential entlang eines Motoneurons zur neuromuskulären Verbindung propagiert wird, werden dort aufgrund der Spannung  $Ca^{2+}$ -Kanäle geöffnet. Der Anteil der  $Ca^{2+}$ -Ionen im Neuron erhöht sich, was dazu führt, dass das Motoneuron Neurotransmitter freigibt. Deshalb öffnen sich  $Na^+$ -Ionenkanäle in der Muskelfasermembran. Die weitere Ausbreitung des Aktionspotentials entlang einer Muskelfaser wird in Abschnitt 2.1.2 beschrieben.

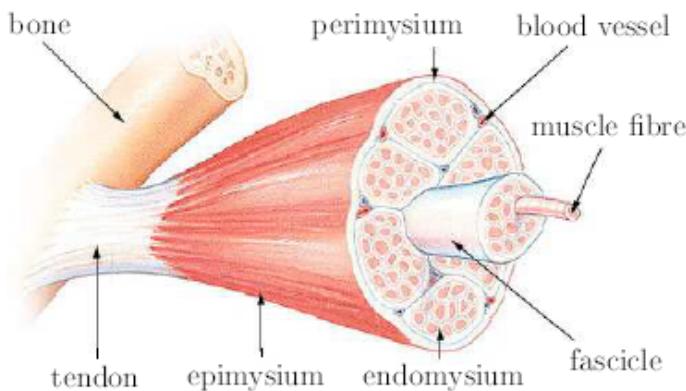
Motoneuronen können mit einer unterschiedlichen Anzahl an Muskelfasern im Muskel verbunden sein. Die Anzahl der mit ihnen verbundenen Fasern reicht von einigen Dutzend bis zu mehreren Zehntausend. Die Fasern, die mit einem bestimmten Motoneuron verbunden sind, müssen nicht alle im selben Bereich des Muskels liegen, sie können auch verteilt sein. Die Einheit von Motoneuron und damit verbundenen Muskelfasern nennt man Motor Unit.

#### 2.1.2 Skelettmuskel und Muskelfasern

Der Skelettmuskel besteht aus einzelnen Muskelfasern, die mit extrazellulärem Bindegewebe gebündelt und in einer bestimmten Form gehalten werden. In Abbildung 2.2 kann man die vom Bindegewebe (hier Epimysium, Endomysium und Perimysium) gebündelten Muskelfasern des



**Abbildung 2.1:** Motoneuron



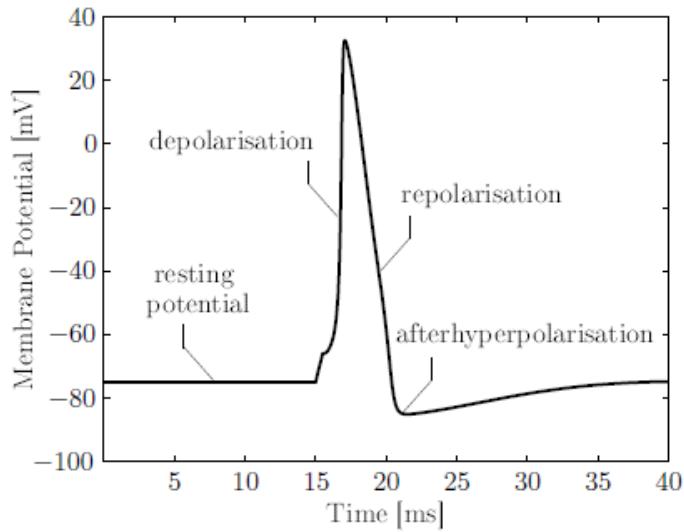
**Abbildung 2.2:** Skelettmuskel

Muskels sehen. Je nach Größe des Muskels können so zwischen einigen 100 bis zu mehreren Millionen Muskelfasern gebündelt werden.

Die Zellmembran dieser Muskelfasern enthält Ionenpumpen und Ionenkanäle. Diese ermöglichen es, Ionen aus dem Zellinneren nach außen und Ionen von außerhalb der Zelle ins Zellinnere zu befördern. Ein Beispiel für eine dieser Ionenpumpen ist die  $Na^+ - K^+$ -Ionenpumpe, die für drei  $Na^+$ -Ionen, die aus der Zelle bewegt werden, zwei  $K^+$ -Ionen ins Zellinnere befördert. Dieser Ionenaustausch führt bei Muskelfasern im Ruhezustand zu einer Spannung zwischen Zellinnerem und dem Äußeren der Zelle. Das Potential liegt ungefähr im Bereich von -70 mV bis -80 mV.

Für jedes Ion ( $Na^+$ ,  $K^+$ ,  $Cl^-$ ,  $Ca^{2+}$ ) gibt es einen Gleichgewichtszustand mit einem bestimmten elektrischen Potential, in dem die Anzahl der nach innen beförderten Ionen der Anzahl der nach außen gebrachten entspricht. Dieses liegt für  $K^+$  bei -90mV, für  $Na^+$  aber bei 75mV. Das ist weit von der Ruhespannung der Zelle entfernt, weil die Zellmembran im Ruhezustand kaum  $Na^+$ -Ionen durchlässt.[Hei]

Das ändert sich, wenn ein zusätzlicher Strom wie das Aktionspotential der Motoneuronen an einem Teil der Muskelfaser anliegt. Wenn das Aktionspotential ausreichend groß ist, kommt es zu einer Depolarisierung. Die Membran wird durchlässiger für  $Na^+$ -Ionen und die Spannung ändert sich aufgrund dieser Ionen, die nun in das Innere der Zelle gelangen können. Wenn die Spannung einen bestimmten Wert übersteigt, werden weitere Ionenkanäle geöffnet, die weitere  $Na^+$ -Ionen ins Zellinnere lassen, das Potential nähert sich der Gleichgewichtsspannung von  $Na^+$  an. Daraufhin werden die Ionenkanäle wieder geschlossen und die Zelle repolarisiert, das bedeutet,



**Abbildung 2.3:** Aktionspotential

dass das Potential wieder in den Ruhezustand zurückkehrt. War der Stimulationsstrom groß genug, wird nun das Aktionspotential entlang der Zelle propagierte, und die oben beschriebenen Schritte wiederholen sich an anderen Stellen entlang der Zellmembran.

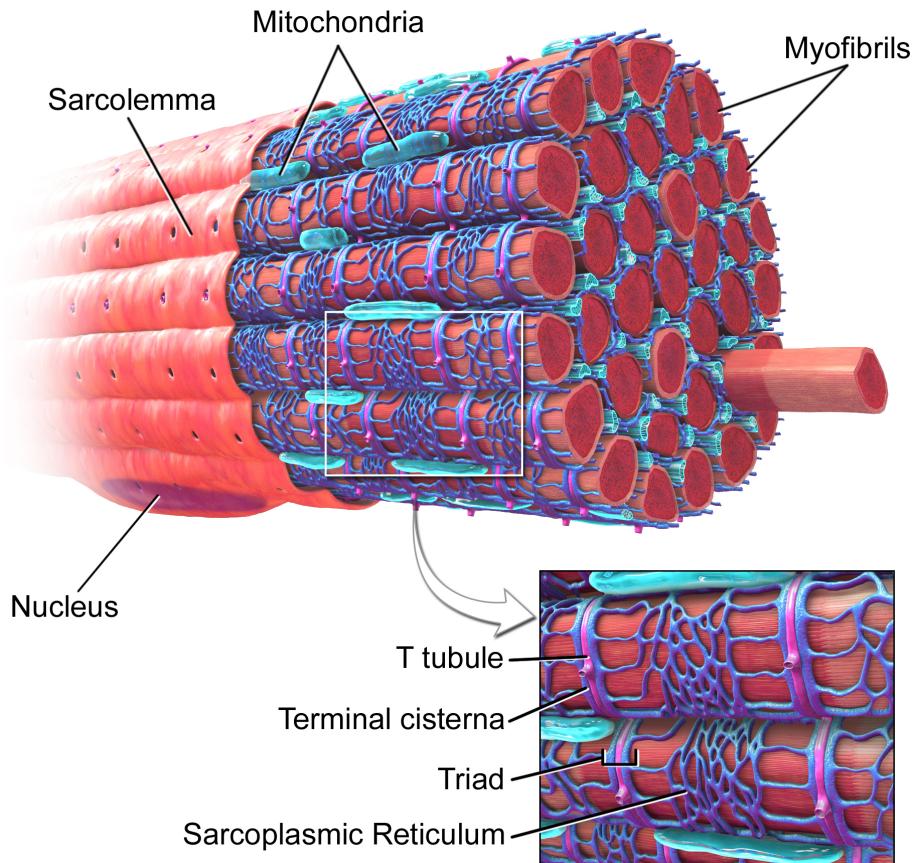
In Abbildung 2.3 ist der zeitliche Verlauf eines Aktionspotentials zu sehen. Das Ruhepotential liegt bei circa -75mV und es kommt zu einer Depolarisierung. Nach der Repolarisierung gibt es eine kurze Phase, in der die Spannung unter dem Ruhepotential liegt. In dieser Zeitspanne ist daher eine höhere Spannung nötig, die Schwellspannung zu erreichen und ein weiteres Aktionspotential zu propagieren.

Das Aktionspotential, das entlang der Zellmembran propagierte, wird über die sogenannten T-Tubuli von der Zellmembran ins Zellinnere geleitet, diese kann man in Abbildung 2.4 aus [14] sehen. Sie sind um die Myofibrillen geschlungen, die die einzelnen Muskelfasern weiter strukturieren. Myofibrillen sind Ketten von Sarkomeren. Diese Sarkomere sind die Teile des Muskels, in denen die Kraft oder Bewegung erzeugt wird. Ein Sarkomer besteht aus einer Reihe von dicken und dünnen Filamenten, Proteinen, die einander überlappen. Im Inneren der Muskelfaser führt eine Spannungsänderung zur Öffnung bestimmter Ionenkanäle und der Ausschüttung von  $Ca^{2+}$  aus dem sarkoplasmatischen Retikulum, dieses ist ein Speicher von  $Ca^{2+}$  in hoher Konzentration. Die Ionen verbinden sich mit Troponin-C-Molekülen in den Sarkomeren, was es ermöglicht, dass sich die dicken und dünnen Fasern der Sarkomere verbinden und die Sarkomere sich zusammenziehen.

## 2.2 Nichtlineare dynamische Differentialgleichungen

Jede nichtlineare dynamische Differentialgleichung (ohne Parameter) kann in folgender Form beschrieben werden:

$$\mathbf{x}'(t) = \mathbf{f}(\mathbf{x}(t), t) + \mathbf{A}(t)\mathbf{x}(t) + \mathbf{B}(t)\mathbf{u}(t) \quad (2.1)$$



**Abbildung 2.4:** Sarkomere

$$\mathbf{x}(0) = \mathbf{x}_0 \quad (2.2)$$

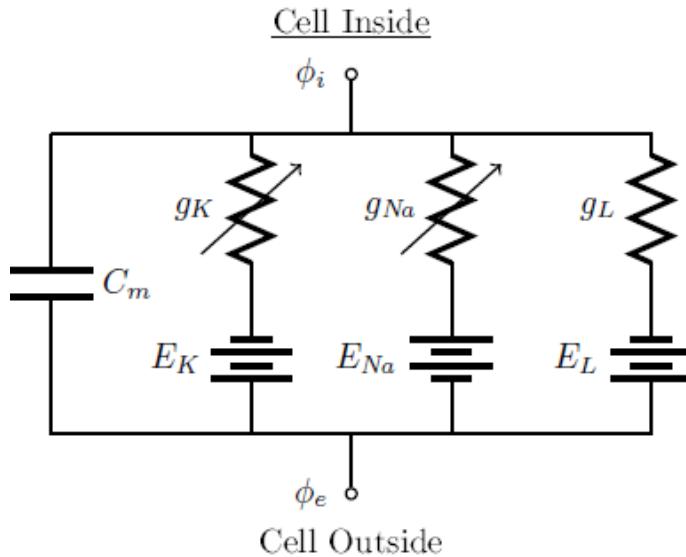
$$\mathbf{w}(t) = \mathbf{C}(t)\mathbf{x}(t) \quad (2.3)$$

Hier enthält  $\mathbf{A}(t)\mathbf{x}(t)$  mit der Matrix  $A$  den linearen Teil des Systems und  $\mathbf{f}(\mathbf{x}(t), t)$  den nichtlinearen Teil.  $\mathbf{B}(t)\mathbf{u}(t)$  ist die von der Eingabematrix  $\mathbf{B}(t)$  gewichtete Eingabefunktion und enthält somit alles, was mit der Eingabe oder dem Input zusammenhängt. [Wir14]

Die Ausgabe des Systems wird mit  $\mathbf{w}(t)$  berechnet, indem  $\mathbf{x}(t)$  mit  $\mathbf{C}(t)$  skaliert wird.

## 2.3 Hodgkin-Huxley-Modell

Das Hodgkin-Huxley-Modell ist ein biophysikalisches Modell, das dazu genutzt wird, die Spannung sowie die Entwicklung der Spannung, die an einer Zellmembran anliegt, zu simulieren. Dazu wird in diesem biophysikalischen Modell, anders als in phänomenologischen Modellen, Wissen über Funktionsweise und Aufbau von Zelle und Zellmembran verwendet. Das Modell beschreibt das Verhalten der Spannung an einer Zellmembran mit Hilfe eines elektrischen Stromkreises.[Hei]



**Abbildung 2.5:** Hodgkin-Huxley-Modell: Stromkreis

Abbildung 2.5 zeigt den Aufbau dieses Stromkreises. Wie in Abschnitt 2.1.2 beschrieben, ermöglichen es Ionenpumpen, Ionenkanäle und Durchlässigkeit der Zellmembran, die Anzahl der Ionen zu beeinflussen, die die Membran passieren. Das führt zu einer elektrischen Spannung zwischen Äußerem und Innerem der Zelle. Das Verhalten dieser Pumpen, Kanäle und der Zellmembran ist wiederum von der Spannung abhängig, die an der Membran anliegt.[Hei]

Man kann nun in Abbildung 2.5 sehen, dass Ionenpumpen und Ionenkanäle mit spannungsabhängigen Widerständen vor Stromquellen simuliert werden, die zu einem Kondensator parallel geschaltet sind. Diese spannungsabhängigen Widerstände simulieren für die einzelnen Ionen  $Na^+$  und  $K^+$ , welchen Einfluss sie bei einer bestimmten Spannung auf die Gesamtspannung haben, was davon abhängt, wie durchlässig die Zellmembran für diese Ionen ist. Die Spannungen der Stromquellen entsprechen dem Gleichgewichtszustand der jeweiligen Ionen.  $E_K$  und  $E_{Na}$  haben also das Potential des Gleichgewichtszustands von  $Na^+$  und  $K^+$ , der Wert von  $E_L$  ist nicht vom Gleichgewichtszustand eines bestimmten Ions abhängig, da damit der Leckstrom simuliert werden soll. Dieser beschreibt das Verhalten der Zellmembran, die immer für einen gewissen Anteil von  $Cl^-$  oder  $K^+$  aufgrund von Leckkanälen durchlässig ist. Der Widerstand  $g_L$  ist daher auch nicht spannungsabhängig wie die anderen beiden Widerstände im Modell.[Hei]

Das Verhalten von Widerständen und externem Simulationsstrom führt dazu, dass eine Spannung am Kondensator anliegt, die die elektrische Spannung an der Muskelfasermembran simuliert. Mit diesem Stromkreis kann man Differentialgleichungen aufstellen, mit deren Hilfe eine Approximation der Membranspannung berechnet werden kann. Aus dem Stromkreis ergibt sich Gleichung (2.4), wobei  $I_{ion}$  die Summe der Ströme im Stromkreis repräsentiert. [Hei]

$$I_m = C_m \frac{\partial V_m}{\partial t} + I_{ion} = 0 \quad (2.4)$$

$$I_{ion} = I_{Na} + I_K + I_L - I_{stim} \quad (2.5)$$

---

Die einzelnen Ströme können mit Hilfe des Ohmschen Gesetzes berechnet werden.

$$I_i = g_i(V_m - E_i) \quad \forall i \in \{Na, K, L\} \quad (2.6)$$

Die Widerstände  $g_k$  und  $g_{Na}$  sind über die Gatingvariablen n, m und h von der Spannung abhängig. Diese Variablen werden durch die Reaktionsgleichungen 2.9 berechnet. Gleichung (2.10) zeigt die Standardform der Hin- und Rückreaktionsgleichungen für die Variablen m, n und h.[Hei]

$$g_K = g_K^- n^4 \quad (2.7)$$

$$g_{Na} = g_{Na}^- m^3 h \quad (2.8)$$

$$\frac{\partial \omega}{\partial t} = \alpha_\omega(V_m)(1 - \omega) - \beta_\omega(V_m)\omega \quad \forall \omega \in \{n, m, h\} \quad (2.9)$$

$$\eta_\omega = \frac{a_\omega + b_\omega V_\omega}{c_\omega + d_\omega \exp(\frac{V_m + e_\omega}{f_\omega})} \quad \forall \eta_\omega \in \{\alpha_\omega(V_m), \beta_\omega(V_m)\} \quad (2.10)$$

Aus diesen Gleichungen folgen die nichtlinearen Differentialgleichungen in Gleichung (2.11).

$$\begin{pmatrix} \frac{\partial V}{\partial t} \\ \frac{\partial m}{\partial t} \\ \frac{\partial h}{\partial t} \\ \frac{\partial n}{\partial t} \end{pmatrix} = \begin{pmatrix} -\frac{1}{C_m}(g_{Na}^- m^3 h (V_m - E_{Na}) + g_K^- n^4 (V_m - E_K) + g_L (V_m - E_L) - I_{stim}) \\ \frac{-0.1*(V_m+50)}{-1+\exp(\frac{-(V_m+50)}{10})}(1-m) - 4 * \exp(\frac{-(V_m+75)}{18})m \\ 0.07 * \exp(-\frac{V_m+75}{20})(1-h) - \frac{1}{1+\exp(-\frac{V_m+45}{10})}h \\ \frac{-0.01*(V_m+65)}{-1+\exp(-\frac{V_m+65}{10})}(1-n) - 0.125 * \exp(\frac{V_m+75}{80})n \end{pmatrix} \quad (2.11)$$

In der weiteren Arbeit werden bei der Simulation folgende Werte und Startwerte aus [Hei] verwendet:

$C_m$	$g_{Na}^-$	$g_K^-$	$g_L$	$E_{Na}$	$E_K$	$E_L$
1 F/ cm <sup>2</sup>	120 mS / cm <sup>2</sup>	360 mS / cm <sup>2</sup>	0.3 mS / cm <sup>2</sup>	40 mV	-87 mv	-64.387 mV

$$\mathbf{x}_0 = \begin{pmatrix} V_{m_0} \\ m_0 \\ h_0 \\ n_0 \end{pmatrix} = \begin{pmatrix} -75mV \\ 0.325 \\ 0.05 \\ 0.6 \end{pmatrix} \quad (2.12)$$

---

## 2.4 Monodomain-Modell

Das Hodgkin-Huxley-Modell beschreibt das Verhalten der Muskelfasermembran an einem bestimmten Punkt. Spannungsänderungen an einer Stelle der Membran beeinflussen aber auch das elektrische Potential in anderen Bereichen entlang der Zellmembran. Dieser Zusammenhang wird durch das Monodomain-Modell beschrieben. Der Name Monodomain bezieht sich auf das Bidomain-Modell. Im Bidomain-Modell werden zur Berechnung der elektrischen Spannung elektrisch inaktive Regionen und elektrisch aktive Regionen unterschieden, im Monodomain-Modell werden nur elektrisch aktive Regionen betrachtet. In Gleichung (2.13) ist der rechte Teil schon aus Abschnitt 2.3 bekannt. Dieser wird zu einer Diffusionsgleichung über die Spannung mit dem Diffusionskoeffizienten  $\sigma_{eff}$  verbunden. In Gleichung (2.13) kann Gleichung (2.5) aus Abschnitt 2.3 eingesetzt werden. Dadurch wird die partielle Differentialgleichung des Monodomain-Modells mit dem System nichtlinearer gewöhnlicher Differentialgleichungen des Hodgkin-Huxley-Modell gekoppelt. [Hei]

$$\operatorname{div}(\sigma_{eff} \operatorname{grad}(V_m)) = A_m(C_m \frac{\partial V_m}{\partial t} + I_{ion}) \quad (2.13)$$

Für den Fall, dass es nur eine Raumdimension gibt, kann Gleichung (2.13) umgeschrieben werden in Gleichung (2.14). Mit dieser Gleichung kann nun die Propagierung des Aktionspotentials entlang der Membran einer Muskelfaser beschrieben werden. [Hei]

$$\frac{\sigma_{eff}}{A_m C_m} \frac{\partial^2 V_m}{\partial x \partial x} - \frac{1}{C_m} I_{ion} = \frac{\partial V_m}{\partial t} \quad (2.14)$$

Es gelten für die Berechnungen dieses Gleichungssystems in dieser Arbeit die Neumann-Randbedingungen.

## 2.5 Löser

Zur Lösung von Differentialgleichungen wie Gleichung (2.14) wird ein numerischer Löser benötigt. Ein Beispiel für ein numerisches Lösungsverfahren, mit dem nichtlineare dynamische Differentialgleichungen in der Form von Gleichung (2.1), Gleichung (2.2) gelöst werden können, ist das semi-implizite Eulerverfahren oder Godunov-Splitting, das eine Kombination aus implizitem und explizitem Eulerverfahren ist. [Hei]

Gleichung (2.1) lässt sich mit diesem Verfahren in zwei Differentialgleichungen in folgender Form aufteilen:

$$\begin{aligned} \mathbf{x}'(t) &= \mathbf{A}(t)\mathbf{x}(t) \\ \mathbf{x}'(t) &= \mathbf{f}(\mathbf{x}(t), t) + \mathbf{B}(t)\mathbf{u}(t) \end{aligned} \quad (2.15)$$

Also kann ihre Lösung approximiert werden, indem zuerst der nichtlineare Teil  $x^*(t+1)$  explizit berechnet wird und dann das lineare Gleichungssystem  $dt * \mathbf{A}(t)\mathbf{x}((t+1)) = \mathbf{x}^*(t+1)$  nach  $\mathbf{x}((t+1))$  gelöst wird. [Hei]

---

## 2.6 Model Order Reduction

Model-Order-Reduction-Methoden sind eine Klasse mathematischer Verfahren, die das Ziel haben, Rechenzeit und Komplexität von dynamischen Systemen, die viel Rechenzeit benötigen und sehr komplex sind, zu reduzieren. Diese Verfahren arbeiten unter der Prämisse, dass sich der Mechanismus, der dem zu reduzierenden System zugrunde liegt, in einer niedrigeren Dimension effektiv approximiert werden kann.

Methoden der Model Order Reduction, die häufig eingesetzt werden, sind Unterraumprojektionsverfahren. Diese verwenden einen Unterraum, auf den das komplexe System reduziert wird. In diese Gruppe gehören das POD-Galerkin-Verfahren und DEIM (Discrete Empirical Interpolation Method), deren Anwendung auf nichtlineare dynamische Differentialgleichungen in diesem Kapitel besprochen wird. Davor wird auf das POD-Verfahren (Proper Orthogonal Decomposition) eingegangen, das die Basisvektoren eines möglichst optimalen Unterraums approximieren soll. Die Algorithmen DEIM und POD, die in diesem Kapitel vorkommen, werden in der Simulation der Ergebnisse in dieser Arbeit verwendet, wie sie im MOR-Framework KerMor implementiert wurden und können in [Wir14] und [CS10] nachgelesen werden. [CS10] [Wir14]

### 2.6.1 POD

Die Proper Orthogonal Decomposition oder POD ist eine Methode, mit der eine Unterraumapproximation im Hilbertraum berechnet wird. Die Basis dieser Projektion ist die Menge der orthonormalen Vektoren  $\{\phi_i\}$ , die das Minimierungsproblem Gleichung (2.17) erfüllen.[CS10] [Wir14]

$$\begin{aligned} \min_{\{\phi_i\}_{i=1}^k} & \sum_{j=1}^n \|x_j - \sum_{i=1}^k (x_j^T \phi_i) \phi_i\|_2^2 \\ & \phi_i^T \phi_j = \delta_{ij} \end{aligned} \tag{2.17}$$

Die optimale Basis zur Lösung dieses Optimierungsproblems kann für eine Matrix  $X$  berechnet werden, indem die Singulärwertzerlegung auf dieser Matrix durchgeführt wird. In unserem Fall besteht diese Matrix  $X$  aus den Trajectories  $X = \mathbf{x}^1, \dots, \mathbf{x}^n$ . Jedes  $\mathbf{x}^i$  ist eine Ergebnismatrix der Berechnung des Modells zu den jeweiligen Zeitschritten, die in den späteren Berechnungen verwendet werden sollen. Für die verschiedenen Matrizen  $\mathbf{x}^i$  können somit zum Beispiel unterschiedliche Ergebnisse für verschiedene Eingaben verwendet werden. Die  $k$  Basisvektoren eines Unterraums mit Dimension  $k$  sind dann die ersten  $k$  Spaltenvektoren der Matrix  $V$  in Gleichung (2.18), die die linken Singulärvektoren nach der Singulärwertzerlegung enthält. Außerdem ist hier kurz anzumerken, dass aufgrund des Zusammenhangs in Gleichung (2.19) die Anzahl der Singulärvektoren, die für eine Approximation mit einer bestimmten Genauigkeit benötigt werden, daran erkannt werden kann, wie stark die Wertabnahme der Singulärwerte ist.[CS10] [Wir14]

---


$$Y = V\Sigma W^T \quad (2.18)$$

$$\sum_{j=1}^{n_s} \|y_i - \sum_{i=1}^k (y_j^T v_i) v_i\|_2^2 = \sum_{i=k+1}^r \sigma_i^2 \quad (2.19)$$

## 2.6.2 POD-Galerkin

Mit der Matrix  $V$ , die mit POD berechnet wurde, kann jetzt das reduzierte System zu einem nichtlinearen dynamischen Gleichungssystem berechnet werden. Die volle Ausgabe des Systems kann aus der reduzierten Ausgabe mit

$\mathbf{y}(t) = \mathbf{C}\mathbf{V}_k \tilde{\mathbf{x}}(t)$  und  $\dim(\mathbf{y}(t)) = n$  berechnet werden.  $\mathbf{V}_k$  beschreibt hier die Matrix  $V$  bis zum  $k$ -ten Spaltenvektor mit  $\dim(V_k^T) = k \times n$ . Damit hat  $y(t)$  die Dimension  $k$ . Die Startwerte des reduzierten Systems werden aus den Anfangswerten berechnet, die für das volle System gegeben sind, wie in 2.20 ersichtlich ist.[CS10] [Wir14]

$$\tilde{\mathbf{x}}(0) = V_k^T \mathbf{x}(0) = V_k^T \mathbf{x}_0 \quad (2.20)$$

So kommt man auf Gleichung (2.21) für das reduzierte Modell. A wird hier mit  $\mathbf{V}_k^T A \mathbf{V}_k$  auf eine  $k \times k$ -Matrix reduziert. Die Ergebnisse des reduzierten Modells werden in der nichtlinearen Funktion F mit  $V_k \tilde{\mathbf{x}}$  auf  $\dim(x(t)) = n$  projiziert, also auch  $n$  mal in der Funktion ausgewertet. Diese Eigenschaft ist einer der Hauptnachteile des POD-Galerkin-Ansatzes. Die Reduzierung wirkt sich nämlich nicht auf die Auswertung der nichtlinearen Funktion oder der beiden Matrix-Vektor-Multiplikationen aus. Diese hängen immer noch von  $n$  ab. Je nach Berechnungsaufwand der nichtlinearen Funktion kann hier aber die Hauptlast der Berechnung liegen.[CS10] [Wir14]

Die Matrix  $\mathbf{V}_k$  sowie die Multiplikationen mit den anderen konstanten Matrizen können vor der eigentlichen Berechnung des reduzierten Modells in der sogenannten Offline-Phase berechnet werden. Der beabsichtigte Zeitgewinn erfolgt, wenn diese Matrizen verwendet werden können, das System in der Online-Phase für mehrere verschiedene Parameter oder Eingabefunktionen mit akzeptabler Genauigkeit zu berechnen.[CS10] [Wir14]

$$\begin{aligned} \frac{d}{dt} \tilde{\mathbf{x}}(t) &= \mathbf{V}_k^T \mathbf{A} \mathbf{V}_k \tilde{\mathbf{x}}(t) + \mathbf{V}_k^T \mathbf{F}(\mathbf{V}_k \tilde{\mathbf{x}}(t)) + \mathbf{V}_k^T \mathbf{B} \mathbf{u}(t) \\ \mathbf{y}(t) &= \mathbf{C} \mathbf{V}_k \tilde{\mathbf{x}}(t) \end{aligned} \quad (2.21)$$

## 2.6.3 DEIM

Ein Ansatz, das Problem, dass die Berechnung des reduzierten Modells immer noch von  $n$  abhängt zu lösen, das im POD-Galerkin-Ansatz auftrat, ist die Deterministische Empirische Interpolationsmethode (DEIM). Deren Ansatz ist es, auch die nichtlineare Funktion auf einen Unterraum, hier  $U$ , zu reduzieren und damit die Anzahl der Auswertungen der Funktion zu verringern.[CS10]

Das volle Ergebnis der Funktion F soll über Interpolation eines Koeffizientenvektors  $c(\tau)$  mit der Matrix U berechnet werden können. Also gilt  $P^T f(\tau) = (P^T U)c(\tau)$ , wobei P eine Auswahl

---

**Algorithmus 2.1** DEIM

---

```

procedure DEIM2( $u_{l=1}^m \subset \Re^n$ )
    INPUT :  $u_{l=1}^m$ 
    OUTPUT :  $\vec{\varphi} = [\varphi_1, \dots, \varphi_m]^T \in \Re^m$ 
     $[\|\rho\|, \varphi_1] \leftarrow \max |\mathbf{u}_1|$ 
     $\mathbf{U} \leftarrow \mathbf{u}_1, \mathbf{P} \leftarrow [\mathbf{e}_{\varphi_1}], \vec{\varphi} = [\varphi]$ 
    for  $l = 2 \dots m$  do
        Solve  $(\mathbf{P}^T \mathbf{U}) \mathbf{c} = \mathbf{P}^T \mathbf{u}_l$  for  $\mathbf{c}$ 
         $\mathbf{r} = \mathbf{u}_l - \mathbf{U} \mathbf{c}$ 
         $[\|\rho\|, \varphi_l] = \max |\mathbf{r}|$ 
         $\mathbf{U} \leftarrow [\mathbf{U} \mathbf{u}_l], \mathbf{P} \leftarrow [\mathbf{P} \mathbf{e}_{\varphi_l}], \vec{\varphi} \leftarrow \begin{bmatrix} \vec{\varphi} \\ \varphi_l \end{bmatrix}$ 
    end for
end procedure

```

---

von Einheitsvektoren ist. Diese Eigenschaften können umgestellt werden zu  $f(\tau) \approx U c(\tau) = U(P^T U)^{-1} P^T f(\tau)$ . Es wird also ausgenutzt, dass  $P^T$  dünn besetzt ist und somit  $F$  nur noch an  $k$  Stellen ausgewertet werden muss. Zu beachten ist, dass dieser Vorteil nur erhalten bleibt, wenn die Auswertung von  $F$  an einem bestimmten Punkt nicht von zu vielen benachbarten Punkten abhängt.[CS10]

Der DEIM-Algorithmus ist ein Greedy-Algorithmus. Für einen neuen Basisvektor  $u_l$  wird der Punkt ermittelt, an dem die größte Differenz in der Approximation im Vergleich zur Berechnung mit den schon gewählten Basisvektoren liegt. Für diesen wird dann ein neuer Einheitsbasisvektor zu  $P$  hinzugefügt, um die Approximation in diese Richtung zu verbessern.[CS10] [Wir14] [CS10]

Damit ergibt sich folgendes Gleichungssystem, wenn man POD-Galerkin mit DEIM kombiniert[CS10]:

$$\begin{aligned} \frac{d}{dt} \tilde{\mathbf{x}}(t) &= \mathbf{V}_k^T \mathbf{A} \mathbf{V}_k \tilde{\mathbf{x}}(t) + \mathbf{V}_k^T \mathbf{U} (\mathbf{P}^T \mathbf{U})^{-1} \mathbf{P}^T \mathbf{F} (\mathbf{V}_k \tilde{\mathbf{x}}(t)) + \mathbf{V}_k^T \mathbf{B} \mathbf{u}(t) \\ \mathbf{y}(\mathbf{t}) &= \mathbf{C} \mathbf{V}_k \tilde{\mathbf{x}}(t) \end{aligned} \quad (2.22)$$

# 3 Ergebnisse

In diesem Kapitel wird beschrieben, wie bei der Berechnung der Ergebnisse vorgegangen wurde, welche Schwierigkeiten es bei der Berechnung gab, sowie zuletzt, zu welchen Ergebnissen die Berechnung mit unterschiedlichen MOR-Methoden führte. Zuerst wird in Abschnitt 3.1 beschrieben, wie das System diskretisiert wurde, in Abschnitt 3.2 wird besprochen, was KerMor ist und welche Vorteile oder Nachteile die Berechnung durch KerMor hat. Im Abschnitt 3.3 werden die ersten Ergebnisse vorgestellt. Es geht in diesem Abschnitt darum, ob sich das Modell für die Propagation von Aktionspotentialen entlang von Muskelfasern überhaupt für MOR-Methoden eignet. Ergebnisse der Reduktion mit MOR-Methoden werden dann in Abschnitt 3.4, Abschnitt 3.5 und Abschnitt 3.6 besprochen.

## 3.1 Kombination der Modelle und Diskretisierung

Die Modelle Hodgkin-Huxley und Monodomain werden für die Berechnungen des Systems kombiniert, wie in Gleichung (2.14) beschrieben. Für das System gelten Neumann-Randbedingungen, das heißt, dass es keine Diffusion "über den Rand hinaus" gibt. Die partielle Ableitung zweiter Ordnung nach der Raumdimension  $x$  wird mit einer central-difference-Approximation zweiter Ordnung nach der Finite-Elemente-Methode diskretisiert. Diese Diskretisierung führt zur Matrix  $A$ , die mit der Spannung  $V$  multipliziert wird und damit den linearen Teil dieses nichtlinearen Modells formt Gleichung (3.1). Diese Matrix und damit das Modell können nach dieser Entscheidung auf zwei verschiedene Weisen diskretisiert werden, welche sich auf den Fehler der Approximation und die Laufzeit auswirken können. Entweder werden die Ergebnisse der Berechnung der Gating-Variablen  $m$ ,  $n$  und  $h$  ins Gesamtmodell übernommen oder sie werden in der Funktion  $f$  berechnet, aber nicht als Ergebnis ausgegeben.

Beide Entscheidungen haben auf den ersten Blick Vor- und Nachteile. Der erste Fall sorgt dafür, dass die Matrix  $A$  um den Faktor vier größer ist als im zweiten Fall und damit die Berechnungsdauer von Multiplikationen mit dieser Matrix größer wäre. Ein Vorteil dieses Ansatzes wäre aber, dass mehr Daten zur Verfügung stehen, um mit POD die Unterraumapproximation  $V$  zu berechnen. Der Grundgedanke der Unterraumapproximation war, dass sich hinter einem komplexen System ein Mechanismus niedrigerer Dimension verbirgt. Wenn die entscheidende Variable nun  $m$ ,  $n$  oder  $h$  wäre, dann kann eine Approximation, die nur die Spannungswerte zur Berechnung bekommt, schlechtere Ergebnisse produzieren.

$$\frac{\partial \mathbf{V}_m}{\partial t} = \frac{\sigma_{eff}}{A_m C_m dx^2} \mathbf{A} \mathbf{V}_m + \left(-\frac{1}{C_m}\right) \mathbf{F}(\mathbf{V}_m, t) + \left(\frac{1}{C_m}\right) \mathbf{B} u(t) \quad (3.1)$$

---

## 3.2 KerMor

Für die Untersuchung von Model-Order-Reduction-Verfahren und dem oben beschriebenen Modell zur Approximation der Propagation von Aktionspotentialen entlang der Membran von Muskelfasern wurde das Framework KerMor verwendet, das in MatLab bereitgestellt wird. Dieses enthält eine Reihe von Methoden und Klassen, die mit Model-Order-Reduction zu tun haben oder für diese benötigt werden können. Darunter sind Löser, der DEIM-Algorithmus und Algorithmen wie POD, Greedy-POD und POD-Galerkin. Implementiert wurden im Zuge der Untersuchung die beiden oben genannten Diskretisierungen des Systems. Das beinhaltet Methoden zum Berechnen der verschiedenen Matrizen (lineare Matrix A, Eingabematrix B, Ausgabematrix C), verschiedene Inputfunktionen und Methoden zur Berechnung der nichtlinearen Funktion an einzelnen bestimmten Punkten oder an allen Punkten in einem bestimmten Zeitschritt, sowie eine Methode zur Berechnung der analytischen Lösung der Jacobimatrix dieser Funktion, da die Approximation mit finiten Differenzen einen zu großen Fehler hatte.

Es mussten einige Änderungen am Framework vorgenommen werden, um vergleichbare Testergebnisse zu erhalten. So war es notwendig, einige Fehler zu korrigieren, die auftraten, wenn man mit Systemen rechnete, die nicht parametrisiert waren. Zudem wurde die Zeitmessung leicht abgeändert, um die tatsächliche Laufzeit der Approximationen zu vergleichen, da ein Problem mit der Messung der Laufzeit von DEIM auftrat. Wenn DEIM in KerMor aktiviert wird, finden mehr Funktionsaufrufe statt, bevor die nichtlineare Funktion ausgewertet werden kann. Diese Aufrufe vor jeder Berechnung von F wurden ursprünglich mitgemessen und verzerrten die Ergebnisse. Bei den Tests stellte sich außerdem heraus, dass kleinste Änderungen am Code zu großen Änderungen in den gemessenen Zeiten führen konnten, deshalb wurde versucht, den Code in den Methoden zur Berechnung der Ergebnisse für alle Punkte und zur Berechnung der Ergebnisse für bestimmte Punkte möglichst vergleichbar zu programmieren.

Für die numerische Lösung der Ergebnisse wurde immer das semi-implizite Eulerverfahren als Löser verwendet. Die in KerMor implementierte Version hat die Eigenschaft, dass immer nur ein Schritt möglich ist und nicht mehrmals in kleineren Zeitschritten zum Beispiel bei der expliziten Lösung des nichtlinearen Teils der Gleichung iteriert werden kann. Die Zeitschritte für expliziten und impliziten Teil können außerdem nicht unabhängig voneinander gewählt werden.

In den Tests wird für die Berechnung der Matrix V bei POD-Galerkin und Deim das Gesamtergebnis von V, m, n und h verwendet. Für die Berechnung der Matrizen U und P bei DEIM wird die explizite Lösung der nichtlinearen Funktion verwendet, also das Teilergebnis dieser Funktion für V, m, n und h.

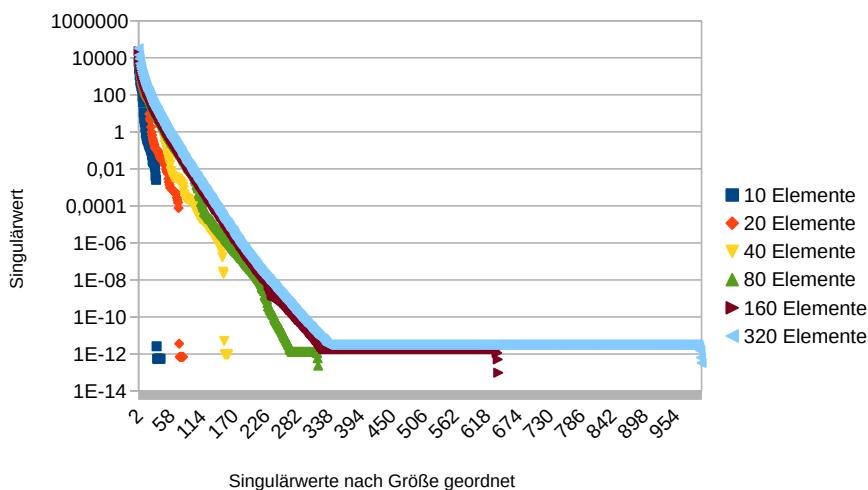
## 3.3 Singulärwerte

Der Singulärwertzerfall ist ein guter Hinweis darauf, wie erfolgreich die Anwendung von Unterraumprojektionsverfahren auf bestimmte Systeme ist Abschnitt 2.6.1. Deshalb wurde der Singulärwertzerfall für verschiedene Mengen von diskreten Elementen und verschiedene Größen der Zeitschritte für die Systeme mit Matrix V,m,n,h und V untersucht. Die Länge der Muskelfaser wurde während aller Simulationen bei einem Zentimeter belassen, was bedeutet, dass sich mit der Anzahl der diskreten Elemente ihre Größe ändert.

Außerdem wurde der Stimulationspunkt in allen Versuchen auf das dritte diskrete Element einer Seite festgelegt. In der Muskelfaser kann diese Stimulation auch an einer anderen Stelle eher in der Mitte der Faser erfolgen, aber aufgrund der Symmetrieeigenschaften der Ausbreitung des Aktionspotentials wäre es überflüssiger Rechenaufwand, den Stimulationspunkt in die Mitte zu legen.

In diesem und jedem Fall ohne andere Angabe in den weiteren Kapiteln wurde über eine Zeitdauer von 10ms die Ausbreitung des Aktionspotentials simuliert, der Stimulationsstrom liegt am Anfang der Simulation für 0.5ms an und die Muskelfaser hat eine Länge von einem Zentimeter. In jedem Fall, in dem nicht explizit die Größe eines Zeitschrittes für die Simulation angegeben ist, wurde das System mit Zeitschritten von 0.0005ms gelöst.

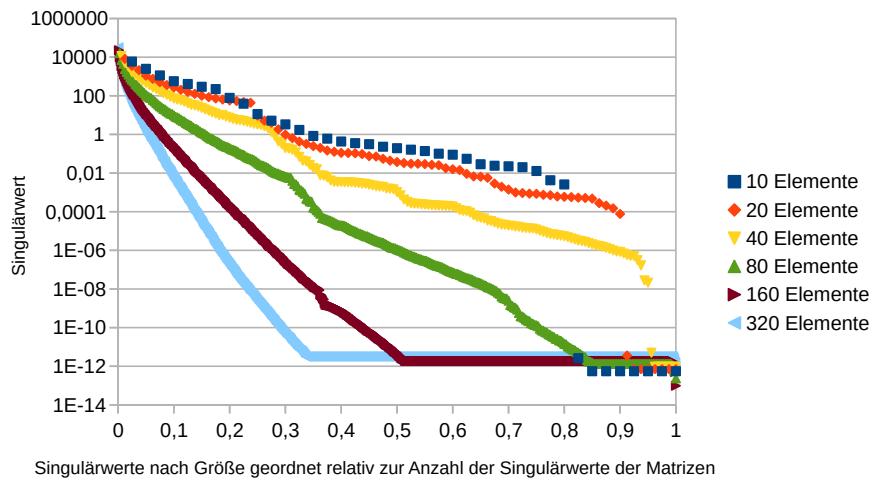
Abbildung 3.1 zeigt den Zerfall von Singulärwerten für eine unterschiedliche Anzahl von Elementen des Systems mit  $V, m, n$  und  $h$  außerhalb der nichtlinearen Funktion. Man erkennt, dass es für jede Anzahl von Elementen einen steilen Abfall der Singulärwerte gibt und die Anzahl der Singulärwerte mit sehr niedrigem Wert zunimmt, je höher die Anzahl der Elemente wird. In Abbildung 3.2 sind die Werte im Verhältnis zu der Gesamtzahl an Singulärwerten des Ergebnisses angegeben. Die Grafik zeigt, dass der Zerfall der Singulärwerte mit der Anzahl an Elementen ansteigt. Bei 10 diskreten Elementen ist er noch flach, bei 320 schon so steil, dass schon ab weniger als 40% der Gesamtzahl von Singulärwerten die niedrigste Ordnung erreicht wird.



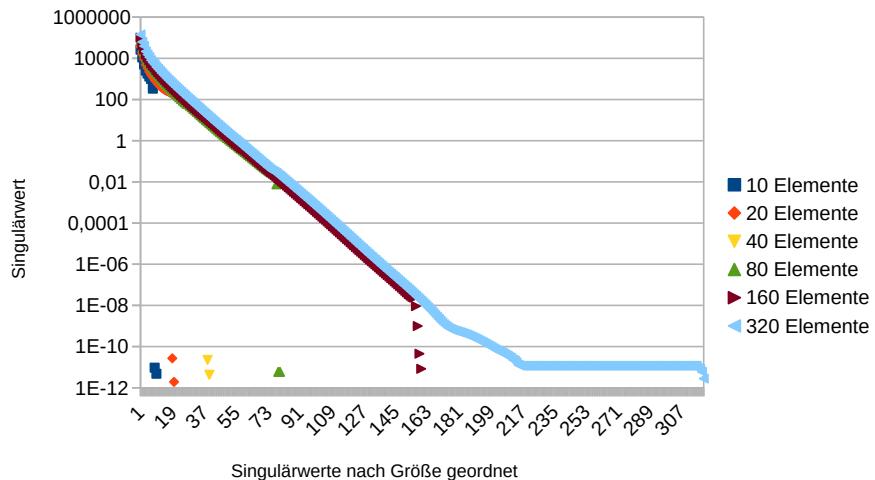
**Abbildung 3.1:** Singulärwerte für Matrix mit  $V, m, n$  und  $h$ -Werten (absolut)

Im Vergleich zu diesen Ergebnissen für  $V, m, n, h$  ist der Zerfall der Singulärwerte in Abbildung 3.3 und Abbildung 3.4 sowohl in relativer als auch absoluter Betrachtung flacher. Es zeigt sich aber auch hier, dass die Zerfallsrate mit Verringerung der Größe der diskreten Elemente steigt.

Diese Ergebnisse lassen auf den ersten Blick darauf schließen, dass es wahrscheinlich besser funktioniert, die erste Diskretisierungsvariante mit Unterraumprojektionsverfahren zu reduzieren, aber auch eine Beschleunigung der Berechnung erfolgen kann, wenn die zweite Variante reduziert wird.



**Abbildung 3.2:** Singulärwerte für Matrix mit V, m, n und h-Werten (relativ)

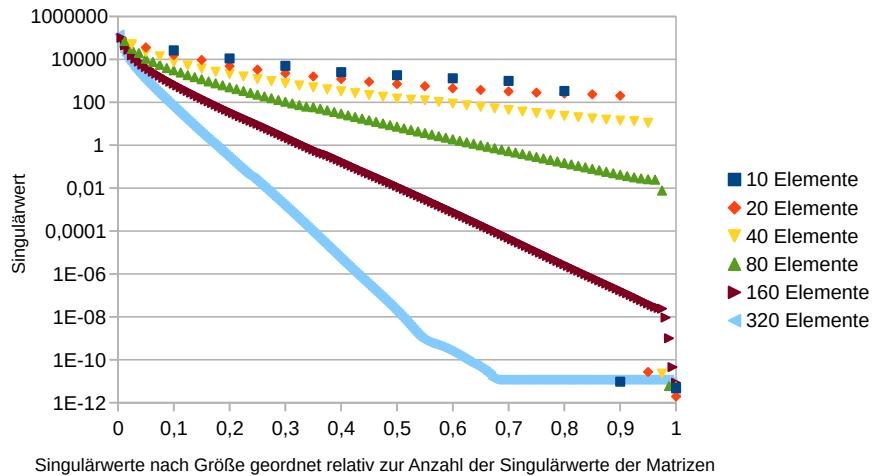


**Abbildung 3.3:** Singulärwerte für Matrix mit V-Werten (absolut)

### 3.4 Einzelnes Aktionspotential

In diesem Kapitel werden die Ergebnisse der MOR-Verfahren DEIM, POD-Galerkin, DEIM + POD-Galerkin für eine Inputfunktion mit einem Stimulationsstrom zu Beginn der Simulation mit einer Dauer von 0.5ms verglichen. Es wird über 10ms simuliert. Zuerst wurden Werte, wie Beschleunigung des reduzierten Modells in der Online-Phase gegenüber dem vollen Modell sowie der Berechnungsfehler, für die Methode POD-Galerkin mit verschiedenen Zeitschritten und unterschiedlicher Anzahl der diskreten Elemente gemessen.

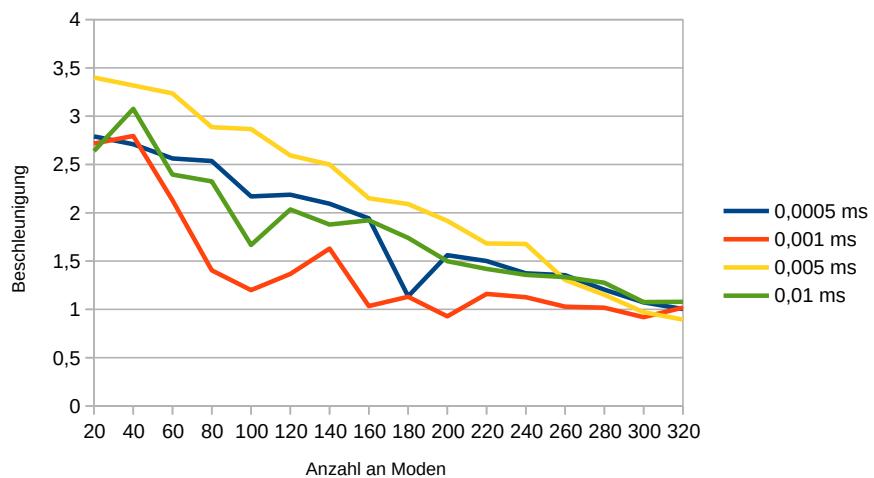
Abbildung 3.5 und Abbildung 3.6 zeigen, dass die Änderung der Größe der Zeitschritte keinen erkennbaren Effekt auf Beschleunigung oder Fehler hat, wenn man die Dimension des reduzierten Models ändert. Die einzige Beobachtung, die gemacht werden kann, ist, dass mit einem größeren



**Abbildung 3.4:** Singulärwerte für Matrix mit V-Werten (relativ)

Zeitschritt der Fehler kleiner wird, wenn die Dimension des reduzierten POD-Galerkin-Modells nahe an der des vollen Modells liegt.

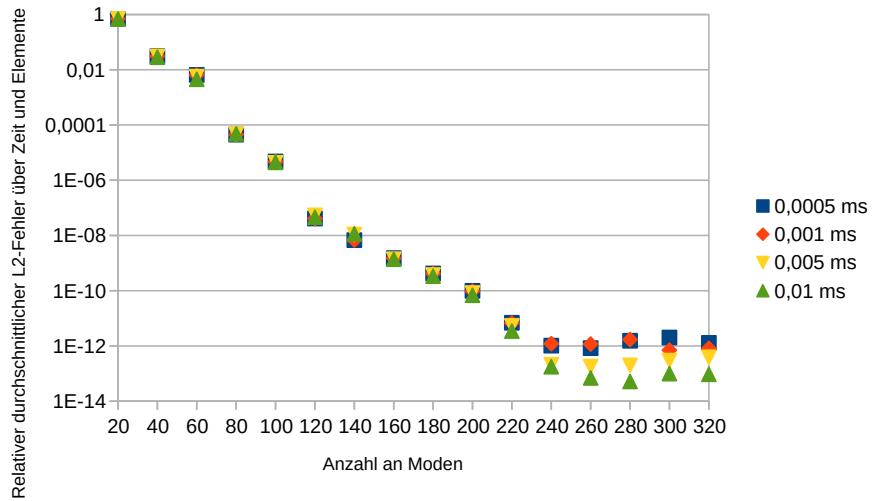
Die X-Werte geben jeweils die Dimension des mit POD-Galerkin reduzierten Modells für 80 diskrete Elemente an, die Y-Werte, um welchen Faktor das reduzierte Modell schneller ist oder den durchschnittlichen reduzierten L2-Fehler des Modells über Raum und Zeit.



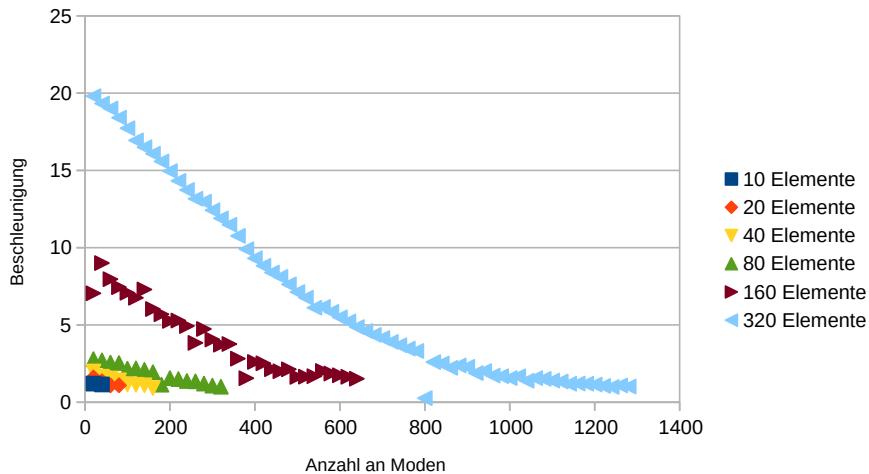
**Abbildung 3.5:** POD-Galerkin Beschleunigung über verschiedene Zeitschritte

Abbildung 3.8 zeigt, dass die Beschleunigung der Berechnung für das gleiche Verhältnis der Dimensionen von vollem und reduziertem Modell steigt, Abbildung 3.10, dass der Fehler für dieses Verhältnis der Dimensionen des vollen und des reduzierten Modells sinkt.

Es ergibt sich also in Abbildung 3.11 die Beobachtung, dass mit steigender Anzahl an Elementen für denselben Fehler eine größere Beschleunigung erreicht werden kann. Diese Ergebnisse würden zu dem Schluss führen, dass das Modell mit der Diskretisierungsvariante, in der die Ergebnisse



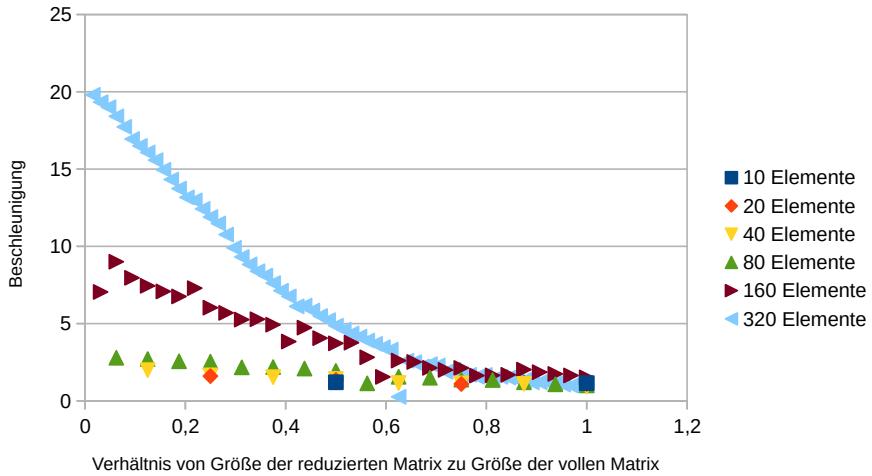
**Abbildung 3.6:** POD-Galerkin Fehler über verschiedene Zeitschritte



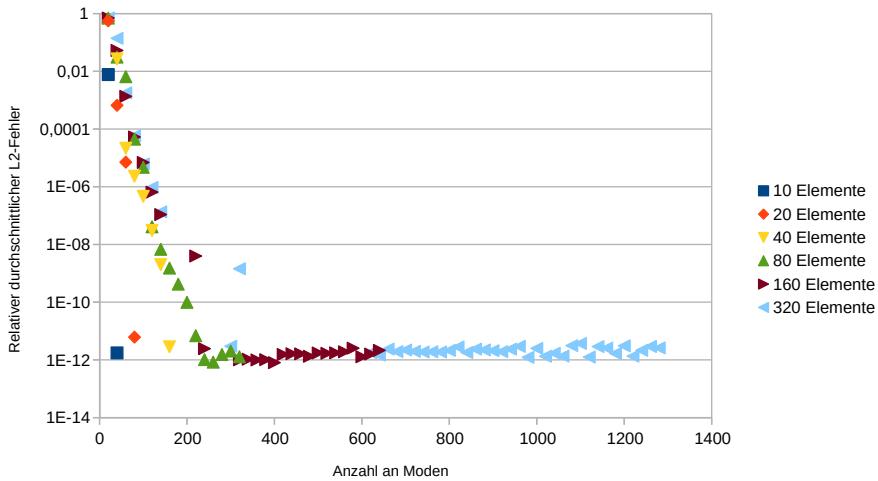
**Abbildung 3.7:** POD-Galerkin Beschleunigung (absolut)

aller Variablen von der nichtlinearen Funktion  $F$  übergeben werden, erfolgreich mit der POD-Galerkin-Methode beschleunigt werden kann. Ein Problem dieses Ansatzes ist aber, dass es ab einer gewissen Anzahl an Elementen einen Bereich gibt, in dem das reduzierte Modell instabil ist. Zum Beispiel gibt es im reduzierten Modell für 320 Elemente einen Bereich zwischen 160 und 600 Moden, in dem das System instabil ist.

In den nächsten Simulationen wurden die Ergebnisse der DEIM-Methode zuerst ohne POD-Galerkin-Methode simuliert. In Abbildung 3.12 ist zu sehen, dass die Beschleunigung für DEIM nicht so stark wie in POD-Galerkin ansteigt, wenn weniger Punkte ausgewertet werden und diese interpoliert werden. Außerdem sinkt im Gegensatz zu POD-Galerkin die Beschleunigung, je mehr Elemente simuliert werden. Der Fehler in Abbildung 3.13 nimmt außerdem nicht so gleichmäßig ab, wie bei POD-Galerkin. Diese Verteilung der Fehlerwerte kann man sehr gut in Abbildung 3.14 für 320 Elemente sehen.



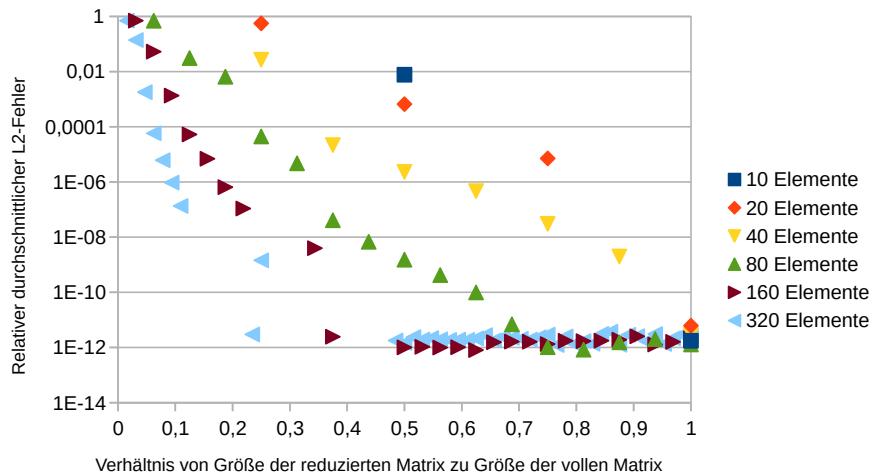
**Abbildung 3.8:** POD-Galerkin Beschleunigung (relativ)



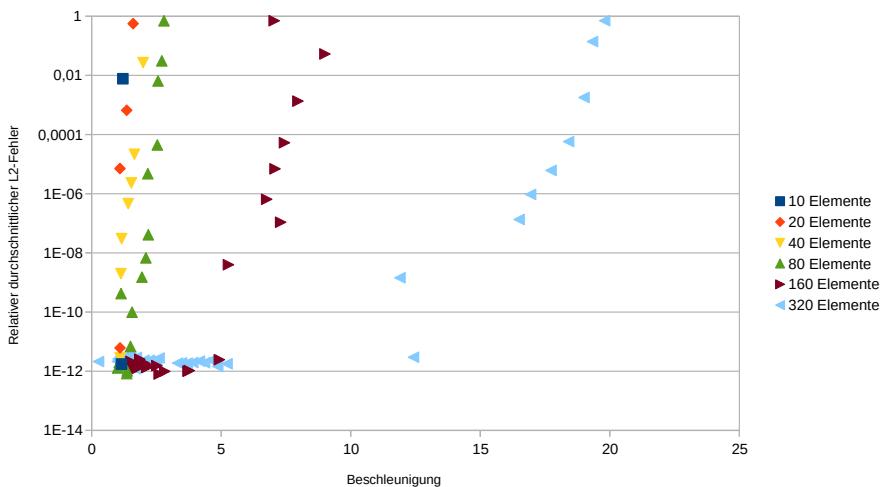
**Abbildung 3.9:** POD-Galerkin Fehler (absolut)

Im nächsten Simulationsdurchlauf wurden daraufhin die Projektion der nichtlinearen Funktion und des Gesamtmodells auf jeweils einen Unterraum getestet, zuerst wurde die Dimension des reduzierten Modells außerhalb der nichlinearen Funktion geändert. Die Anzahl an Punkten, die in der Funktion berechnet werden, wurde auf das 0.8-fache der Gesamtzahl festgesetzt. In Abbildung 3.15 kann man erkennen, dass diese Kombination im Vergleich zu DEIM ohne Projektion des Gesamtmodells zu einem gleichmäßigeren Anstieg der Beschleunigung führt, das Modell auf diese Art zu reduzieren aber keine Zeitvorteile gegenüber der POD-Galerkin-Variante hat, der Fehler in Abbildung 3.16 entwickelt sich auch sehr ähnlich zu POD-Galerkin ohne DEIM.

Daraufhin wurde das Verhalten geprüft, wenn die Anzahl der Moden der Projektion des Gesamt- systems für 80 Elemente auf 200 festgesetzt ist. Auch in diesem Fall gibt es keine Vorteile der Kombination, in Abbildung 3.17 und Abbildung 3.18 schneidet DEIM ohne POD sogar sowohl im Fehler als auch in der Beschleunigung besser ab als die Kombination von beiden.



**Abbildung 3.10:** POD-Galerkin Fehler (relativ)

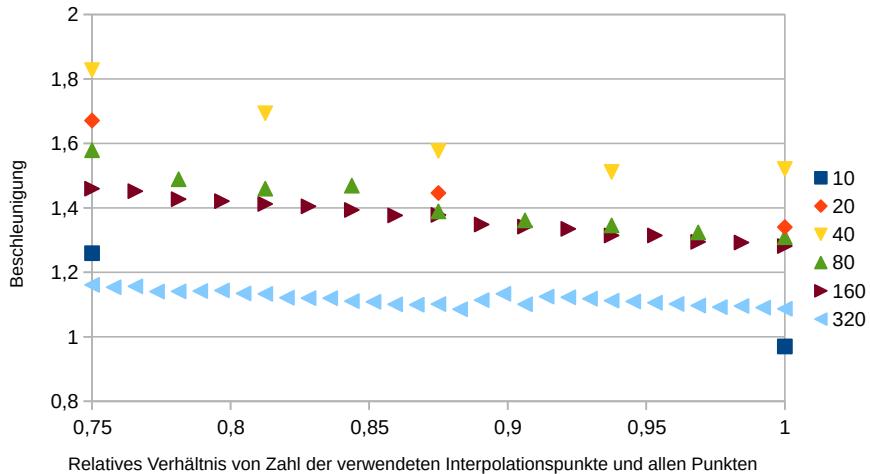


**Abbildung 3.11:** POD-Galerkin Fehler-Beschleunigungsdiagramm

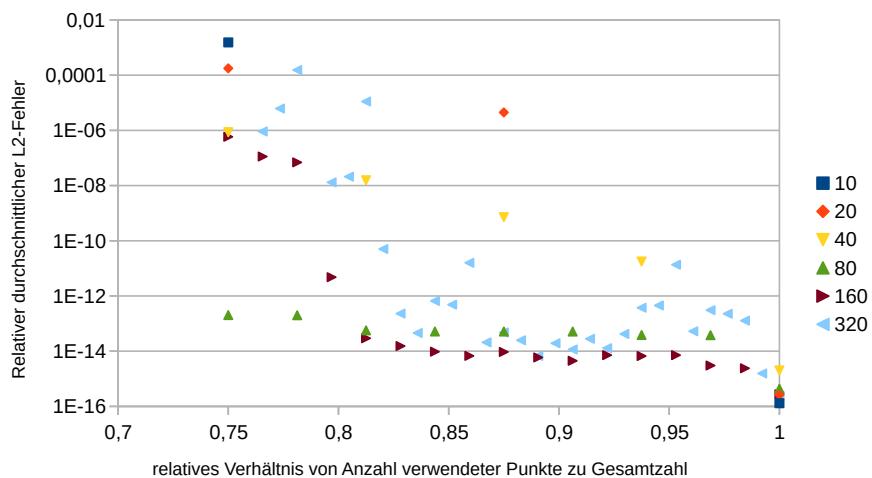
### 3.5 Mehrere Trajectories

Nachdem gezeigt wurde, dass es möglich ist, ein System zu reduzieren, wird nun untersucht, ob es auch möglich ist, die Moden für das POD-Galerkin-Verfahren aus den Ergebnissen von Berechnungen mit verschiedenen Eingabefunktionen zu berechnen, und mit diesen dann akzeptable Fehler- und Beschleunigungswerte für verschiedene Eingaben zu erzielen, womit es möglich wäre, mit den MOR-Verfahren die Berechnung zu beschleunigen, da man somit nicht für jede Eingabe eine Projektionsmatrix ausreichender Größe vorberechnen muss.

In dieser Simulation wurden daher die Ergebnismatrizen für 80 diskrete Elemente und fünf Eingabefunktionen über 200ms simuliert. Die Eingabefunktionen wenden einen Stimulationsstrom auf ein Element mit einer Frequenz zwischen 50Hz und 90Hz für jeweils 0.05ms an. Dann wurden die Moden der Matrix dieser Ergebnisse ausgerechnet. Mit diesen Basen wurden dann die reduzierten



**Abbildung 3.12:** DEIM Beschleunigung (relativ)

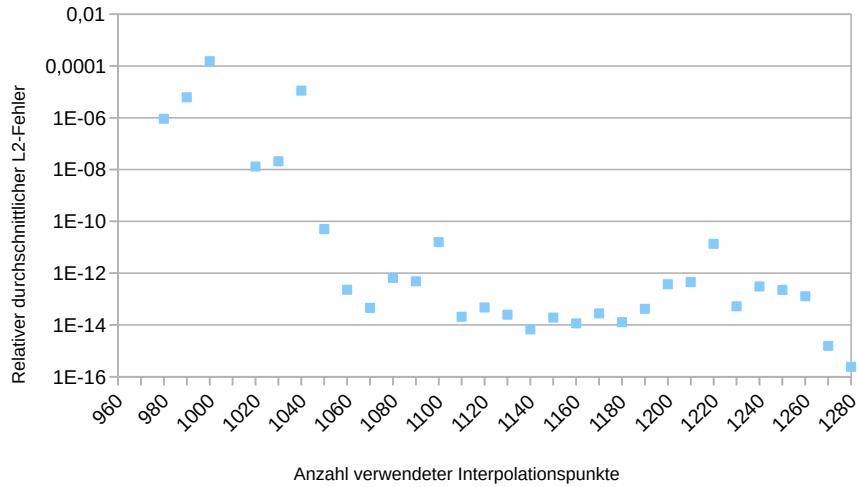


**Abbildung 3.13:** DEIM Fehler (relativ)

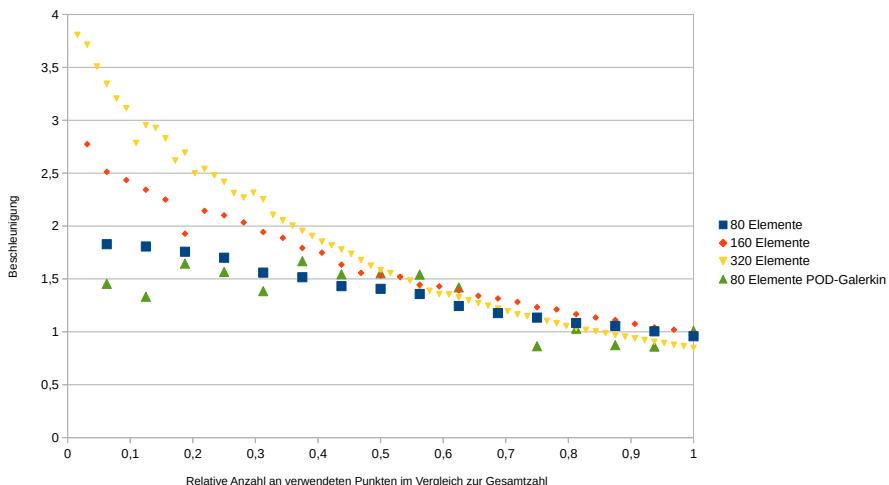
Modelle für die verschiedenen Eingabefunktionen simuliert. Abbildung 3.20 zeigt den Fehler dieser Berechnungen, Abbildung 3.19 die Beschleunigung für die jeweilige Dimension des reduzierten Modells. Es zeigt sich, dass die Fehlerwerte für die verschiedenen Eingaben relativ nahe zusammen liegen. Außerdem ist die Beschleunigung zwar nicht so stark wie bei der Berechnung der Modelle für einen einzelnen Eingabefall, wie zum Beispiel bei den Berechnungen für eine einzelne Spannungsspitze, aber es kann immer noch die doppelte Geschwindigkeit erreicht werden.

## 3.6 Vergleich der verschiedenen Diskretisierungen

Für die am Anfang des Kapitels besprochene erste Diskretisierungsvariante wurde festgestellt, dass es Probleme damit gibt, dass das Ergebnis des reduzierten Modells für bestimmte Dimensionen

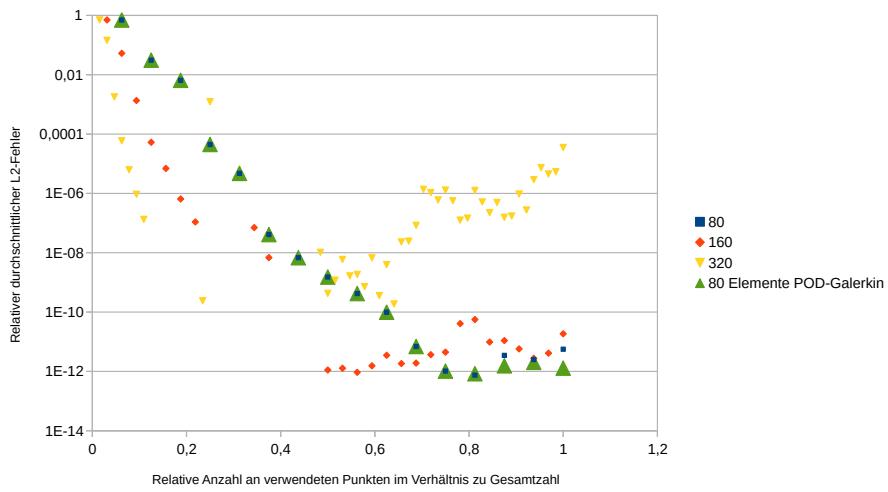


**Abbildung 3.14:** DEIM Fehler 320 Elemente (relativ)

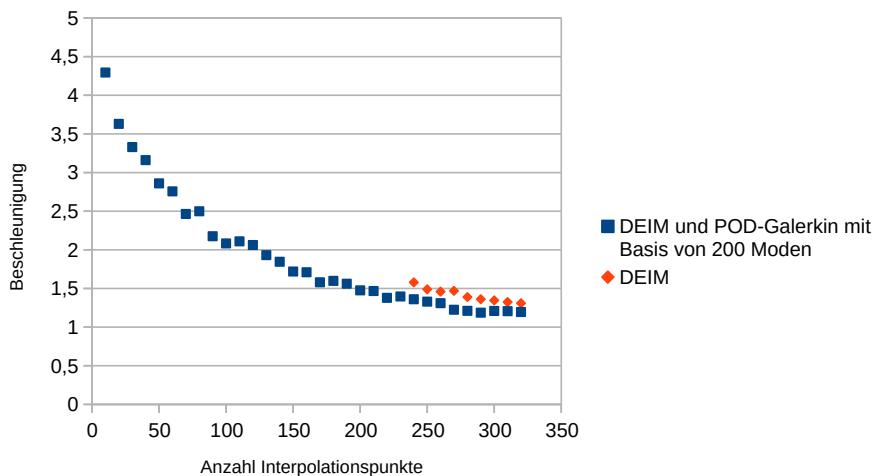


**Abbildung 3.15:** DEIM und POD-Galerkin, DEIM fest, Beschleunigung (relativ)

dieses Modells instabil wird. Deshalb wird nun die zweite Diskretisierungsvariante untersucht. Für dieselben Einstellungen wie in Abschnitt 3.4 wird nun die zweite Diskretisierungsvariante untersucht. In Abbildung 3.21 und Abbildung 3.22 ist zu sehen, dass die Beschleunigung der Berechnung mit sinkender Dimension des reduzierten Modells oder steigender Anzahl an diskreten Elementen nicht so sehr steigt wie in der ersten Diskretisierungsvariante. Das liegt aber daran, dass die Matrix A in diesem Fall anders als im ersten Fall die Form einer Bandmatrix hat, also ein Gleichungssystem mit dieser Matrix deutlich schneller gelöst werden kann als normale Matrizen. Wenn diese aber reduziert wird, dann muss die reduzierte Matrix mit einem höheren Rechenaufwand gelöst werden. Die Beschleunigung fällt also nicht so hoch aus, weil die Berechnung sowieso schon schneller ist. Der Berechnungsfehler nimmt dafür auch nicht so stark ab wie im ersten Fall – damit kann die Vermutung aus Abschnitt 3.3 bestätigt werden, dass die erste Diskretisierungsvariante genauere Ergebnisse liefert –, aber immer noch so schnell, dass genaue Ergebnisse in der Hälfte der Berechnungszeit des vollen Modells berechnet werden können. Dazu

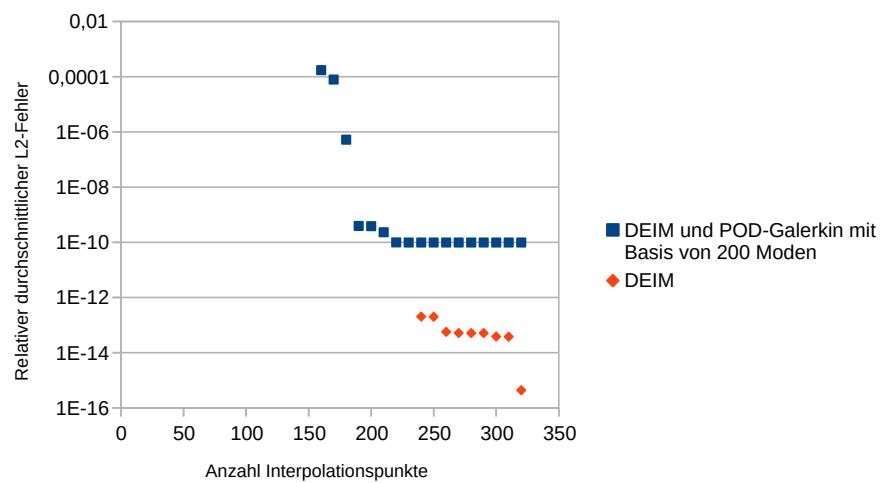


**Abbildung 3.16:** DEIM und POD-Galerkin, DEIM fest, Fehler (relativ)

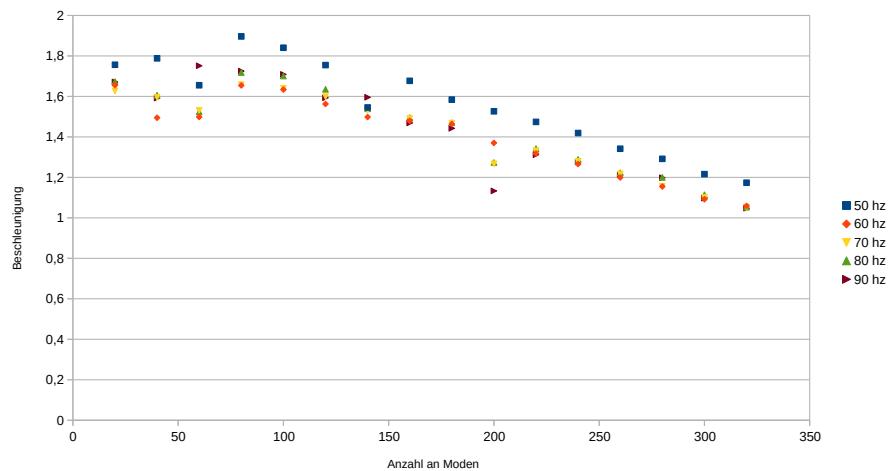


**Abbildung 3.17:** DEIM und POD-Galerkin, POD fest, Fehler (relativ)

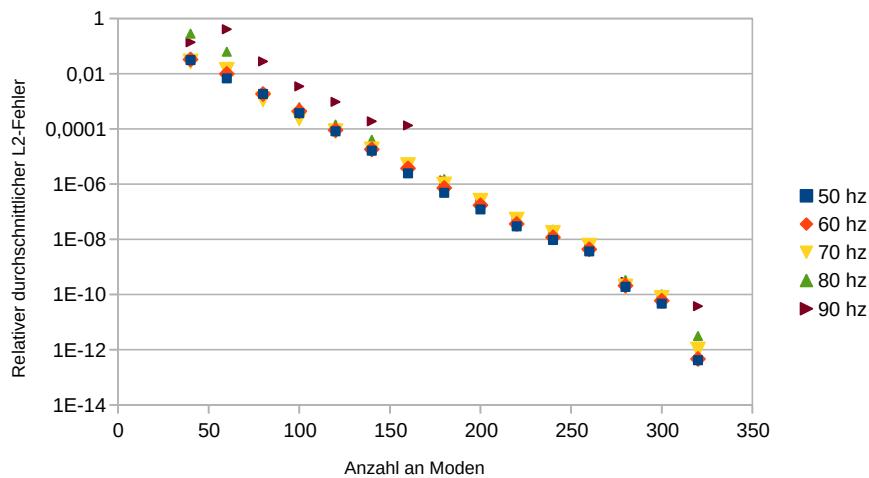
kommt noch, dass es bei der Berechnung mit POD-Galerkin in dieser Diskretisierungsvariante keine Bereiche gibt, in denen das reduzierte Modell instabil wird. Diese Ergebnisse sprechen dafür, im Anwendungsfall diese praktikablere zweite Variante zu verwenden.



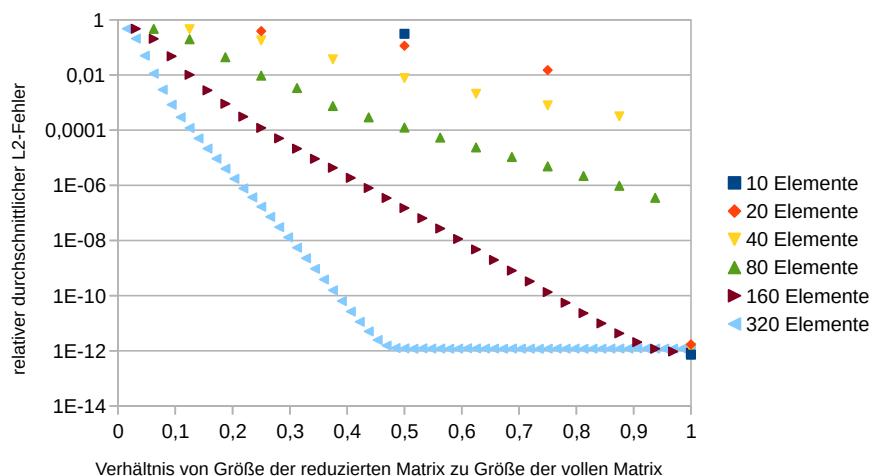
**Abbildung 3.18:** DEIM und POD-Galerkin, POD fest, Fehler (relativ)



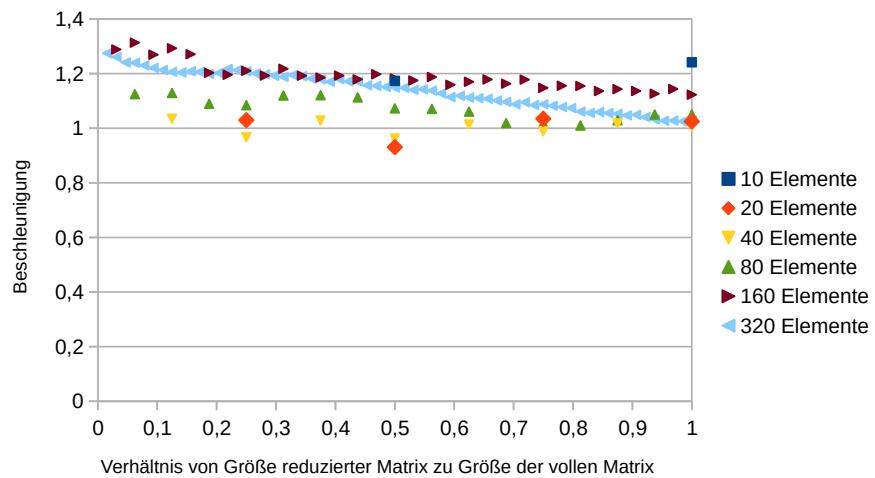
**Abbildung 3.19:** POD-Galerkin mit mehreren Input-Funktionen Beschleunigung



**Abbildung 3.20:** POD-Galerkin mit mehreren Input-Funktionen Fehler



**Abbildung 3.21:** POD-Galerkin mit V-Matrix, Fehler



**Abbildung 3.22:** POD-Galerkin mit V-Matrix, Fehler

## 4 Zusammenfassung und Ausblick

In der Arbeit wurden zwei verschiedene Diskretisierungsansätze zur Berechnung der Kombination des Hodgkin-Huxley-Modells und des Monodomain-Modells zur Propagation von Aktionspotentialen entlang von Muskelfasern beschrieben. Diese wurden für verschiedene Eingaben, Anzahl diskreter Elemente und Zeitschritte für die Model-Order-Reduction-Methoden POD-Galerkin und DEIM sowie die Kombination beider untersucht. Dabei ergab sich, dass beide das Potential haben, die Berechnung des Modells zu beschleunigen. Mit POD-Galerkin ist es außerdem möglich, die Moden für das Modell mit reduzierter System für mehrere Eingabefunktionen mit hoher Genauigkeit vorausberechnen.

Die Kombination von POD-Galerkin und Interpolation über bestimmte Punkte des nichtlinearen Teils des Modells ergab nicht die gewünschten Beschleunigungssteigerungen gegenüber der separaten Anwendung dieser Methoden.

Das Ergebnis des Vergleichs der beiden Diskretisierungsvarianten ist, dass die zweite Variante, in der nur das Ergebnis für die elektrische Spannung aus der nichtlinearen Funktion verwendet wird, besser ist, weil die Berechnung schneller möglich ist und das reduzierte System – anders als in der anderen Variante – für alle Dimensionen dieses Systems stabil ist.

### Ausblick

In der Arbeit wurden einige Fragen beantwortet, aber auch einige neue aufgeworfen. So könnte man untersuchen, weshalb die Kombination von DEIM und POD-Galerkin in KerMor langsamer war, als die einzelne Anwendung dieser Verfahren. Außerdem muss für den Anwendungsfall ein Verfahren entwickelt werden, mit dem festgelegt wird, welche Daten in der Offline-Phase berechnet werden oder wie effektiv entschieden werden kann, welche Dimension das reduzierte Modell haben soll (zum Beispiel mit Fehlerschätzern). In der Online-Phase ist die Frage, ob und wie die Moden, die zur Reduktion des Models verwendet werden, während der Berechnung geändert werden können, wenn sich zum Beispiel die Eingabe ändert. Außerdem kann für weitere Verfahren der Model Order Reduction getestet werden, ob diese für das fragliche Modell bessere Approximations- und Beschleunigungsergebnisse erzielen. Am verwendeten Framework KerMor kann auch weiter gearbeitet werden. Weitere Methoden der Model Order Reduction können implementiert werden, die Laufzeitberechnung kann vergleichbarer gestaltet werden.



# Literaturverzeichnis

- [14] „Medical gallery of Blausen Medical 2014“. In: *WikiJournal of Medicine* 1.2 (2014). doi: [10.15347/wjm/2014.010](https://doi.org/10.15347/wjm/2014.010) (zitiert auf S. 19).
- [CS10] S. Chaturantabut, D. C. Sorensen. „Nonlinear Model Reduction via Discrete Empirical Interpolation“. In: *SIAM Journal on Scientific Computing* 32.5 (Jan. 2010), S. 2737–2764. doi: [10.1137/090766498](https://doi.org/10.1137/090766498) (zitiert auf S. 24–26).
- [Hei] T. Heidlauf. *Chemo-electro-mechanical modelling of the neuromuscular system: Stuttgart, Universität Stuttgart, Diss.*, 2015. ISBN: 978-3-946412-00-7. URL: <http://nbn-resolving.de/urn:nbn:de:bsz:93-opus-104496> (zitiert auf S. 17, 18, 20–23).
- [Wir14] D. Wirtz. *Model Reduction for Nonlinear Systems: Kernel Methods and Error Estimation*. epubli GmbH, 2014. ISBN: 978-3-8442-7671-8. URL: <https://www.amazon.com/Model-Reduction-Nonlinear-Systems-Estimation/dp/3844276718?SubscriptionId=0JYN1NVW651KCA56C102&tag=techkie-20&linkCode=xm2&camp=2025&creative=165953&creativeASIN=3844276718> (zitiert auf S. 20, 24–26).

Alle URLs wurden zuletzt am 20. 12. 2017 geprüft.



### **Erklärung**

Ich versichere, diese Arbeit selbstständig verfasst zu haben. Ich habe keine anderen als die angegebenen Quellen benutzt und alle wörtlich oder sinngemäß aus anderen Werken übernommene Aussagen als solche gekennzeichnet. Weder diese Arbeit noch wesentliche Teile daraus waren bisher Gegenstand eines anderen Prüfungsverfahrens. Ich habe diese Arbeit bisher weder teilweise noch vollständig veröffentlicht. Das elektronische Exemplar stimmt mit allen eingereichten Exemplaren überein.

---

Ort, Datum, Unterschrift