

Fakultät für Physik und Astronomie
Ruprecht-Karls-Universität Heidelberg

Bachelorarbeit
im Studiengang Physik
vorgelegt von
Jonathan Born
geboren in Gießen, Deutschland

März 2012

Distanzmaße für Spiketrains im Rahmen der Optimierung von Neuronparametern

Diese Bachelorarbeit wurde von Jonathan Born ausgeführt am

KIRCHHOFF-INSTITUT FÜR PHYSIK

RUPRECHT-KARLS-UNIVERSITÄT HEIDELBERG

unter der Betreuung von

Prof. Dr. Karlheinz Meier

Distance measures between spike trains in the scope of neuron parameter fitting

In this thesis, the methodology of fitting an AdEx model neuron to biological data is investigated. A comparison of resulting spike trains via several distance measures is analyzed in detail. Based upon these measures, a particle swarm algorithm performs the final optimization in the space of AdEx model parameters. The focus lies upon the capability of the presented measures to guarantee a fast convergence of the optimization to a good set of neuron parameters. By comparing the performance of the various measures, it is possible to narrow down the choice among them, as well as to prepare and discuss future improvements, concerning both the computation of the measures themselves and further developments of the optimization algorithm.

Distanzmaße für Spiketrains im Rahmen der Optimierung von Neuronparametern

In dieser Arbeit werden Methoden zur Anpassung des Verhaltens von AdEx-Modellneuronen auf Daten aus biologischen Messungen untersucht. Zum Vergleich der resultierenden Spiketrains werden verschiedene Distanzmaße herangezogen und analysiert. Diese stellen die Grundlage für einen Partikelschwarm-Optimierungsalgorithmus im Raum der Parameter des AdEx-Modells. Insbesondere werden die vorgestellten Maße bezüglich ihrer Fähigkeit, eine schnelle Konvergenz der Optimierung zu einem guten Parametersatz zu gewährleisten, untersucht. Ein Vergleich der Leistungsfähigkeit der Maße ermöglicht schließlich eine Vorauswahl unter diesen zu treffen und erlaubt auch die Vorbereitung und Diskussion zukünftiger Verbesserungsmöglichkeiten - sowohl bezogen auf die Berechnung der Maße, als auch auf die weitere Entwicklung des Optimierungsalgorithmus.

Inhaltsverzeichnis

1	Einleitung	1
1.1	Motivation	1
1.2	Adaptive Exponential Integrate-and-Fire Modell	1
1.3	Single Neuron Modeling Competition	3
2	Methoden	4
2.1	Partikelschwarm-Optimierung	4
2.2	Spiketrain-Maße	5
2.2.1	Gamma-Maß	5
2.2.2	Exponential-Maß	6
2.2.3	Gauß-Maß	7
2.2.4	Reduziertes Gauß-Maß	8
2.2.5	Victor-Purpura-Maß	11
2.3	Normierung der Maße	12
2.3.1	Normierung des Gamma-Maßes	12
2.3.2	Normierung des Exponential-Maßes	13
2.3.3	Normierung des Gauß-Maßes	13
2.3.4	Normierung des reduzierten Gauß-Maßes	13
2.3.5	Victor-Purpura-Maß	13
2.4	Erzeugung von Spiketrains	14
2.4.1	Erzeugung von Spiketrains mit der Gammaverteilung	14
2.4.2	Verrauschen von Spiketrains	14
2.5	Analyse der Potentiallandschaften	15
3	Ergebnisse	17
3.1	Untersuchung der Spiketrain-Maße	17
3.1.1	Vergleich der Maße bei vergleichbaren Zeitparametern	17
3.1.2	Vergleich der Maße bei unterschiedlichen Zeitparametern	21
3.1.3	Laufzeitmessung der Berechnung der Maße	21
3.2	Analyse der Potentiallandschaften	23
4	Diskussion und Ausblick	28
	Literaturverzeichnis	33
	Danksagungen	35

1 Einleitung

1.1 Motivation

Im Rahmen des *FACETS*¹-Projektes und seines Nachfolgers *BrainScaleS* wird eine neuartige, sogenannte *neuromorphe Hardware* entwickelt, welche hochgradig beschleunigte Emulationen neuronaler Netze bei geringem Energieverbrauch ermöglichen soll. Um Messungen an biologischen Neuronen reproduzieren zu können, ist es zunächst notwendig, die auf neuromorpher Hardware emulierten Neuronen an diese Messdaten anzupassen.

Zu diesem Zweck wird in der vorliegenden Arbeit ein auf dem Partikelschwarm-Optimierungsverfahren basierender Optimierungsalgorithmus (siehe Abschnitt 2.1) benutzt, der sowohl für Software-Simulationen als auch Hardware-Emulationen geeignet ist. In dieser Arbeit werden Neuronen mit dem Software-Simulator NEST (Gewaltig & Diesmann, 2007) simuliert, um den Algorithmus ohne Einsatz der Hardware verbessern zu können. Zur Beschreibung neuronaler Netze (hier ein Neuron) wird die Beschreibungssprache PyNN benutzt (Davison et al., 2008), welche sich auch zur Ansteuerung der Hardware eignet. Das langfristige Ziel besteht darin, die Neuronen der neuromorphen Hardware an Messdaten anzupassen.

In Pfeil (2011) wurde bereits die Dynamik von mit Strom angeregten Neuronen erfolgreich reproduziert, hier soll die Anregung durch Aktionspotentiale (im Folgenden auch Spikes genannt) geschehen. Dies ist jedoch schwieriger zu optimieren, da die Anzahl der freien Parameter (insbesondere synaptische Parameter) größer ist. Die Effizienz des Optimierungsalgorithmus sowie die Entscheidung, ob und welcher Optimierungsalgorithmus überhaupt angewandt werden kann bzw. sollte, ist erwartungsgemäß stark davon abhängig, welches Maß zum Vergleich der Referenzdaten (z.B. biologische Messungen) mit den emulierten Daten verwendet wird.

Daher werden in dieser Arbeit verschiedene Maße (siehe Abschnitt 2.2) eingeführt, die dann sowohl in Bezug auf ihr Verhalten im Optimierungsalgorithmus als auch in Bezug auf ihre Komplexität und Effektivität verglichen und bewertet werden.

Das Ziel dieser Arbeit ist es, herauszufinden, welches Maß sich zum einen hinsichtlich des PSO-Verfahrens und zum anderen für Optimierungsverfahren generell, am besten eignet.

1.2 Adaptive Exponential Integrate-and-Fire Modell

In meiner Untersuchung benutze ich das *Adaptive Exponential Integrate-and-Fire Modell* (Brette & Gerstner, 2005), kurz AdEx, da dieses das auf der BrainScaleS-Hardware im-

¹Fast Analog Computing with Emergent Transient States

1 Einleitung

plementierte Neuronenmodell repräsentiert. Es zeichnet sich dadurch aus, dass es eine Vielzahl von Mustern neuronaler Dynamik zufriedenstellend reproduzieren kann (Naud et al., 2008).

Das AdEx-Modell lässt sich durch die folgende Differentialgleichungen beschreiben:

$$C \frac{dV}{dt} = -g_L(V - E_L) + g_L \Delta_T \exp\left(\frac{V - V_T}{\Delta_T}\right) - g_e(t)(V - E_e) - g_i(t)(V - E_i) - w \quad (1.1)$$

$$\tau_w \frac{dw}{dt} = a(V - E_L) - w \quad (1.2)$$

$$\text{Falls } V = V_{Spike} : \quad V \rightarrow E_R, \quad w \rightarrow w + b \quad (1.3)$$

Dabei beschreibt Gleichung 1.1 die Dynamik des Membranpotentials V , welche mit der Gleichung 1.2 gekoppelt ist, die Adaption des Neurons.

Im Ruhezustand kann das Neuron als einfaches RC -Glied betrachtet werden, wobei die Membrankapazität C über einen Widerstand (hier die Leckkonduktanz $g_L = \frac{1}{R}$) mit dem Ruhepotential E_L verbunden ist. Ein synaptischer Stimulus (d.h. ein afferenter Spike) führt dazu, dass eine zusätzliche, zeitabhängige Konduktanz g_x zu einem entsprechenden Umkehrpotential E_x geöffnet wird (dabei steht x für den Typ der Synapse, d.h. e für exzitatorisch und i für inhibitorisch). Typischerweise liegt E_e bei 0 bis 20 mV und E_i bei -90 bis -80 mV. Die Konduktanzen g_x bewegen sich dabei im nS-Bereich.

Wenn die Membran stark genug angeregt wird, um V_T zu erreichen oder sogar zu überschreiten, beginnt der Exponentialterm in Gleichung 1.1 die Membrandynamik zu dominieren: das Membranpotential fängt an, exponentiell zu wachsen. Die Geschwindigkeit dieses Wachstums wird von dem Parameter Δ_T bestimmt. Der Exponentialterm soll den Mechanismus biologischer Neuronen reproduzieren, welcher durch Öffnung von Natriumkanälen ein Spike einleitet. Wenn das Membranpotential den endlichen Grenzwert V_{Spike} erreicht (siehe Gleichung 1.3), feuert das Neuron ein Spike ab. Unmittelbar danach wird V auf das Resetpotential E_R zurückgesetzt.

Zusätzlich zum synaptischen Input wird das Membranpotential auch von der Adaptionsvariable w beeinflusst, welche die Maßeinheit eines Stroms besitzt. Im Ruhezustand beträgt sie 0, ändert sich aber aufgrund von zwei Mechanismen. Zum einen ist sie ihrerseits abhängig vom Membranpotential; diese Abhängigkeit wird von der Konstante a gesteuert. Des weiteren bewirkt jedes Feuern des Neurons eine Veränderung von w um den Wert b . Der zeitliche Verlauf von w wird von der Zeitkonstante τ_w bestimmt. Oftmals werden sowohl für a als auch für b positive Werte verwendet, was zur Folge hat, dass w anwächst, wenn das Neuron angeregt wird bzw. feuert. Da w negativ in die Differentialgleichung für das Membranpotential eingeht, bewirkt es dann eine "Abschwächung" des Neurons, was beispielsweise dazu führen kann, dass es seltener feuert oder schwieriger anzuregen ist.

1.3 Single Neuron Modeling Competition

Die International Neuroinformatics Coordination Facility (INCF) veranstaltet unter anderem die *Single Neuron Modeling Competition* (Jolivet et al., 2008), im Rahmen derer verschiedene Neuronenmodelle bezüglich ihrer Fähigkeit, die Dynamik von biologischen Neuronen zu reproduzieren, untersucht werden. Ein solcher Wettbewerb beinhaltet mehrere Aufgaben, unter anderem eine Aufgabe die auch für diese Arbeit von Interesse ist: die Analyse prä- und postsynaptischer Spiketrains (Abfolge von Aktionspotentialen), um daraus das zukünftige Verhalten des postsynaptischen Spiketrains unter Kenntnis des präsynaptischen Spiketrains vorherzusagen zu können. Zu diesem Zweck wurden extrazelluläre Messungen einzelner Neuronen an lebenden Rhesusaffen durchgeführt. Diese Messungen werden in Carandini et al. (2007) und Sincich et al. (2007) beschrieben.

Der visuelle Stimulus, der den Spiketrains zugrunde liegt, wurde 76 mal wiederholt, so dass jeweils 76 prä- und postsynaptische Spiketrains vorliegen, die sich bezüglich ihrer Feuerrate und Feuermuster nur wenig voneinander unterscheiden. Diese Spiketrains werden im Abschnitt 3.1.2 dazu benutzt, um das Verhalten der Spiketrain-Maße (siehe Abschnitt 2.2) bei variierenden Zeitparametern zu untersuchen.

2 Methoden

2.1 Partikelschwarm-Optimierung

Die Partikelschwarm-Optimierung (PSO) ist ein Optimierungsverfahren, bei dem eine Population von Partikeln \vec{x} , also Parametersätzen, die sich im Parameterraum der möglichen Lösungen befinden, iterativ optimiert wird (Rossant et al., 2010). In diesem Fall besteht der Parameterraum aus den Parametern des AdEx-Modells (siehe Abschnitt 1.2).

Zu Beginn kann sich jedes Partikel an einem willkürlich gewählten Ort befinden, allerdings sollten die Partikel weit genug über das Gebiet verteilt sein, in dem die Lösung vermutet wird. Dies wird hier ermöglicht, indem sich die Partikel in gleichmäßigen Abständen gitterförmig über den Parameterraum verteilen. Um die Parameter bewerten zu können, werden Maße eingeführt. Mithilfe des jeweiligen Maßes kann eine "Potentiallandschaft" erstellt werden, indem man im Parameterraum das Ergebnis des Maßes für jeden Parametersatz \vec{x} aufträgt. Genau genommen erfüllen diese Landschaften der verschiedenen Maße nicht die Eigenschaften eines Skalarpotentials, werden aber aufgrund der Ähnlichkeit zu anderen Optimierungsproblemen Potentiallandschaften genannt. Diese Potentiallandschaft besitzt dann am Ort des optimalen Parametersatzes ein Maximum. Die PSO dient zur Auffindung des Parametersatzes, welcher diese Lösung liefert.

Diese Optimierung verläuft in mehreren Schritten, die sich in jeder Iteration wiederholen, bis eine Abbruchbedingung erreicht ist. Die Abbruchbedingung kann schlicht die Anzahl der Iterationen oder ein bestimmter Wert des Maßes sein. In jeder Iteration werden folgende Schritte der Reihe nach ausgeführt:

1. Simulation (oder im Fall der Hardware Emulation) aller Parametersätze \vec{x} , die den Positionen der Partikel entsprechen
2. Berechnung des Maßes durch Vergleich der durch Simulation entstandenen Spiketrains mit dem Referenz-Spiketrain
3. Bestimmung der besten bisherigen Position jedes einzelnen Partikels \vec{x}_{lb} , beruhend auf dem größten Wert des Maßes aus Schritt 2, sowie der absolut besten Position aller Partikel $\vec{x}_{gb} = \max(\vec{x}_{lb})$
4. Aktualisierung der Geschwindigkeit der Partikel (Die Anfangsgeschwindigkeit der Partikel beträgt 0):

$$\vec{v}_{i+1} = \Omega \vec{v}_i + c_l r_l (\vec{x}_{lb} - \vec{x}_i) + c_g r_g (\vec{x}_{gb} - \vec{x}_i) \quad (2.1)$$

5. Bestimmung der neuen Position jedes Partikels:

$$\vec{x}_{i+1} = \vec{x}_i + \vec{v}_{i+1} \quad (2.2)$$

Anzumerken ist, dass jedes Partikel ein ‘‘Gedächtnis’’ besitzt, es merkt sich die beste Position \vec{x}_{lb} an dem es selbst war, sowie die beste, bislang bekannte Position \vec{x}_{gb} im Parameterraum. Je nach Lage des Partikels im Parameterraum bewegt es sich mehr oder weniger schnell in Richtung dieser Positionen (siehe Gleichung 2.1). Der Einfluss der jeweiligen Positionen auf die Bewegung wird durch die Gewichtungsfaktoren $c_g = 1.5$ bzw. $c_l = 0.1$ bestimmt. Um die Chance zu verringern, dass die Partikel zu einem lokalen anstatt einem globalen Maximum konvergieren, wird der Parameterraum durch die Partikel folgendermaßen erkundet. Zum einen hat die Geschwindigkeit \vec{v} der vorherigen Iteration einen Einfluss (die Größe dieses Einflusses wird durch den Vorfaktor $\Omega = 0.9$ bestimmt) auf die weitere Bewegung des Partikels (siehe Gleichung 2.1 und Gleichung 2.2). Durch diesen Mechanismus können die Partikel sich über ein Maximum hinweg bewegen. Zum anderen gibt es zwei Zufallsfaktoren, die diese Geschwindigkeit beeinflussen (r_l und r_g in Gleichung 2.1). Diese Zufallsfaktoren können dabei Werte im Intervall $[0, 1]$ annehmen.

2.2 Spiketrain-Maße

Um die Güte eines Parametersatzes bestimmen zu können, muss der mithilfe des Parametersatzes erzeugte Mess-Spiketrain t_M mit dem Referenz-Spiketrain t_R verglichen werden. Dabei ist der Referenz-Spiketrain der (z.B. durch biologische Messungen) vorgegebene Spiketrain, dessen erzeugende Neuronparameter ermittelt werden sollen. In diesem Abschnitt wird eine Auswahl an Maßen anhand eines einfachen Beispiels vorgestellt. Dazu werden stets die gleichen Spiketrain benutzt. Zudem werden alle Maße auf den Bereich $[0, 1]$ normiert, wobei 1 bedeutet, dass die Spiketrain identisch sind, und 0, dass keine Spikes im Rahmen des Maßes übereinstimmen. Die exakte Normierung bzw. Ausnahmen bei der Normierung werden im Teil Normierungen (Abschnitt 2.3) besprochen.

2.2.1 Gamma-Maß

Das Gamma-Maß ist ein Koinzidenzmaß, welches ursprünglich Spiketrain zweier verschiedener Neuronenmodelle vergleicht und in Kistler et al. (1997) eingeführt wird. Dieses Maß berechnet sich mittels der Gleichung

$$\Gamma = \frac{N_{\text{coin}} - \langle N_{\text{coin}} \rangle}{\frac{1}{2} (N_{M_1} + N_{M_2})} \mathcal{N}^{-1} \quad (2.3)$$

Der Wert Γ berechnet sich aus der Summe N_{coin} der Fälle, in denen sich innerhalb des Zeitfensters $\pm \Delta T$ eines Spikes des Referenz-Spiketrain ein Spike des Mess-Spiketrain befindet. Falls sich innerhalb dieses Zeitfensters mehrere Spikes des Mess-Spiketrain befinden, erhöht sich N_{coin} und damit natürlich auch Γ nicht (z.B. im Fall des ersten Spikes des Referenz-Spiketrain in Abbildung 3.8). Dasselbe gilt auch für den Fall, in

2 Methoden

dem sich ein Spike des Mess-Spiketrains im Zeitfenster mehrerer Spikes des Referenz-Spiketrains befinden (z.B. im Fall des zweiten und dritten Spikes des Referenz-Spiketrains in Abbildung 3.8). Dabei ist der Normierungsfaktor gegeben durch $\mathcal{N} = 1 - \frac{N_{M_1}}{K}$. Der Erwartungswert von N_{coin} ist $\langle N_{\text{coin}} \rangle = \frac{N_{M_1} N_{M_2}}{K}$, wobei K gegeben ist durch $K = \frac{T}{2\Delta T}$ (für die Motivation dieser Wahl siehe Kistler et al. (1997)). N_{M_1} und N_{M_2} sind in diesem und den nachfolgenden Abschnitten die Anzahl der Spikes des Referenz- bzw. Mess-Spiketrains und T steht für die Simulationsdauer.

Dieses Maß kann als einziges negativ werden, nämlich im Fall negativer Korrelation zwischen den Spiketrains.

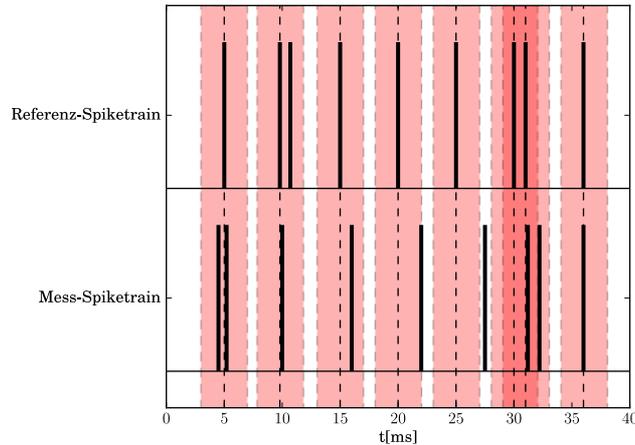


Abbildung 2.1: Erzeugung des Γ -Wertes. Wenn sich innerhalb des gewählten Zeitfensters $\pm\Delta T$ um einen Spike des Referenz-Spiketrains ein Spike des Mess-Spiketrains befindet, wird N_{coin} um 1 erhöht. Jeder Spike kann maximal einmal in die Wertung eingehen. In diesem Fall ist $N_{\text{coin}} = 7$. Die Dauer der für jeden Spike im Referenz-Spiketrain in rot eingezeichneten Zeitfenster $2\Delta T$ beträgt in diesem Fall 4 ms.

2.2.2 Exponential-Maß

Das Exponential-Maß aus (van Rossum, 2001) repräsentiert jeden Spike mittels einer Exponentialfunktion. Die Funktion $f(t)$ erzeugt aus Spiketrains eine Superposition von abfallenden Exponentialfunktionen. Jeder Spike (mit Spikezeitpunkt t_i) des Spiketrains entspricht dabei einer Exponentialfunktion mit Zeitkonstante t_c ,

$$f(t) = \sum_{i=1}^N H(t - t_i) \exp(-(t - t_i)/t_c) \quad , \quad (2.4)$$

wobei H die Heaviside-Funktion ist und N die Anzahl der Spikes des Spiketrains.

Die Distanz x_E zwischen zwei Spiketrains f und g berechnet sich dann mittels der L^2 -Norm der Differenz der beiden Funktionen:

$$x_E = \frac{1}{t_c} \int_0^\infty |f_R(t) - f_M(t)|^2 dt \quad (2.5)$$

Dabei wird $f_R(t)$ aus dem Referenz-Spiketrain und $f_M(t)$ aus dem Mess-Spiketrain erstellt.

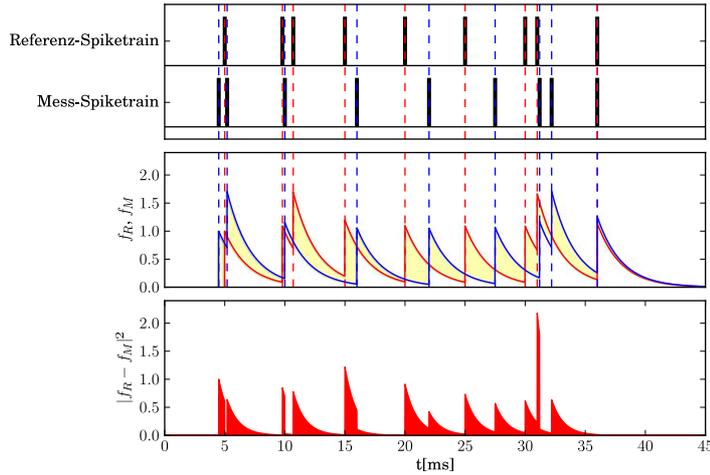


Abbildung 2.2: Die Berechnung der Exponential-Maßes. Oben: Die Beispiel-Spiketrains. Mitte: Die Differenz der aus beiden Spiketrains erstellten Funktionen (rot für den Referenz-Spiketrain und blau für den Mess-Spiketrain) ist gelb eingezeichnet. Unten: Differenz der beiden Funktionen, deren L^2 -Norm das Ergebnis des Maßes darstellt. Die Zeitkonstante t_c beträgt 2 ms.

2.2.3 Gauß-Maß

Beim Gauß-Maß wird jeder Spike durch eine Gaußfunktion (siehe Gleichung 2.7) repräsentiert.

Um die Berechnung des Maßes zu beschleunigen, werden die Gaußfunktionen mit $t_i \pm T_g$ begrenzt unter der Bedingung, dass $f(t_i \pm T_g) \approx 0$ gilt. Damit diese Bedingung erfüllt ist, wird T_g in Abhängigkeit von ρ gewählt:

$$T_g = 8 \cdot \rho \quad (2.6)$$

Somit kann man die abgeschnittenen Teile der Gaußfunktionen bei der Berechnung von x_G vernachlässigen. Dabei ist ρ der Zeitparameter der Gaußfunktion, der der Breite

2 Methoden

der folgenden Gaußfunktion entspricht.

$$g_i(t) = \begin{cases} \frac{1}{\sqrt{\rho} \cdot \sqrt{\pi}} \cdot \exp\left(-\frac{1}{2} \cdot \left(\frac{t-t_i}{\rho}\right)^2\right) & , \text{ falls } |t - t_i| < T_g \\ 0 & , \text{ sonst} \end{cases} \quad (2.7)$$

Die Gaußfunktion $g(t)$ ist so normiert, dass für ihre L^2 -Norm gilt: $\int_{-\infty}^{+\infty} g(t)^2 dt = 1$. Es folgt für die Berechnung des Gaußwertes x_G :

$$g_R(t) = \sum_i g_i(t), \text{ mit } t_i \in t_R \quad (2.8)$$

$$g_M(t) = \sum_i g_i(t), \text{ mit } t_i \in t_M \quad (2.9)$$

$$x_G = \int_{-\infty}^{\infty} g_R(t) \cdot g_M(t) dt \quad (2.10)$$

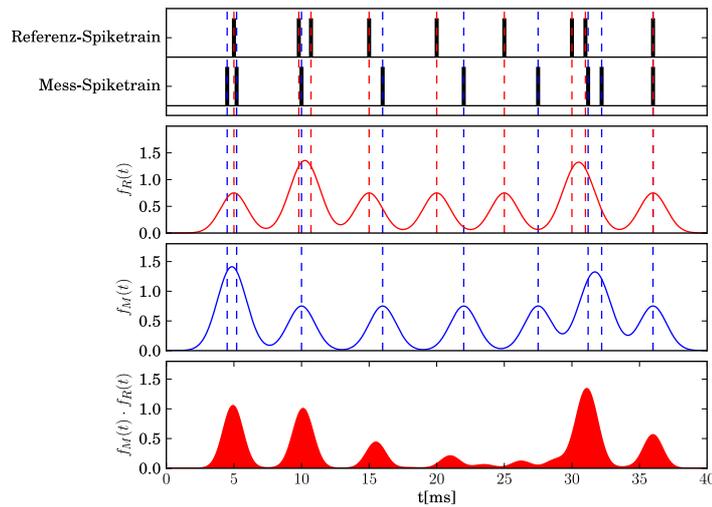


Abbildung 2.3: Erzeugung des Gauß-Wertes x_G . Aus Referenz-Spiketrain und Mess-Spiketrain werden zunächst $g_R(t)$ und $g_M(t)$ erstellt. ρ beträgt in diesem Fall 1 ms und folglich ist $T_g = 8$ ms (siehe Gleichung 2.6). In der untersten Abbildung sieht man die Funktion $g_R(t) \cdot g_M(t)$, deren Integral über die Zeit t zum Endergebnis x_G des Maßes führt.

2.2.4 Reduziertes Gauß-Maß

In diesem Maß werden Spikes ebenfalls mit Gaußfunktionen gefaltet. Diese werden paarweise miteinander multipliziert und der Wert des Maßes ergibt sich aus dem Integral

über das Ergebnis dieser Multiplikation. Jeder Spike kann dabei nur einmal gepaart werden. Dies umgeht das Problem des Gauß-Maßes, dessen Wert sich erhöht, wenn sich viele Spikes des Mess-Spiketrains in der Nähe nur eines Spikes des Referenz-Spiketrains befinden. Das Ergebnis x_G erhöht sich also stark (und suggeriert eine hohe Übereinstimmung) obwohl der Mess-Spiketrain an dieser Stelle zu viele Spikes besitzt und sich somit x_G nicht erhöhen sollte. Gleichzeitig lässt sich dieses Maß deutlich schneller berechnen als das Gauß-Maß (siehe Abschnitt 3.1.3).

Die Gaußfunktionen, die die Spikes repräsentieren, berechnen sich wie in Gleichung 2.7 des Gauß-Maßes. Die so erstellten Funktionen werden dann benutzt, um den reduzierten Gaußwert x_{RG} zu bestimmen:

$$g_R^i(t) = g_i(t), \text{ mit } t_i \in t_R \quad (2.11)$$

$$g_M^j(t) = g_j(t), \text{ mit } t_j \in t_M \quad (2.12)$$

$$x_{RG} = \sum_i \sum_j \int_{-\infty}^{\infty} g_R^i(t) \cdot g_M^j(t) dt, \quad \text{falls } |t_i - t_j| < t_L \quad (2.13)$$

Dabei bestimmt t_L den Bereich (der Standardwert für t_L beträgt 3 ms), innerhalb dessen Spikes gepaart werden können. Jeder Spike kann dabei nur einmal gepaart werden.

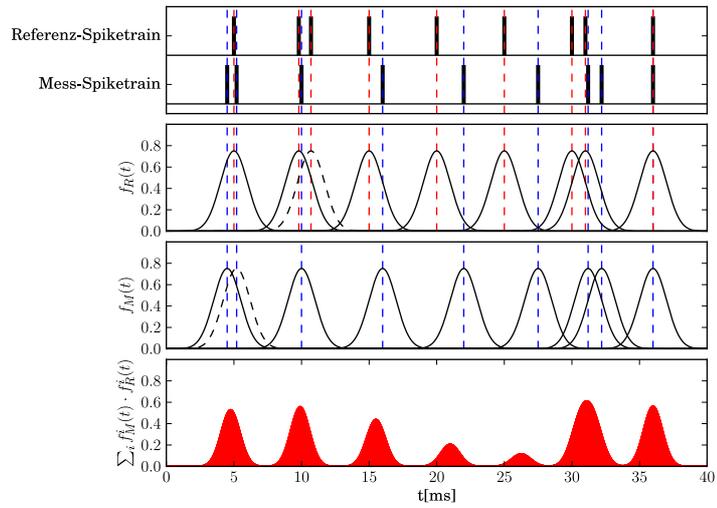


Abbildung 2.4: Erzeugung des reduzierten Gauß-Wertes x_{RG} . Aus Referenz- und Mess-Spiketrain werden zunächst $g_R^i(t)$ und $g_M^i(t)$ erstellt. Im untersten Abschnitt werden die Funktionen dann Paarweise multipliziert; das Integral über diese Funktion liefert den reduzierten Gaußwert. Die gestrichelten Spikes werden von dem Maß nicht beachtet, da sich kein noch nicht benutzter Spike des jeweils anderen Spiketrains in der Nähe befindet.

2.2.5 Victor-Purpura-Maß

Das von Victor & Purpura (1996) eingeführte Maß vergleicht einen Mess-Spiketrain t_M mit einem Referenz-Spiketrain t_R , indem es den Mess-Spiketrain mittels von den Autoren definierten elementaren Umformungsschritten in den Referenz-Spiketrain überführt. Dabei sind jeweils das Erstellen, Löschen und Verschieben von Spikes mit Kosten verbunden. Es bestehen mehrere Möglichkeiten, die Kosten C bei der Verschiebung eines Spikes zu gewichten. In der vorliegenden Arbeit gibt es zwei verschiedene Arten, C zu berechnen. Einmal steigt C linear (siehe Gleichung 2.14) mit dem zeitlichen Abstand zwischen zwei Spikes und einmal exponentiell (siehe Gleichung 2.15). Die maximalen Kosten C_{\max} einer Verschiebung können einen Wert von 2 nicht überschreiten, da die Kosten für das Löschen und Erstellen eines Spikes jeweils 1 betragen und damit die Option, einen Spike zu löschen und an anderer Stelle einzufügen, günstiger wird. Falls zwei Spikes jeweils genau übereinstimmen, betragen die Kosten 0.

- Lineare Kostenfunktion: Verschiebung eines Spikes kostet

$$C_{lin} = q \cdot |t_M - t_R| \quad (2.14)$$

- Exponentielle Kostenfunktion: Verschiebung eines Spikes kostet

$$C_{exp} = \exp\left(\frac{|t_M - t_R|}{t_c}\right) - 1 \quad (2.15)$$

q und t_c sind die jeweiligen Zeitparameter.

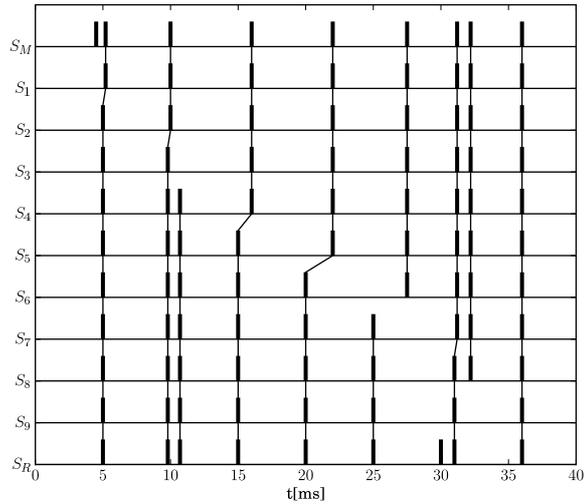


Abbildung 2.5: Erzeugung des Wertes x_{VP} des Victor-Purpura-Maßes. Hier sieht man, wie der Mess-Spiketrain schrittweise mittels der elementaren Umformungsschritte in den Referenz-Spiketrain überführt wird. Im Schritt zwischen S_6 und S_7 sieht man den Fall, in dem die Kosten für die Verschiebung C_{\max} überschreiten würden und der Spike daher gelöscht und wieder eingefügt wird. Diese Abbildung wurde mit linearem C und $q = 0.5$ erstellt.

2.3 Normierung der Maße

Um einen Vergleich zwischen den Maßen zu ermöglichen, werden die Maße normiert. Jedes Maß sollte einen Wert von 1 erreichen, wenn die zugrunde liegenden Spiketrains identisch sind und einen Wert von 0 falls die Spiketrains nicht übereinstimmen, z.B. wenn ein Spiketrain überhaupt keine Spikes aufweist. Manche Maße können davon abweichen, dies wird dann in der jeweiligen genaueren Beschreibung der Normierung des jeweiligen Maßes angemerkt.

2.3.1 Normierung des Gamma-Maßes

Das Gamma-Maß ist nach Kistler et al. (1997) auf $[-1, 1]$ normiert. $\Gamma = -1$ steht für eine maximal negative Korrelation zwischen den Spiketrains, $\Gamma = 0$ für einen Mess-Spiketrain, dessen Statistik der eines Poisson-Prozesses entspricht und $\Gamma = 1$ für identische Spiketrains. Negative Werte werden im Folgenden nicht betrachtet, ein Mess-Spiketrain ohne Spikes würde wie gewünscht dem Wert $\Gamma = 0$ entsprechen.

2.3.2 Normierung des Exponential-Maßes

Das Exponential-Maß, welches in Abschnitt 2.2.2 beschrieben wird, wird normiert, indem man es durch die Anzahl der Spikes beider Spiketrains teilt:

$$x'_E = 1 - \frac{x_E}{n_r + n_m} \quad (2.16)$$

Die Kosten für das Einfügen als auch für das Entfernen eines alleinstehenden Spikes betragen $\frac{1}{2}$ (siehe (van Rossum, 2001)). Um zwei Spikes aneinander anzugleichen, betragen also die maximalen Kosten 1. Das Verschieben bzw. Löschen überlagerter Spikes ist deutlich komplexer (siehe Gleichung 2.4 und Abbildung 2.2). Liegen die einzelnen Spikes jedoch weit genug voneinander entfernt, ist die hier gemachte Annahme sinnvoll, da die Beeinflussung durch andere Spikes in diesem Fall minimal ist. Bei identischem Referenz- und Mess-Spiketrains erhält man also wie gefordert das Ergebnis von 1, da x_E in diesem Fall 0 beträgt.

2.3.3 Normierung des Gauß-Maßes

Das Gauß-Maß ist unter der Annahme normiert, dass die zeitlichen Abstände zwischen aufeinanderfolgenden Spikes innerhalb eines Spiketrains hinreichend groß sind. Dies hat den Nachteil, dass dieses Maß im Fall kleiner Abstände größer als 1 werden kann. Der in Abschnitt 2.2.3 erhaltene Wert wird normiert, indem er durch die Anzahl der Spikes geteilt wird.

$$x'_G = \frac{2x_G}{n_r + n_m} \quad (2.17)$$

Die Gaußfunktionen in Gleichung 2.7 sind so normiert, dass das Ergebnis für x_G genau 1 beträgt, wenn zwei Spikes den selben Spikezeitpunkt haben und nicht durch benachbarte Spikes beeinflusst werden.

2.3.4 Normierung des reduzierten Gauß-Maßes

Das reduzierte Gauß-Maß wurde erstellt, um die Normierungsprobleme bei kleinen Spikeabständen zu beheben. Der Vorteil ist, dass benachbarte Spikes keinen Einfluss auf die Berechnung der Distanz zweier Spikes haben, so dass sich die möglichen Werte für x'_{RG} auf den Bereich $[0, 1]$ begrenzen.

$$x'_{RG} = \frac{2x_{RG}}{n_r + n_m} \quad (2.18)$$

2.3.5 Victor-Purpura-Maß

Die Kosten für das ineinander Überführen zweier Spiketrains können maximal $n_r + n_m$ betragen. Dies entspricht den Kosten für das Entfernen und Einfügen aller Spikes. Daher

2 Methoden

wird dieses Maß wie folgt normiert:

$$x'_{VP} = 1 - \frac{x_{VP}}{n_r + n_m} \quad (2.19)$$

2.4 Erzeugung von Spiketrains

Um die Maße vergleichen zu können, wird ein Referenz-Spiketrain mit dem Gamma-Prozess erzeugt. Der Mess-Spiketrain ergibt sich dann durch das kontrollierte “Verrauschen” der einzelnen Spikes des Referenz-Spiketrains.

2.4.1 Erzeugung von Spiketrains mit der Gammaverteilung

Die Verteilung der zeitlichen Abstände zwischen aufeinanderfolgenden Spikes ähnelt einer Gammaverteilung. Der Unterschied der Gammaverteilung gegenüber beispielsweise der Poissonverteilung besteht darin, dass die Intervalle zwischen zwei Spikes bei biologischen Messungen, aufgrund der Existenz einer endlichen Refraktärzeit, einen minimalen Abstand haben.

Die Gammaverteilung ist definiert durch ihre Wahrscheinlichkeitsdichte $p(x)$

$$p(x) = x^{k-1} \frac{\exp -x/\Theta}{\Theta^k \Gamma(k)} \quad (2.20)$$

mit dem Formparameter k , dem Skalierungsparameter Θ und der Gammafunktion $\Gamma(k)$, die definiert ist als:

$$\Gamma(k) = \int_0^\infty t^{k-1} \exp -t \, dt \quad (2.21)$$

Die Gammaverteilung kann benutzt werden, um Spiketrains zu erstellen, indem der Abstand x zweier aufeinanderfolgender Spikes entsprechend der Wahrscheinlichkeitsdichte $p(x)$ gewählt wird.

2.4.2 Verrauschen von Spiketrains

Die Motivation dafür, Spiketrains in dieser Art und Weise zu erzeugen, besteht darin, dass in Abschnitt 3.1.1 Mess-Spiketrains benötigt werden, die das ganze Spektrum von hohen und niedrigen Werten aller Maße erzeugen können. Spiketrains, die von einem gegebenen Referenz-Spiketrain in einstellbarer Stärke abweichen, können zum Beispiel erstellt werden, indem die einzelnen Spikes des Referenz-Spiketrains gaußverteilt verschoben werden. Die Verschiebung der einzelnen Spikes kann durch die Varianz der Gaußverteilung, hier als s bezeichnet, variiert werden. Dies sorgt dafür, dass die Verschiebung der Spikes bei kleinem s im Mittel klein sind sowie bei großem s im Mittel groß werden.

Die Parameter werden für die Messungen in Abschnitt 3.1.1 wie folgt gewählt: Die Parameter der Gammaverteilung betragen $\Theta = 100$ und $k = 2$, um einen Spiketrain mit einer Feuerrate von ca. 5 Hz zu simulieren. Diese Werte wurden gewählt, da der Erwartungswert der Gammaverteilung bei $\Theta \cdot k$ liegt. Unter Benutzung dieser Parameter

wird eine Anzahl von ca. 100 Spikes in einem Zeitraum von 20s erzeugt. Die Werte für die Varianz s der Gaußverteilung liegen im Intervall $[10^{-3} \text{ ms}, 10^7 \text{ ms}]$. Dabei wurden die einzelnen Werte $s(k, i) = k \cdot 10^i$ durch Wahl aller möglichen Kombinationen von $i \in \{-3, -2, \dots, 6\}$ und $k \in \{1, 2, \dots, 9\}$ bestimmt. Diese Auswahl von i und k wurde getroffen, um eine möglichst homogene Verteilung der Werte der Maße über das Intervall $[0, 1]$ zu erreichen. Die Zeitkonstanten der Maße ($\rho, t_c, \Delta T, \frac{1}{q}$) betragen jeweils 1 ms.

2.5 Analyse der Potentiallandschaften

Dieser Abschnitt beschäftigt sich mit den Potentiallandschaften, die man mithilfe der Maße erzeugen kann. Die Potentiallandschaften werden erstellt, indem die mit jeweils unterschiedlichen Neuronparametern erzeugten Mess-Spiketrains mit dem Referenz-Spiketrain durch das jeweilige Maß verglichen werden. Daraufhin werden die Werte des Maßes abhängig von den Neuronparametern der Mess-Spiketrains aufgetragen.

Die Potentiallandschaften spielen bei der Wahl des Optimierungsverfahren eine wichtige Rolle. Insbesondere kann es von Bedeutung sein, ob und wie viele Maxima bei der Erzeugung der Landschaft auftreten. Die Untersuchung der lokalen Maxima (in Abgrenzung zum globalen Maximum) kann zum Beispiel Einsicht bei der Frage liefern, ob Optimierungsverfahren wie das z.B. das "Hill-Climbing-Verfahren" angewandt werden können. Der Einfachheit halber werden in dieser Arbeit hierfür jeweils 2 Neuronparameter variiert, wobei die Referenzparameter in Tabelle 2.1 aufgelistet sind.

Im Idealfall sollte eine Potentiallandschaft nur ein globales Maximum und keine weiteren lokalen Maxima besitzen. Falls es doch lokale Maxima geben sollte, sollten sich diese möglichst wenig von ihrer Umgebung abheben.

Um diesen Anforderungen gerecht zu werden, wurden die Potentiallandschaften auf Maxima und Minima untersucht und die Maxima entsprechend bewertet. Um die Potentiallandschaften zu bewerten, wird der Verlauf der Werte eines Maßes über den Bereich eines Parameters P_1 betrachtet, welcher in äquidistanten Schritten variiert wird. Der andere Parameter P_2 bleibt dabei konstant. Dies kann dann für alle Werte des Parameters P_2 wiederholt werden. Um diesen Verlauf quantitativ zu bewerten, wird folgendes Maß eingeführt:

$$M_{i,j} = \frac{B_{i,j} - \min(A_{\text{pre}}, A_{\text{post}})}{B_{\text{Global}} - A_{\text{Global}}} \quad (2.22)$$

Dem Parameter P_1 wird dabei der Index i zugeordnet während der Index j dem Parameter P_2 zugeordnet wird. Dabei ist die Bewertung des jeweiligen Maximums $M_{i,j}$ zunächst natürlich vom Wert des Maximum selbst $B_{i,j}$ abhängig. Zudem ist sie sowohl von dem globalen Maximum B_{Global} und globalem Minimum A_{Global} des jeweiligen Verlaufs, als auch von den umliegenden Minima mit jeweils um eins kleinerem bzw. größerem Index i (A_{pre} bzw. A_{post}) abhängig. Um zwischen Maxima differenzieren zu können, erhalten Maxima, die sich stark von ihrer Umgebung abheben, hohe Werte und Maxima, die sich kaum von ihrer Umgebung abheben, niedrige Werte. Die Werte, die ein Maximum erhalten kann, sind auf den Bereich zwischen 0 und 1 begrenzt.

In jedem der oben genannten Verläufe sollen nur Maxima beachtet werden, die zu-

2 Methoden

Parameter	Referenzwert
C	0.27 pF
$\tau_m = \frac{C}{g_i}$	9 ms
E_L	-70.6 mV
V_T	-50.4 mV
τ_w	150 ms
a	4.0 nS
b	0.035 nA
Δ_T	2.0 ms
V_{Spike}	0 mV
E_R	-60.0 mV
E_e	0 mV
τ_{syn_e}	5.0 ms

Tabelle 2.1: Werte der Referenzparameter zur Erzeugung der Potentiallandschaften. Die Werte wurden so gewählt das sie nahe den Werten der Original-Publikation des Modells liegen, siehe (Brette & Gerstner, 2005). Die Simulationsdauer beträgt 1000 ms. Als Stimulus dient ein Poisson-Prozess mit einer Feuerrate von 50 Hz.

sätzlich zum (erwünschten) Hauptmaximum vorkommen. Daher wird in Gleichung 2.23 das jeweilige Hauptmaximum von M abgezogen. Das Endergebnis M besteht dann aus der Summe der Bewertungen aller lokalen Maxima $M_{i,j}$ einer Potentiallandschaft (mit Ausnahme der jeweiligen Hauptmaxima).

$$M := \sum_j \left(\sum_i M_{i,j} \right) - \max(M_{i,j}) \quad (2.23)$$

Werden mehrere Potentiallandschaften analysiert, können natürlich auch die Werte aller Landschaften aufsummiert werden.

3 Ergebnisse

3.1 Untersuchung der Spiketrain-Maße

In diesem Abschnitt werden die verschiedenen Spiketrain-Maße zum einen bei unterschiedlichen Zeitparametern (siehe Abschnitt 3.1.2) und zum anderen bei zwischen den Maßen vergleichbarer Zeitparametern (siehe Abschnitt 3.1.1) untersucht. Zuletzt wird noch die Laufzeit der Berechnung der verschiedenen Maße gemessen (siehe Abschnitt 3.1.3).

3.1.1 Vergleich der Maße bei vergleichbaren Zeitparametern

Zunächst sollen die folgenden Maße bei der Verwendung vergleichbarer Zeitparameter und kontrolliert verrauschter Spiketrains (siehe Abschnitt 2.4.2) paarweise miteinander verglichen werden: das Gamma-Maß, das Victor-Purpura-Maß (mit sowohl exponentieller als auch linearer Kostenfunktion) und das reduzierte Gauß-Maß. Das Exponential- und Gauß-Maß weisen viel zu lange Laufzeiten für eine praktische Verwendung auf (siehe Abschnitt 3.1.3), und werden daher nicht weiter untersucht. Der Referenz- und Mess-Spiketrain werden, wie in Abschnitt 2.4.1 und Abschnitt 2.4.2 beschrieben, erzeugt. Zu erwarten ist, dass $\lim_{s \rightarrow 0} x = 1$ und $\lim_{s \rightarrow \infty} x = 0$, wobei x der Wert des jeweiligen Maßes ist. Begründen kann man diese Erwartungen so, dass im Fall von $s = 0$ alle Spikes übereinstimmen und damit der Wert des Maßes maximal wird. Im anderen Fall wird bei sehr großen Verschiebungen die Wahrscheinlichkeit, dass zwei Spikes innerhalb der Reichweite des jeweiligen Maßes übereinstimmen, sehr gering.

Das Ziel des PSO-Algorithmus ist die Neuronparameter zu finden, mit denen ein Spiketrain erzeugt werden kann, der beim Vergleich mittels des jeweiligen Maßes, so exakt wie möglich mit dem Referenz-Spiketrain übereinstimmt. Je nach Optimierungsverfahren kann es dabei von Vorteil sein, wenn kleine Abweichungen zwischen Referenz- und Mess-Spiketrain zu großen Änderungen des Wertes eines Maßes führen. Vergleicht man die Werte, die beim Vergleich derselben Spiketrainpaare durch unterschiedliche Maße erzielt werden (siehe Abbildung 3.1 bis Abbildung 3.4), kann man anhand dessen feststellen, welches Maß hohe Werte besser differenzieren kann. Dies wird in Abbildung 3.1 deutlich, in der die Steigung für hohe Werte kleiner als 1 ist und sich somit das exponentielle Victor-Purpura-Maß deutlich besser zur Bestimmung feiner Unterschiede im Bereich hoher Werte eignet, als das reduzierte Gauß-Maß. Das reduzierte Gauß-Maß hingegen kann Unterschiede bei niedrigen Werten besser auflösen. Dies ist zur genauen Bestimmung der Maxima allerdings nicht von Relevanz.

Eine Erkenntnis der Abbildung 3.3 ist die mangelnde Fähigkeit des Gamma-Maßes, zwischen sehr ähnlichen Spiketrains differenzieren zu können. Dies folgt daher, dass

3 Ergebnisse

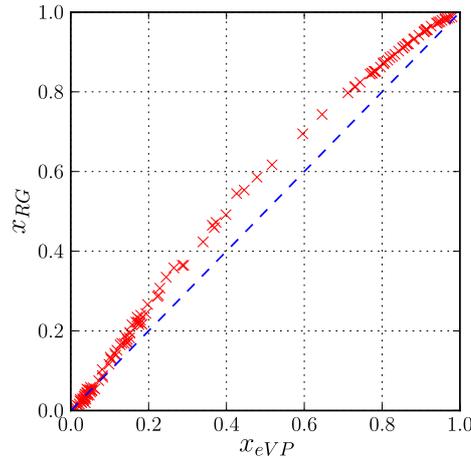


Abbildung 3.1: Vergleich des exponentiellen Victor-Purpura-Maßes mit dem reduzierten Gauß-Maß. Abhängig von der Steigung der Werte eignen sich die verschiedenen Maße in unterschiedlichen Wertebereichen besser dazu, um zwischen kleinen Änderungen im Mess-Spiketrain zu unterscheiden. Je höher die erzielten Werte, desto besser kann das exponentielle Victor-Purpura-Maß zwischen den Werten differenzieren, verglichen mit dem reduzierten Gauß-Maß.

$N_{\text{coin}} = 1$ ist, sobald sich ein Spike des Mess-Spiketrains innerhalb des Intervalls $\pm\Delta T$ eines Spikes des Referenz-Spiketrains befindet. Hierbei wird kein Unterschied zwischen Spikes des Mess-Spiketrains gemacht, die sich exakt am gleichen Zeitpunkt befinden oder um bis zu $\pm\Delta T$ verschoben sind.

Doch auch bei den anderen Maßen gibt es in dieser Hinsicht Unterschiede. Dabei kann man die Maße folgendermaßen abstufen: Bezüglich der Differenzierung zwischen hohen Werten der Maße ist das am besten geeignete Maß das exponentielle Victor-Purpura-Maß (siehe Abbildung 3.2) gefolgt vom linearen Victor-Purpura-Maß (siehe Abbildung 3.4) und dem reduzierten Gauß-Maß (siehe Abbildung 3.3). Deutlich schlechter als die anderen Maße ist das Gamma-Maß geeignet, das ähnliche Spiketrains (ähnlich in dem Sinne, dass die Spikes um weniger als $\pm\Delta T$ verschoben sind) nicht differenziert unterscheiden kann. Der Bereich hoher Werte ist von besonderer Bedeutung, weil erwartet wird, dass das Optimierungsverfahren gegen Werte nahe von 1 konvergiert.

3.1 Untersuchung der Spiketrain-Maße

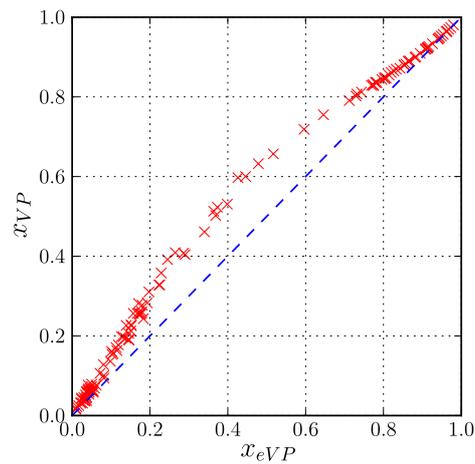


Abbildung 3.2: Vergleich des exponentiellen Victor-Purpura-Maßes mit dem Victor-Purpura-Maß. Bei sehr kleinen und großen Unterschieden zwischen den verglichenen Spiketrains sind die erzielten Ergebnisse beider Maße fast gleich. Dies erklärt sich bei niedrigen Werten dadurch, dass in diesem Fall fast keine Spikes verschoben, sondern statt dessen Spikes entfernt und eingefügt werden. Der Verlauf der Datenpunkte lässt darauf schließen, dass sich das exponentielle Victor-Purpura-Maß am besten für die Bestimmung der exakten Position des besten Parametersatzes eignet. Die Begründung hierfür lautet wieder, dass das exponentielle Victor-Purpura-Maß besser zwischen sehr hohen Werten differenzieren kann.

3 Ergebnisse

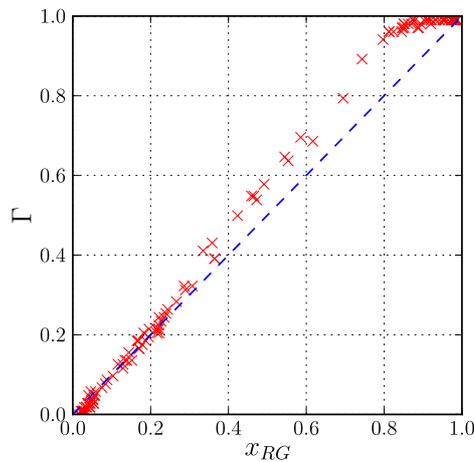


Abbildung 3.3: Vergleich des reduzierten Gauß-Maßes mit dem Gamma-Maß. Wenn die Abstände zwischen zweier verglichener Spikes kleiner als 1 ms werden, kann das Gamma-Maß nicht mehr zwischen diesen differenzieren. Daher sind alle Werte des Gamma-Maßes ab einem gewissen Punkt nahezu 1, während das reduzierte Gauß-Maß sich in diesem Bereich der 1 langsam annähert. Dies führt dazu, dass sich das Gamma-Maß schlechter für das Auffinden der genauen Position des besten Parametersatzes eignet.

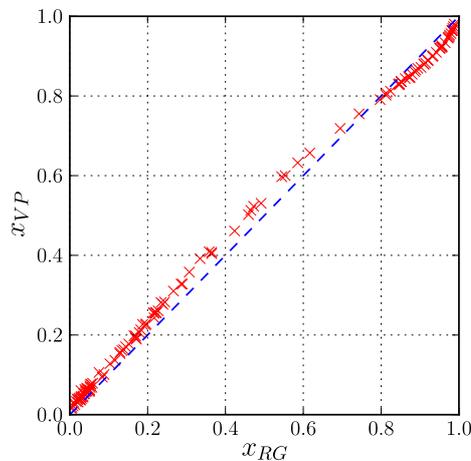


Abbildung 3.4: Vergleich des reduzierten Gauß-Maßes mit dem Victor-Purpura-Maß. Bei großen Unterschieden zwischen den verglichenen Spiketrains (hohem Wert für s) erzielen beide Maße ähnliche Ergebnisse. Um den genauen Ort eines Maximums zu finden, eignet sich das exponentielle Victor-Purpura-Maß besser, da es, wie man an der Steigung bei hohen Werten erkennen kann, besser zwischen fast identischen Spiketrains differenzieren kann.

3.1.2 Vergleich der Maße bei unterschiedlichen Zeitparametern

Ein Parameter, der entscheidenden Einfluss auf die Werte der Maße haben kann, ist der Zeitparameter c des jeweiligen Maßes. Der Einfluss soll in diesem Abschnitt veranschaulicht werden. Bei Betrachtung der Abbildung fällt auf, dass alle Maße abhängig von dem jeweiligen Zeitparameter sind. Das heißt, alle Maße können bei besonders schlechter Wahl des Zeitparameters ein unerwünschtes Verhalten zeigen. Wählt man den Zeitparameter zu groß, kann dies dazu führen, dass auch sehr unterschiedliche Spiketrains hohe Werte erzielen. Dies führt dazu, dass nicht der ganze Wertebereich ausgenutzt wird. Eine sehr niedrige Wahl des Zeitparameters kann dazu führen, dass fast alle Werte sehr niedrig sind. Eine falsche Wahl kann auf das Gauß- und das Gamma-Maß besonders gravierende Auswirkungen haben, da ihre Normierung zu negativen Werten beim Gamma-Maß, und zu Werten über 1 beim Gauß-Maß führen kann.

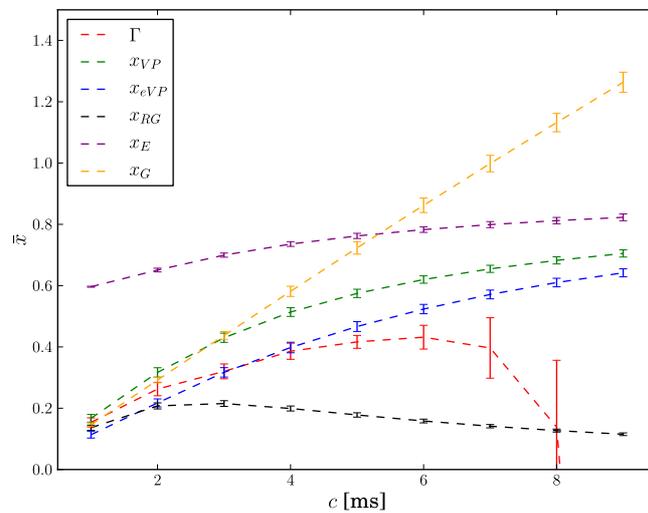


Abbildung 3.5: Die Abbildung beschreibt das Verhalten der Maße bei unterschiedlichen Zeitparametern unter Verwendung der ersten 6 postsynaptischen Spiketrains der Wettbewerbsdaten. Der erste postsynaptische Spiketrain wird mit 5 anderen verglichen. Hier aufgetragen ist der Mittelwert \bar{x} aus diesen 5 Vergleichen, mit der Standardabweichung dieser 5 Werte als Fehler. Die Skalierungsparameter sind jeweils in Abhängigkeit einer verallgemeinerten Zeitkonstante c gewählt. Dabei gilt: $c = \Delta T = \rho = t_c = \frac{1}{q}$. Dabei sind ΔT , ρ , $\frac{1}{q}$ und t_c in ms angegeben.

3.1.3 Laufzeitmessung der Berechnung der Maße

Ein weiterer Punkt, der von Interesse ist, ist die Laufzeitmessung der Maße. Sollte die Berechnung der Maße deutlich mehr Zeit verbrauchen, als beispielsweise die anderen Re-

3 Ergebnisse

Maß	Laufzeit T_l für 75 Spiketrains
exponentielle Victor-Purpura-Maß	63.42 s
Victor-Purpura-Maß	61.39 s
reduziertes Gauß-Maß	18.63 s
Gamma-Maß	0.08 s
Exponential-Maß	1855.11 s
Gauß-Maß	551.45 s

Tabelle 3.1: Laufzeitmessung der verschiedenen Maße. Der Prozessor der für die Simulation verwendet wurde ist ein AMD Athlon(tm) 64 X2 Dual Core Prozessor 4600+.

chenschritte eines Optimierungsverfahrens, kann es sein, dass ein eigentlich “schlechteres” Maß effizienter zu benutzen ist. Dann können möglicherweise mehr Iterationen benötigt werden, das finden des besten Parametersatzes trotzdem jedoch schneller sein. In diesem Abschnitt werden die verschiedenen Maße anhand von den ersten 5 Sekunden der 76 postsynaptischen Spiketrains der “Single Neuron Modeling Competition” (siehe Abschnitt 1.3) auf ihre Schnelligkeit bei der Berechnung mit einem Computer untersucht. Innerhalb dieses Zeitraums von 5 s weisen die Spiketrains jeweils 150-200 Spikes auf. Dabei wurde einer dieser Spiketrains als Referenz-Spiketrain gewählt und die anderen 75 Spiketrains mit diesem verglichen. Da die Laufzeit der Berechnung natürlich auch vom jeweiligen Computer abhängt, ist vor allem der relative Unterschied zwischen den Berechnungszeiten von Interesse. Deshalb werden die Berechnungszeiten der Maße in Abbildung 3.6 als Vielfache des schnellsten Maßes (des Gamma-Maßes) angegeben. Dabei muss man beachten, dass die Berechnung aller Maße, die keine rechenintensiven Integrations-schritte beinhalten, durch Implementierung in C-Code deutlich beschleunigt werden können. Auch die Maße, die auf Integration zurückgreifen, können optimiert werden, z.B. durch Wahl effizienterer Integrationsverfahren, allerdings voraussichtlich in geringerem Maße.

Eine Erkenntnis dieses Abschnittes ist, dass das Gauß- und Exponential-Maß aufgrund der vielen zu berechnenden Integrale um mehrere Größenordnungen langsamer sind als die anderen Maße. Aufgrund dieser Erkenntnis werden beide Maße nicht weiter untersucht. Die durchschnittliche Berechnungsdauer der verbliebenen liegt immerhin noch unter einer Sekunde pro Vergleich. Für die theoretischen Untersuchungen im Rahmen dieser Arbeit war diese Laufzeit kein Hindernis. Im Fall einer regelmäßigen, wiederholten Anwendung - wie beispielsweise zur Kalibrierung einer großen Anzahl von Neuronen auf neuromorpher Hardware - ist eine Optimierung aber auf jeden Fall notwendig. Auch wenn beispielsweise die beiden zu vergleichenden Spiketrains eine sehr hohe Feuerrate aufweisen sollten, würden alle Maße, bis auf das Gamma-Maß, in ihrer jetzigen Form das PSO-Verfahren erheblich verlangsamen. Die Komplexität der Maße, und damit deren Laufzeit, ist nicht nur von dem jeweiligen Maß abhängig sondern auch von der Anzahl der Spikes.

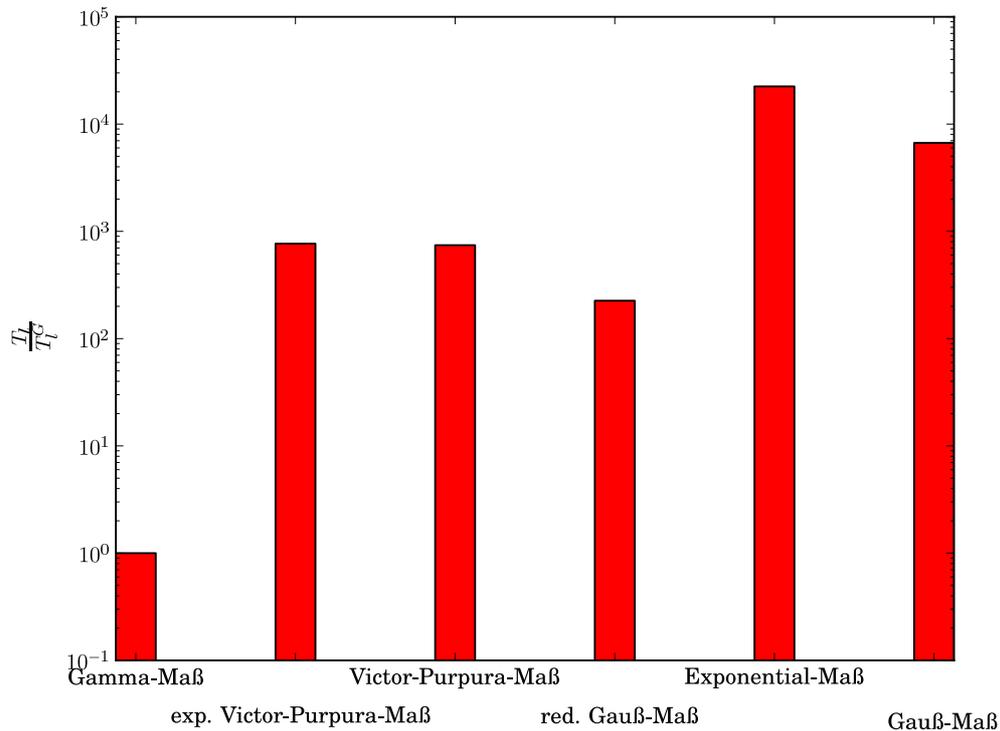


Abbildung 3.6: Die Berechnungszeiten T_l im Verhältnis zur Laufzeit des Gamma-Maßes T_l^G . Das Gamma-Maß ist um mehrere Größenordnungen schneller als alle anderen Maße in ihrer derzeitigen Umsetzung.

3.2 Analyse der Potentiallandschaften

In diesem Abschnitt wird das Verhalten der verschiedenen Maße bei der Veränderung eines oder mehrerer Neuronparameter untersucht, um zum einen das PSO-Verfahren zu optimieren und auch, um eine Grundlage zur Wahl des optimalen Optimierungsverfahren treffen zu können. Bei der Benutzung des Optimierungsverfahrens werden alle Parameter gleichzeitig optimiert. Der Einfachheit wegen wurden hier die Neuronparameter nur paarweise variiert. Dabei kann man unter anderem auch Abhängigkeiten zwischen den Parametern erkennen. Um die Potentiallandschaften bewerten zu können, wird zu diesem Zweck ein Maß eingeführt, welches sich an der Anzahl und der Höhe der auftretenden Maxima orientiert (siehe Gleichung 2.22 und 2.23). Die Abbildung 3.8 und 3.9 zeigen die Werte für das Gamma- bzw. das Victor-Purpura-Maß über zwei Neuronparameter.

Das Ziel dieser Untersuchung ist die Auffindung desjenigen Maßes, welches eine möglichst glatte und unimodale Potentiallandschaft erzeugt. Dies geschieht hier jeweils für Paare von Neuronparametern damit auch Abhängigkeiten zwischen Neuronparametern, die sich z.B. gegenseitig kompensieren, entdeckt werden können. Insgesamt sind 36 ver-

Parameter	Standardwert	Interval, falls variiert
C	0.27 pF	[0.24, 0.30] pF
$\tau_m = \frac{C}{g_t}$	9 ms	[5, 13] ms
E_L	-70.6 mV	[-75.6, -65.6] mV
V_T	-50.4 mV	[-55.4, -45.4] mV
τ_w	150 ms	[100, 200] ms
a	4.0 nS	nicht variiert
b	0.035 nA	[0.01, 0.1] nA
Δ_T	2.0 ms	nicht variiert
V_{Spike}	0 mV	nicht variiert
E_R	-60.0 mV	[-65.0, -55.0] mV
E_e	0 mV	[-5.0, 5.0]
τ_{syn_e}	5.0 ms	[2.0, 8.0] ms

Tabelle 3.2: In dieser Tabelle werden die Neuronparameter, die zur Simulation verwendet werden, zusammengefasst. Dabei werden nicht alle Parameter variiert, da es besonders bei bestimmten Parametern zu Abbrüchen der Simulation aufgrund von Integrationsproblemen in NEST kommt. Zudem traten weitere Fehler auf die aus Zeitgründen nicht behoben werden konnten. Der Parameter V_{Spike} wurde nicht variiert da dieser voraussichtlich keinen großen Einfluss auf den jeweiligen Spiketrain haben wird. Die Parameter a und der Steigungsparameter der Exponentialfunktion Δ_T erweisen sich als problematisch, da der Integrator von NEST bei bestimmten Werten dieser Parameter die Integrations-Schrittweite nicht mehr wie benötigt anpassen kann und werden daher nicht variiert.

schiedene Kombinationen an Parametern möglich, daher liegen den folgenden Ergebnissen genau 36 Potentiallandschaften zugrunde.

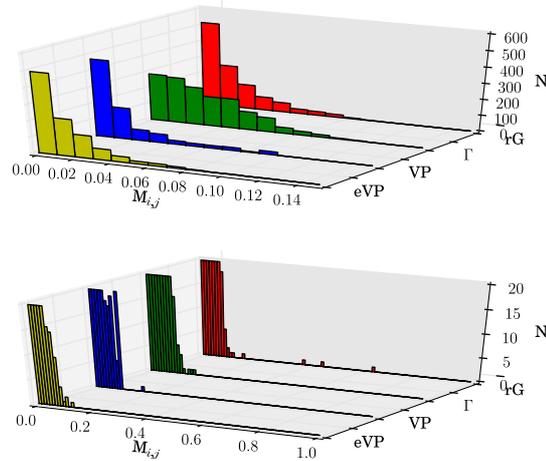


Abbildung 3.7: In dieser Abbildung wird die Anzahl N der normierten lokalen Maxima $M_{i,j}$ aller erzeugten Potentiallandschaften verglichen. Dabei werden die Maxima entsprechend ihrer Ausprägung sortiert. Die obere Abbildung zeigt dabei die Verteilung der Maxima bis zu einer Ausprägung von 0.15. Das Gamma-Maß führt hier zu Maxima mit größerer Ausprägung. Dieses Verhalten spiegelt sich auch in Abbildung 3.8 wieder. In der unteren Abbildung sieht man, dass bei manchen Potentiallandschaften vereinzelt noch sehr große Maxima auftreten, die bei genauerer Betrachtung der Potentiallandschaften allerdings nicht ins Bild passen und voraussichtlich auf Rechenfehlern basieren.

Die Anzahl und Ausprägungen der Nebenmaxima wird in Abbildung 3.7 gezeigt. Das aufsummierte Endergebnis M fasst Abbildung 3.7 für jede Metrik in Tabelle 3.3 in einem Zahlenwert zusammen. Beim Vergleich der Maße kann der Wert von M unterschiedliche Ursachen haben; zum Beispiel kann derselbe Wert sowohl von wenigen großen als auch von vielen kleinen Maxima erzeugt werden. Das Ergebnis von M liefert dabei eine quantitative Bewertung der Potentiallandschaften, während die Verteilung der Maxima die Entstehung dieses Wertes erklären kann. In Bezug zur Wahl des Optimierungsverfahrens können beide relevant sein. Wie man Tabelle 3.3 entnehmen kann, wird das beste Ergebnis vom linearen Victor-Purpura-Maß erzielt, wobei das Ergebnis des reduzierten Gauß-Maßes und des exponentiellen Victor-Purpura-Maßes nicht wesentlich schlechter ausfallen. Das Gamma-Maß hat dabei mit $M = 46.43$ Punkten mit Abstand das schlechteste Ergebnis.

Dieses Ergebnis lässt sich intuitiv anhand der Potentiallandschaften erklären, wie man

3 Ergebnisse

verwendetes Maß	M
exponentielle Victor-Purpura-Maß	21.38
Victor-Purpura-Maß	15.43
reduziertes Gauß-Maß	16.59
Gamma-Maß	46.43

Tabelle 3.3: In dieser Tabelle ist M (siehe Abschnitt 2.5) für die jeweiligen Maße aufgelistet.

in den Abbildungen 3.8 und 3.9 gut erkennen kann.

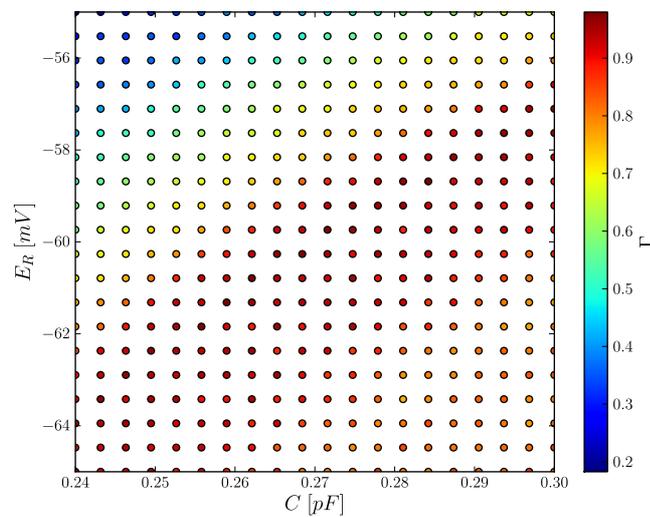


Abbildung 3.8: Verlauf des Maßes M über die Neuronparameter C und E_R mit dem Gamma-Maß. Wie man zum Beispiel im Fall des Verlaufs der Datenpunkte bei $E_R = -62 \text{ mV}$ sehen kann, gibt es viele kleine lokale Maxima. In dieser Abbildung kann man auch sehen, dass sich die 2 Parameter zum Teil gegenseitig aufhaben, da die Potentiallandschaft kein klar ersichtliches punktuell Maximum sondern ein “Gebirgskamm” darstellt.

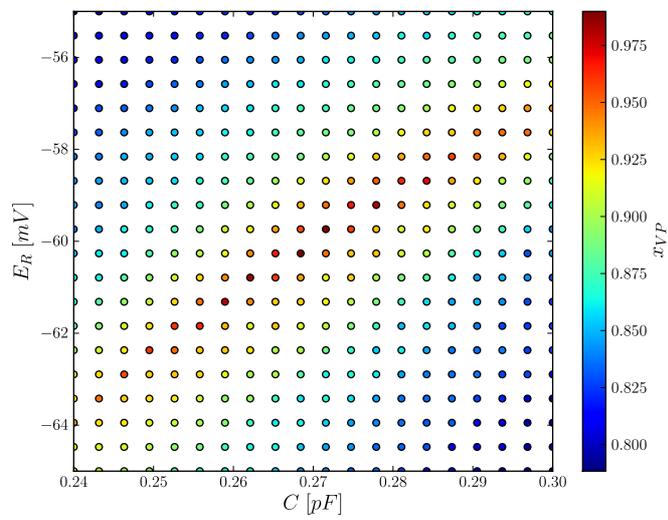


Abbildung 3.9: Verlauf des Maßes M über die Neuronparameter C und E_R mit dem Victor-Purpura-Maß. Im Vergleich zu Abbildung 3.8 sieht man für das Victor-Purpura-Maß eine deutlich “glattere” Landschaft, die weniger lokale Maxima aufweist.

4 Diskussion und Ausblick

Eine realitätsgetreue Anpassung der Neuronparameter des AdEx-Modells (siehe Abschnitt 1.2) an biologische Messungen erfordert die Verwendung von Spikes als Stimulus. Die unmittelbare Folge der Komplexität des AdEx-Modells ist, dass über hochdimensionale Parameterräume optimiert werden muss. Eine gute Wahl des zu optimierenden Maßes kann die Funktionalität des zur Zeit verwendeten Optimierungsalgorithmus (siehe Abschnitt 2.1) verbessern bzw. die Wahl eines effizienteren Optimierungsalgorithmus ermöglichen.

Eine Feststellung bei der Untersuchung der Maße bezüglich deren Zeitparameter ist, dass diese einen großen Einfluss auf die Robustheit der Maße gegenüber Spiketrains mit unterschiedlichen Feuerraten haben. Die Wahl des verwendeten Zeitparameters muss gut überlegt sein, da diese Wahl je nach Feuerrate der verwendeten Spiketrains nur bis zu einer bestimmten Obergrenze sinnvoll ist.

Des Weiteren wurden die verschiedenen Maße bezüglich ihres Auflösungsvermögens über den möglichen Wertebereich miteinander verglichen. Der Abschnitt 3.1.1 lässt Aussagen über die Eignung der Maße für das PSO-Verfahren zu. Dabei ist das Gamma-Maß in der Genauigkeit zur Bestimmung der Parameter limitiert, da das Maß kaum Auflösungsvermögen für sehr ähnliche Spiketrains besitzt. Diese Limitierung hängt vom gewählten Zeitparameter des Maßes ab. Je kleiner dieser gewählt wird, desto genauer lässt sich voraussichtlich der optimale Parametersatz bestimmen, desto geringer wird allerdings voraussichtlich auch die Chance, das jeweilige Maximum überhaupt zu finden, da sich die Werte des Maßes in diesem Fall alle in einem kleinen Bereich niedriger Werte befinden werden. Sehr unterschiedliche Spiketrains lassen sich schlecht auflösen. Dies ist insbesondere bei Simulationen schlecht, bei denen ein großer Parameterraum erkundet werden muss, da dies sehr unterschiedliche Spiketrains zur Folge hat. Nach mehreren Iterationen werden allerdings voraussichtlich sehr ähnliche Spiketrains erzeugt, und das Auflösungsvermögen bei hohen Werten des verwendeten Distanzmaßes wird relevant. Eine Möglichkeit wäre daher, die Zeitparameter anzupassen, wenn der Ort des voraussichtlichen Maximums eingegrenzt wurde. Ohne Anpassung der Zeitparameter eignen sich die anderen Maße (im Vergleich zum Gamma-Maß) für eine Optimierung besser, da sie sowohl den hohen Wertebereich besser auflösen können, als auch weniger anfällig in Bezug auf die Wahl des Zeitparameters sind.

Bei der Laufzeitmessung der Berechnung der Maße stellte sich heraus, dass die verwendeten Berechnungsalgorithmen noch erheblich optimiert werden müssen. Dabei waren die Laufzeiten zweier Maße, dem Gauß-Maß und dem Exponential-Maß, trotz Integration in C-Code, in gravierendem Ausmaß zu groß. Da die Simulationsdauer im Vergleich zur Berechnung dieser Maße um Größenordnungen kürzer ist, lohnt sich keine weitere Betrachtung dieser Maße. Bei den Victor-Purpura-Maßen und dem Gamma-Maß ist bei der

Umschreibung in C-Code im Vergleich zu Python ein Geschwindigkeitszuwachs der jeweils gleichen Größenordnung zu erwarten. Das reduzierte Gauß-Maß muss deutlich weniger Integrationsschritte als z.B. das Gauß-Maß durchführen und ist daher in seiner jetzigen Form sogar schneller als das Victor-Purpura-Maß, allerdings kann man es voraussichtlich weniger stark optimieren im Vergleich zu den restlichen Maßen. Insgesamt betrachtet ist das Gamma-Maß das mit Abstand schnellste Maß, während die Laufzeit der Victor-Purpura-Maße und des reduzierten Gauß-Maßes nach der Umschreibung in C-Code befriedigend sein dürfte.

Zuletzt wurden die Potentiallandschaften untersucht, mit dem Ergebnis, dass sich das Victor-Purpura-Maß am besten für eine Optimierung eignet. Die jeweilige verwendete Kostenfunktion spielt dabei eher eine untergeordnete Rolle, da die Ergebnisse, die mit beiden Kostenfunktionen erzielt werden, dicht beisammen liegen.

Neben der Optimierung der Maße bezüglich ihrer Berechnungsdauer, besteht auch für den Algorithmus, der die Maxima der Potentiallandschaften bewertet, Optimierungsbedarf. Dieser weist momentan noch Schwächen in bestimmten künstlich konstruierten Sonderfällen auf, bei denen im Zusammenhang mit dem Neuronmodell und dem verwendeten Maß bestimmte Maxima viel zu klein eingeschätzt werden. Da die Wahrscheinlichkeit, dass diese Sonderfälle im vorgestellten Optimierungsproblem auftreten, gering ist, reicht jedoch der bestehende Algorithmus für die grobe Abschätzung der Güte der Potentiallandschaften aus.

Ein sehr interessanter Punkt, der aufgrund von technischen Problemen und aus Zeitgründen leider nicht mehr untersucht werden konnte, ist der Einfluss der Maße auf die Anzahl der Iterationen, die beim PSO-Verfahren benötigt werden, um einen zufriedenstellenden Parametersatz zu erhalten. Allerdings kann man auch hier aufgrund der Ergebnisse in Abschnitt 3.2 und Abschnitt 3.1.1 vermuten, dass das Gamma-Maß auch in dieser Hinsicht voraussichtlich das am wenigsten effiziente Maß ist. Die Begründung dafür sind sowohl die Schwankungen in den Potentiallandschaften als auch die mangelnde Robustheit des Maßes gegenüber Spiketrains unterschiedlicher Feuerrate. Insgesamt eignet sich das Victor-Purpura-Maß voraussichtlich am besten für das PSO-Verfahren, unter der Voraussetzung, dass die Laufzeiten wie erhofft durch das Umschreiben des Maßes in C-Code noch deutlich reduziert werden können. Diese Vermutung sollte allerdings durch eine erneute Laufzeitmessung bestätigt werden. Nach diesem Schritt kann man dann erneut abwägen, welches Maß wirklich das effizienteste ist, da die erheblich schnellere Laufzeit des Gamma-Maßes natürlich auch mehr Iterationen des PSO-Verfahrens in der selben Zeit zulässt. Da eine solche vollständige Untersuchung nicht erfolgt ist, soll mit Abbildung 4.1 und Abbildung 4.2 noch einmal die Notwendigkeit dafür begründet werden. Die Abbildung 4.1 zeigt ein Beispiel in dem die Optimierung bei Benutzung des Gamma-Maßes ein schlechteres Resultat für die Neuronparameter als bei Benutzung des Victor-Purpura-Maßes erzielt. Diese Optimierung beruht auf den selben Parametern, mit denen schon Abbildung 3.8 und Abbildung 3.9 erzeugt wurden.

Letztendlich ist natürlich das Ziel die Software Simulation des AdEx-Neurons durch die BrainScaleS-Hardware zu ersetzen und dann damit an der Single Neuron Modeling Competition teilzunehmen.

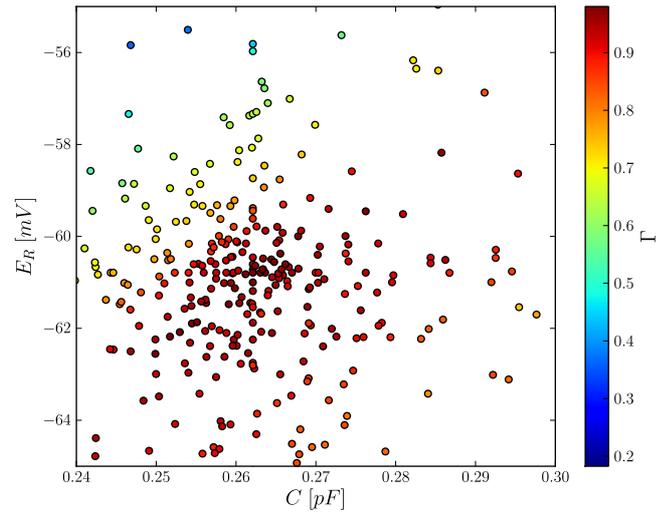


Abbildung 4.1: Ergebnis des PSO-Verfahrens nach 10 Iterationen: Bei Benutzung des Gamma-Maßes konvergiert das Ergebnis nicht zu den Parametern, mit denen der Referenz-Spiketrain erzeugt wurde ($C = 0.27$ pF und $E_R = -60$ mV), sondern zu $C = 0.2621$ pF und $E_R = -60,79$ mV. Dies scheint die Vermutung zu bestätigen, dass sich das Gamma-Maß schlecht zur Optimierung eignet, wie in Abschnitt 3.1.1 und 3.2 beschrieben. Allerdings ist eine Untersuchung des gesamten Parameterraums und mehrerer verschiedener Saate des Zufallsgenerators notwendig, um eine wirklich fundierte Aussage bezüglich der Maße in Verbindung mit dem PSO-Verfahren treffen zu können.

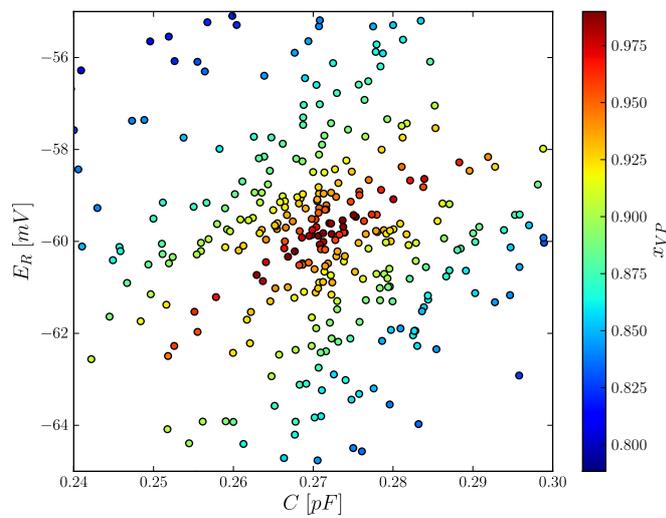


Abbildung 4.2: Ergebnis des PSO-Verfahrens nach 10 Iterationen: Bei Benutzung des Victor-Purpura-Maßes konvergiert das Ergebnis ziemlich präzise zu den Parametern, mit denen der Referenz-Spiketrain erzeugt wurde. Die beste Position im Parameterraum ist $C = 0.2698$ pF und $E_R = -60,03$ mV, während die Referenzdaten $C = 0.27$ pF und $E_R = -60$ mV betragen.

Literaturverzeichnis

- Brette, R., & Gerstner, W. (2005). Adaptive exponential integrate-and-fire model as an effective description of neuronal activity. *J. Neurophysiol.* *94*, 3637 – 3642.
- Carandini, M., Horton, J. C., & Sincich, L. C. (2007). Thalamic filtering of retinal spike trains by postsynaptic summation. *Journal of Vision* *7*(14).
- Davison, A. P., Brüderle, D., Eppler, J., Kremkow, J., Müller, E., Pecevski, D., Perrinet, L., & Yger, P. (2008). PyNN: a common interface for neuronal network simulators. *Front. Neuroinform.* *2*(11).
- Gewaltig, M.-O., & Diesmann, M. (2007). NEST (NEural Simulation Tool). *Scholarpedia* *2*(4), 1430.
- Jolivet, R., Kobayashi, R., Rauch, A., Naud, R., Shinomoto, S., & Gerstner, W. (2008). A benchmark test for a quantitative assessment of simple neuron models. *Journal of Neuroscience Methods* *169*(2), 417 – 424.
- Kistler, W., Gerstner, W., & van Hemmen, J. L. (1997). Reduction of the Hodgkin-Huxley equations to a single-variable threshold model. *Neural Computation* *9*, 1015–1045.
- Naud, R., Marcille, N., Clopath, C., & Gerstner, W. (2008). Firing patterns in the adaptive exponential integrate-and-fire model. *Biological Cybernetics* *99*(4), 335–347.
- Pfeil, T. (2011). Configuration strategies for neurons and synaptic learning in large-scale neuromorphic hardware systems. Diploma thesis (English), University of Heidelberg, HD-KIP 11-34.
- Rossant, C., Goodman, D. F., Platkiewicz, J., & Brette, R. (2010). Automatic fitting of spiking neuron models to electrophysiological recordings. *Frontiers in Neuroinformatics* *4*.
- Sincich, L. C., Adams, D. L., Economides, J. R., & Horton, J. C. (2007). Transmission of spike trains at the retinogeniculate synapse. *Journal of Neuroscience* *27*(10), 2683–2692.
- van Rossum, M. C. W. (2001). A novel spike distance. *Neural Computation* *13*(4), 751–763.
- Victor, J. D., & Purpura, K. P. (1996). Nature and precision of temporal coding in visual cortex: a metric-space analysis. *J Neurophysiol* *76*(2), 1310–1326.

Danksagungen

Zum Schluss möchte ich mich bei allen bedanken, die mich während meiner Bachelorarbeit unterstützt haben:

- Herrn Professor Dr. Meier für die freundliche Aufnahme in die Arbeitsgruppe und die Unterstützung während meiner Bachelorarbeit
- Herrn Dr. Johannes Schemmel dafür mein Zweitprüfer zu sein
- Eric für die Hilfe bei technischen Problemen
- Den Kicker-Menschen für einige spannende Spiele
- Kathi dafür in den letzten Wochen ohne mich auskommen zu müssen
- meinen Eltern, die mir die Bachelorarbeit ermöglicht haben
- allen anderen in der Visionsgruppe für die freundliche Atmosphäre
- Besonderer Dank gilt Mihai und Thomas
- Mihai für das Diskutieren und Korrekturlesen meiner Arbeit sowie insbesondere für die Unterstützung während der ganzen Zeit in der Visions-Gruppe
- Thomas für seine Geduld und immerwährende Unterstützung und Hilfe während der ganzen Zeit meiner Bachelorarbeit, ich hätte mir keinen besseren Betreuer wünschen können
- und natürlich allen, die ich in der Eile vielleicht Vergessen habe zu erwähnen

Erklärung:

Ich versichere, dass ich diese Arbeit selbständig verfasst und keine anderen als die angegebenen Quellen und Hilfsmittel benutzt habe.

Heidelberg, 12. März 2012

.....
(Unterschrift)