

Versuch F09

Neuromorphic Computing

Xeno Boecker, Jan Jakob

Laborbuch und Auswertung

Inhaltsverzeichnis

1 Untersuchung eines einzelnen Neurons	1
1.1 Aufgabe 1	1
1.2 Aufgabe 2	1
1.3 Aufgabe 3	1
1.4 Aufgabe 4	2
2 Kalibrierung der Neuron-Parameter	4
2.1 Aufgabe 1	4
2.2 Aufgabe 2	4
2.3 Aufgabe 3	4
2.4 Aufgabe 4	5
2.5 Aufgabe 5	5
3 Einzelnes Neuron mit synaptischem Input	6
3.1 Aufgabe 1	6
3.2 Aufgabe 2	7
3.3 Aufgabe 3	7
3.4 Aufgabe 4	7
4 Kurzzeit-Plastizität	8
4.1 Aufgabe 1	8
4.2 Aufgabe 2	12
4.3 Aufgabe 3	12
5 Feed-Forward Netzwerke	16
5.1 Aufgabe 1	16
5.2 Aufgabe 2	17
5.3 Aufgabe 3	17
6 Rekurrente Netzwerke	19
6.1 Aufgabe 1	19
6.2 Aufgabe 2	19
6.3 Aufgabe 3	20
7 Eine einfache Berechnung - XOR	21
7.1 Aufgabe 1	21
7.2 Aufgabe 2	21
7.3 Aufgabe 3	21
7.4 Aufgabe 4	22

1 Untersuchung eines einzelnen Neurons

Im ersten Experiment untersuchen wir das Zündverhalten eines einzelnen Neurons ohne Input / äußere Stimuli. Das Neuron kann in einen kontinuierlichen Feuerzustand versetzt werden, wenn das Leck-Umkehrpotential die Zündschwelle übersteigt.

1.1 Aufgabe 1

Im Folgenden zeichnen wir ein Ersatzschaltbild eines Neurons in der beschriebenen Konfiguration. Dazu nehmen wir das Neuronenschema in Abbildung 5 als Referenz und stellen uns das Neuron als einen Kondensator vor, der an mehrere Spannungsquellen angeschlossen ist. Die verschiedenen Ionenkanäle der Nervenzelle werden zu einem einzigen erregenden und einem einzigen hemmenden Potenzial zusammengefasst, und die Diffusion durch die Membran wird durch das Leckpotential ersetzt.

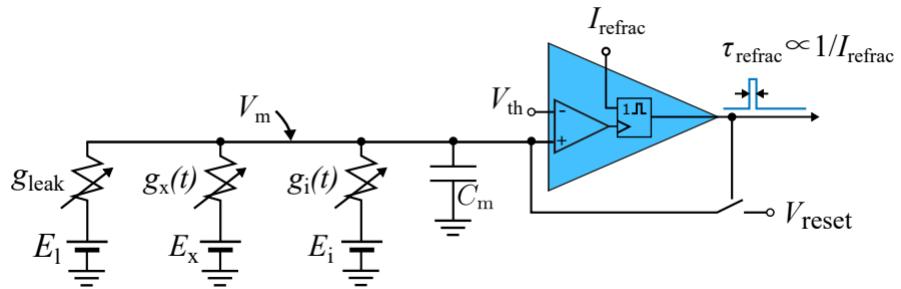


Abbildung 1: Ersatzschaltbild eines Neurons in der beschriebenen Konfiguration

Frage: Welche Parameter des Neurons beeinflussen die Feuerrate?

1. Refraktärzeit τ_{refrac}
2. Membranzeitkonstante $\tau_m = \frac{C_m}{g_l}$
3. Schwellwert- und Resetspannung $V_{\text{th}}, V_{\text{reset}}$

1.2 Aufgabe 2

Bei dieser Aufgabe haben wir nur wie in der Versuchsanleitung beschrieben Vorbereitungen getroffen und die aktuelle Shell für die Verwendung von Spikey eingerichtet.

1.3 Aufgabe 3

Auf dem Spikey-Chip wird ein einzelnes Neuron mit folgenden Parametern konfiguriert:

Tabelle 1: Parameter für ein einzelnes Neuron

Parameter	Wert
V_{reset}	-80.0mV
E_{rev_I}	-75.0mV
V_{rest}	-50.0mV
V_{thresh}	-55.0mV
g_{leak}	20.0

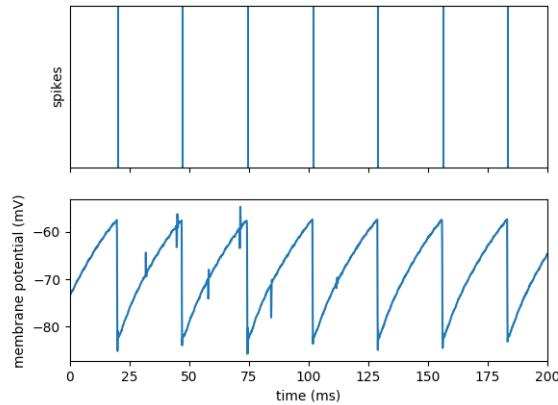
Der Potentialverlauf ist in Abbildung ?? dargestellt. Wir lesen für die Werte V_{reset} und V_{th} folgende Werte ab:

Tabelle 2: Abgelesene Werte aus dem Verlauf des Membranpotentials

Parameter	Wert
V_{reset}	$-83.0 \pm 3.0\text{mV}$
V_{thresh}	$-56.0 \pm 3.0\text{mV}$

Die Werte liegen jeweils im 3σ -Bereich.

Abbildung 2: Verlauf des Membranpotentials



1.4 Aufgabe 4

Durch Einsatz des Oszilloskops bestimmen wir die durchschnittliche Feuerrate des Neurons und ihre Standardabweichung mit Hilfe der Mess- und Statistikfunktion. Wir erhalten folgende Ergebnisse:

$$f_{firing} = 355.4 \pm 3.4\text{kHz}$$

Abbildung 3: durchschnittliche Feuerrate durch Einsatz des Oszilloskops



Wir berechnen die mittlere Feuerrate der im Spikes-Array empfangenen Spikes und vergleichen sie mit der Messung des Oszilloskops. Um eine Verteilung der Interspike-Intervalle (ISI) zu erhalten, berechnen wir die paarweise Differenz der empfangenen Spike-Zeiten und speichern sie in einem neuen Array. Den Mittelwert dieser Differenzen und seine Standardabweichung berechnen wir mit den entsprechenden NumPy-Funktionen.

1 Untersuchung eines einzelnen Neurons

Listing 1: Berechnung der mittleren Feuerrate

```
# time differences
timeDifferences = np.diff(spikes)

print("time_differences_mean", np.mean(timeDifferences))
print("time_differences_std_deviat", np.std(timeDifferences))

print("firing_rate_mean", 1e4 / np.mean(timeDifferences))
print("firing_rate_std_deviat", 1e4 / np.std(timeDifferences))
```

Damit erhalten wir in Übereinstimmung mit der obig bestimmten Feuerrate einen Wert von

$$f_{firing} = 356.5 \pm 4.7 \text{kHz}.$$

Den Faktor $1e4$ benötigen wir zur Umrechnung von biologischen Zeitskalen auf Chip-Zeitskalen.

2 Kalibrierung der Neuron-Parameter

Durch Produktionsschwankungen bei der Herstellung des Chips kommt es zur sogenannten fixed pattern noise, was dazu führt, dass die Neuronen und die Synapsen-Parameter über den Chip variieren. Dies lässt sich durch Kalibrationsroutinen minimieren, was hier am Beispiel der Membranzeitkonstante τ_m durchgeführt wird.

2.1 Aufgabe 1

Zunächst berechnen wir die theoretische Feuerrate aus den voreingestellten Werten. Nach Gleichung (6) aus der Versuchsanleitung gilt:

$$I_m(t) = C_m \frac{dV_m(t)}{dt} = g_l(E_l - V_m(t)) \quad (1)$$

Nach Lösen der DGL erhält man:

$$V_m(t) = E_{leak} + (V_{res} - E_{leak}) \cdot e^{-\frac{1}{\tau_m t}} \quad (2)$$

Es soll nun

$$V_m(t) = V_{thresh} \quad (3)$$

gelten. Mit den Werten aus dem Python Skript erhält man nach Umformen von (3) nach t einen Wert von:

$$f_{firing} = -52.86 \text{Hz}$$

Die dabei verwendeten Werte für V_{thresh} usw. entsprechen den in Versuchsteil 1 verwendeten Werten und können Tabelle 1 entnommen werden.

2.2 Aufgabe 2

Wir setzen den Schwellwert gerade auf den 1/e-ten Teil über V_{reset} , damit ist dieser nach τ_m erreicht. Eine Periode hat also gerade die Dauer $\tau_m + \tau_{refrac}$.

$$V_m(t = 1/\tau_m) = V_{thresh} = E_{leak} + (V_{res} - E_{leak}) \cdot e \quad (4)$$

Stellen wir Gleichung (4) nach V_{thresh} , so erhalten wir

$$V_{thresh} = E_{leak} - 1/e \cdot (E_{leak} - V_{reset}), \quad (5)$$

in Übereinstimmung mit Gleichung (8) aus der Versuchsanleitung. Mit den Parametern aus dem Skript ergibt sich ein neuer Wert von

$$V_{thresh} = -51.69 \text{mV}. \quad (6)$$

2.3 Aufgabe 3

Wir ändern die Einstellung für V_{thresh} im Skript entsprechend und führen das Skript aus. Außerdem stellen wir die Oszilloskop-Aufzeichnung so ein, dass alle 4 Membranspannungen zu sehen sind (Abbildung ??). Wir verwenden die Messfunktionen des Oszilloskops, um gleichzeitig die Feuerfrequenz der vier angeschlossenen Neuronen zu messen. Außerdem aktivieren wir die Statistikfunktion, damit das Oszilloskop Mittelwerte und Standardabweichung berechnet. Wir erhalten folgende Werte:

Tabelle 3: Feuerrate der vier Neuronen

Neuron	Feuerrate
1	$377.4 \pm 36.8 \text{kHz}$
2	$343.8 \pm 39.6 \text{kHz}$
3	$446.7 \pm 24.8 \text{kHz}$
4	$218.8 \pm 73.9 \text{kHz}$

2 Kalibrierung der Neuron-Parameter

Abbildung 4: Membranspannungen bei vier verschiedenen Neuronen



2.4 Aufgabe 4

Über die Variation der Leitfähigkeiten g_{leak} kann die Feuerrate verändert werden. Wir kalibrieren die 4 Neuronen für eine identische Feuerrate und damit identische Membranzeitkonstante, indem wir einen neuen Leckleitwert g'_{leak} für jedes Neuron einstellen. Zur Berechnung von g'_{leak} verwenden wir folgende Überlegung:

$$\begin{aligned}f_{firing} &= g_{leak}/C_m \\f_{firing}' &= g_{leak}/C_m * 1/g_{leak} * g'_{leak} \\&= f_{firing}/g_{leak} * g'_{leak} \\g'_{leak} &= f_{firing}' / f_{firing} * g_{leak}\end{aligned}$$

Setzen wir die entsprechenden Werte ein, so erhalten wir:

Tabelle 4: Neue g'_{leak} Werte der vier Neuronen

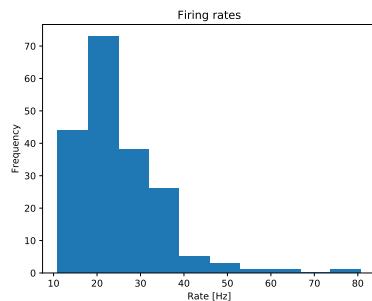
Neuron	g'_{leak}
1	145.6nS
2	90.8nS
3	115.1nS
4	107.5nS

Die Schwankungen kommen von Herstellungsungenauigkeiten (fixed pattern noise). Zudem ist uns aufgefallen, dass es eine gewisse trial to trial Variabilität gab (durch elektrisches und thermisches Rauschen).

2.5 Aufgabe 5

Wie man sehen kann, erhält man eine asymmetrische Verteilung, die poissonartig aussieht. Dies ist in sehr guter Übereinstimmung mit dem Histogramm aus der Versuchsanleitung.

Abbildung 5: Histogramm der Membranzeitkonstanten von 192 unkalibrierten Neuronen



3 Einzelnes Neuron mit synaptischem Input

In dieser Aufgabe bewerten wir den Einfluss des synaptischen Inputs auf das Neuron.

3.1 Aufgabe 1

Die Parameter drvifall und drviout werden variiert, dabei wird beobachtet, dass die EPSP-Kurve für kleine Werte von drvifall schneller abfällt und der Peak wird größer. Für kleinere Werte von drviout wird der Peak hingegen leicht kleiner und die EPSP-Kurve fällt minimal langsamer ab.

Abbildung 6: EPSP mit Parametern drvifall=1.5, drviout=1.6

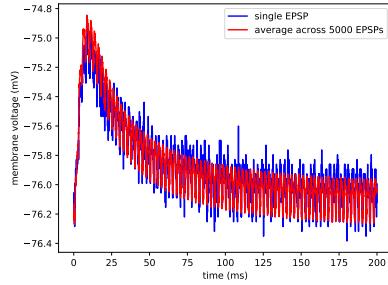


Abbildung 7: EPSP mit Parametern drvifall=0.1, drviout=1.6

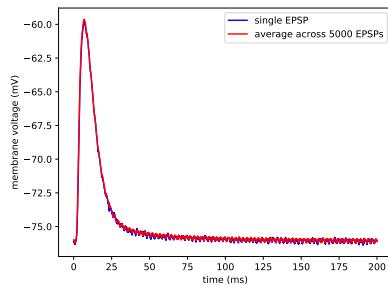
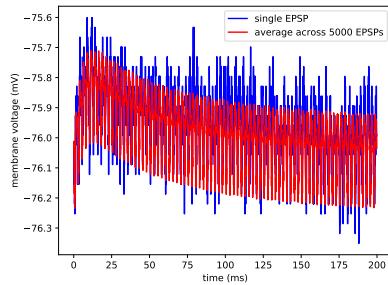


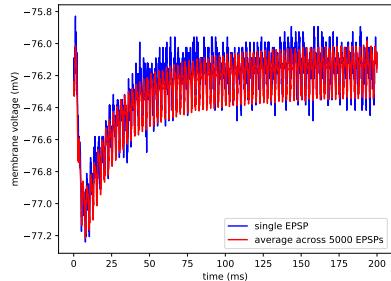
Abbildung 8: EPSP mit Parametern drvifall=1.5, drviout=0.3



3.2 Aufgabe 2

Wie erwartet ist der Peak umgekehrt. Das Verhalten der beiden Parameter drvifall und drviout ist hingegen unverändert im Vergleich zum EPSP.

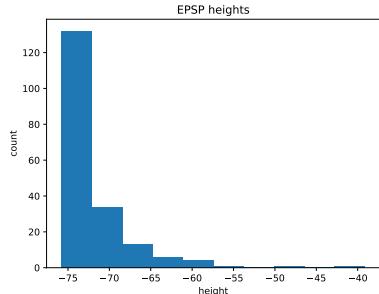
Abbildung 9: IPSP mit Parametern drvifall=1.5, drviout=1.6



3.3 Aufgabe 3

Wir untersuchen das fixed-pattern noise zwischen verschiedenen Synapsen. Hierzu werden von 192 Synapsen gemittelte EPSP-Läufe aufgenommen und deren Höhe bestimmt. Die Verteilung ist als Histogramm in Abbildung ?? zu sehen. Die berechnete Varianz der EPSP-Höhen beträgt $\sigma^2 = 23.14mV^2$.

Abbildung 10: Histogramm der EPSP-Höhen von 192 Synapsen

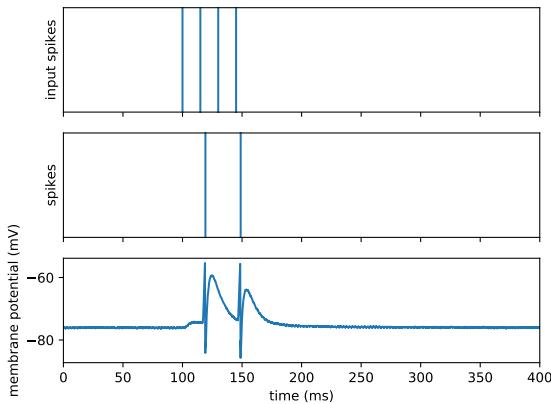


3.4 Aufgabe 4

Wir verwenden eine andere Stimuluserzeugung zur Beobachtung gestapelter EPSPs und reduzieren den zeitlichen Abstand zwischen den Eingangs-Spikes bis das Neuron mindestens einmal feuert. In einem festen Zeitabstand werden Input-Spikes erzeugt, bis die Summe der Potentiale den Schwellwert (Threshold) übersteigt und das Neuron feuert. Ist der Zeitabstand zwischen zwei Input-Spikes zu groß, so kommt es zu keiner zeitlichen Summation der Signale.

Vergleich der relativen Höhen der verschiedenen PSPs qualitativ mit den vorherigen Ergebnissen des fixed-pattern noise: Der Peak des Membranpotentials hat einen höheren Wert wie die resultierenden PSP-Höhen in Aufgabe 3. Dies liegt daran, dass die Eingangs-Spikes gestapelt werden. Unterschiedliche Peak-Höhen zeigen außerdem das fixed pattern noise sehr deutlich.

Abbildung 11: Anregungen durch verschiedene Synapsen. Unterschiedliche Peak-Höhen zeigen das fixed pattern noise sehr deutlich.



4 Kurzzeit-Plastizität

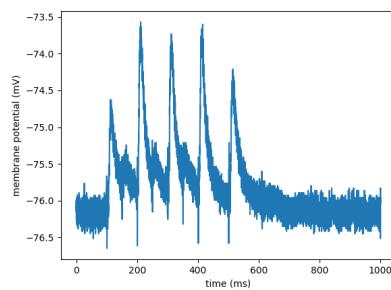
In dieser Aufgabe untersuchen wir die Hardware-Implementierung der Kurzzeitplastizität (STP).

4.1 Aufgabe 1

Wir starten mit dem depressing mode ($\tau_{fac} = 0$). Wir beobachten folgendes Verhalten:

- Verkleinert man die die spike distance, desto schneller erfolgt der Abfall des ersten Peaks zu einem stationären Plateau und je keiner die spike distance, desto größer der Wert dieses Plateaus. Dieses Verhalten liegt nahe, wurde aber nicht beobachtet, da wir anfänglich τ_{rec} zu klein gewählt haben. Hier spielen zwei konträre Effekte eine Rolle: Durch eine kürzere spiking distance ist zwar jeder einzelne Peak kleiner (da stärker unterdrückt), aber durch Summation ist das Gesamtpotential wieder größer.
- Je größer die final spike distance, desto größer ist der letzte Peak (weniger stark depressed).
- Vergrößert man die Utilization U so werden die Werte der PSP-Höhen größer und fallen schneller ab.
- Umso größer man τ_{rec} wählt, desto stärker wird der letzte Peak unterdrückt und desto schneller fällt die Amplitude auf den stationären Fall ab.

Abbildung 12: spike distance: 50.0ms, final spike distance: 100.0ms



4 Kurzzeit-Plastizität

Abbildung 13: spike distance: 100.0ms, final spike distance: 300.0ms

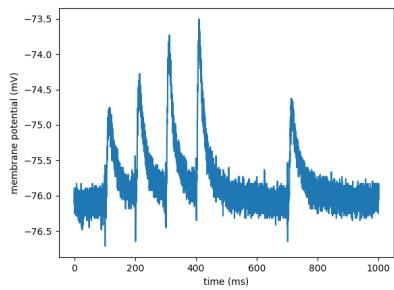


Abbildung 14: spike distance: 10.0ms, final spike distance: 300.0ms

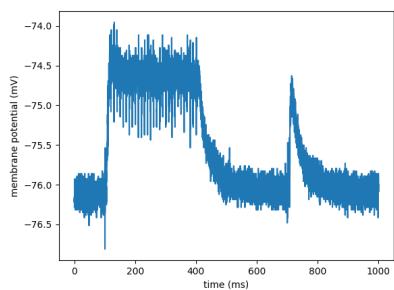
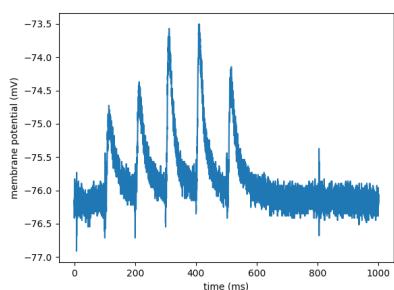


Abbildung 15: spike distance: 100.0ms, final spike distance: 100.0ms



4 Kurzzeit-Plastizität

Abbildung 16: spike distance: 100.0ms, final spike distance: 200.0ms

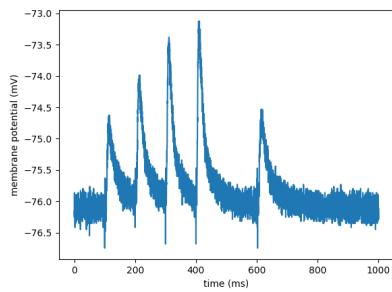


Abbildung 17: spike distance: 100.0ms, final spike distance: 500.0ms

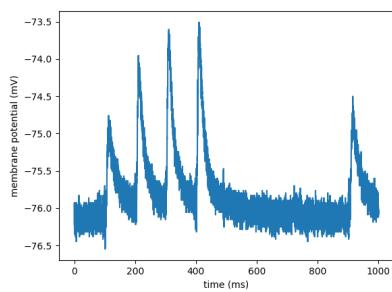
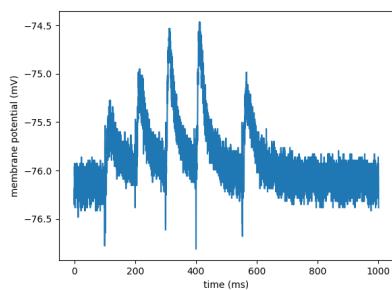


Abbildung 18: U: 0.2, spike distance: 100.0ms, final spike distance: 150.0ms



4 Kurzzeit-Plastizität

Abbildung 19: U: 0.8, spike distance: 100.0ms, final spike distance: 150.0ms

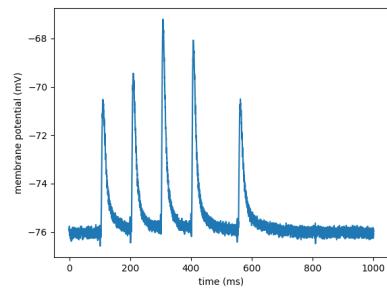


Abbildung 20: τ_{rec} : 100.0ms, final spike distance: 200.0ms, U: 0.6, spike distance: 100.0ms

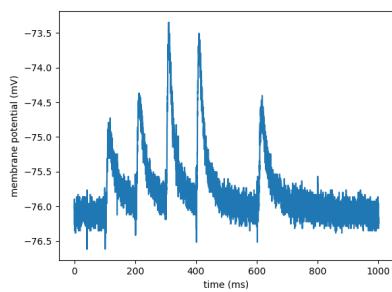


Abbildung 21: τ_{rec} : 150.0ms, final spike distance: 200.0ms, U: 0.6, spike distance: 100.0ms

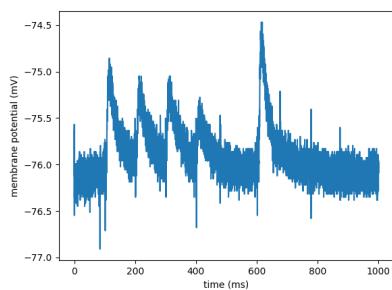
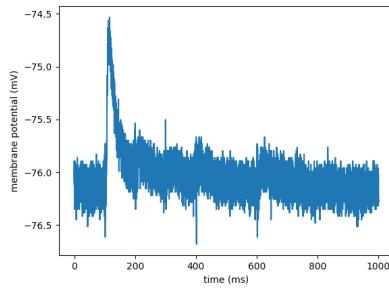


Abbildung 22: τ_{rec} : 200.0ms, final spike distance: 200.0ms, U: 0.6, spike distance: 100.0ms



4.2 Aufgabe 2

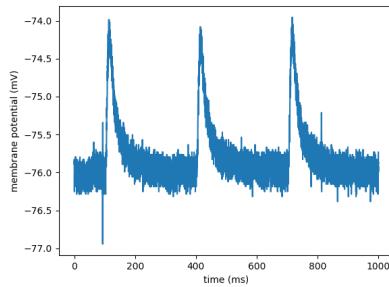
Wird STP deaktiviert, so ist der erste Peak kleiner als die nachfolgenden.

4.3 Aufgabe 3

Wir schalten in den falilating mode ($\tau_{rec} = 0$) und untersuchen analog das Verhalten des Membranpotentials. Wir beobachten folgendes Verhalten:

- Verringert man die spike distance, so erhält man durch Summation der einzelnen Peaks einen höheren Wert für die Peaks des Membranpotentials.
- Je kleiner die final spike distance, desto größer ist der letzte Peak, da er noch stärker begünstigt wird.
- Verkleinert man die Utilization U , so werden die Werte der PSP-Höhen größer und fallen schneller ab.
- Umso größer man τ_{fac} wählt, desto stärker wird der letzte Peak verstärkt.

Abbildung 23: spike distance: 300.0ms, τ_{fac} : 100.0ms, U: 0.8, final spike distance: 300.0ms



4 Kurzzeit-Plastizität

Abbildung 24: spike distance: 100.0ms, τ_{fac} : 100.0ms, U: 0.8, final spike distance: 300.0ms

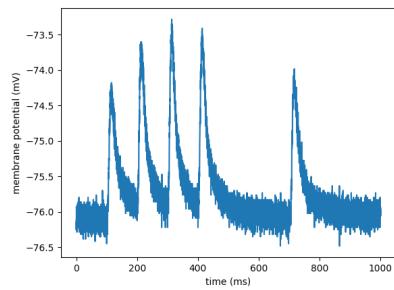


Abbildung 25: final spike distance: 100.0ms, τ_{fac} , 150.0, U: 0.6, spike distance: 100.0ms

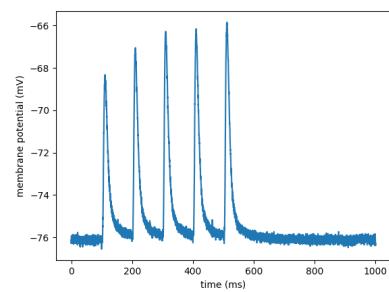
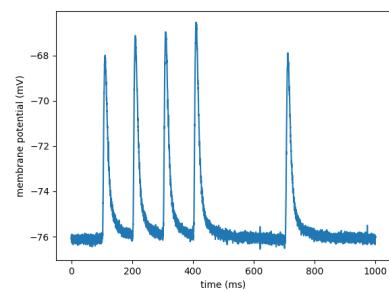


Abbildung 26: final spike distance: 300.0ms, τ_{fac} : 150.0ms, U: 0.6, spike distance: 100.0ms



4 Kurzzeit-Plastizität

Abbildung 27: final spike distance: 500.0ms, τ_{fac} : 150.0ms, U: 0.6, spike distance: 100.0ms

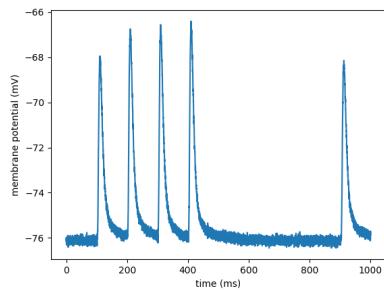


Abbildung 28: U: 0.4, τ_{fac} : 100.0ms, spike distance: 100.0ms, final spike distance: 300.0ms

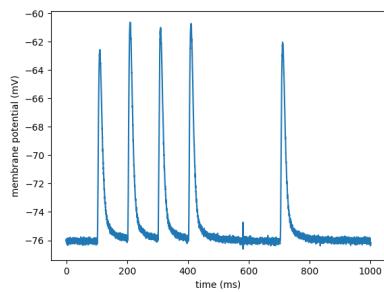
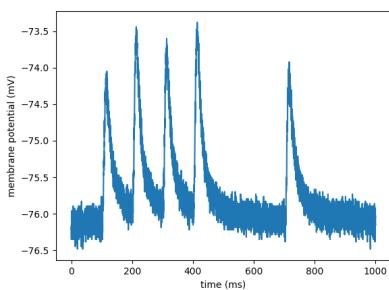


Abbildung 29: U: 0.8, τ_{fac} : 100.0ms, spike distance: 100.0ms, final spike distance: 300.0ms



4 Kurzzeit-Plastizität

Abbildung 30: τ_{fac} : 50.0ms, U: 0.6, spike distance: 100.0ms, final spike distance: 300.0ms

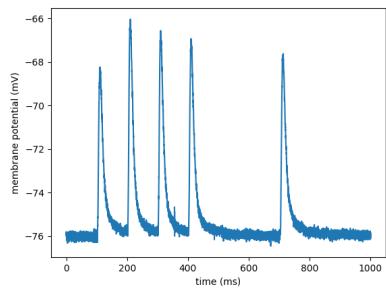
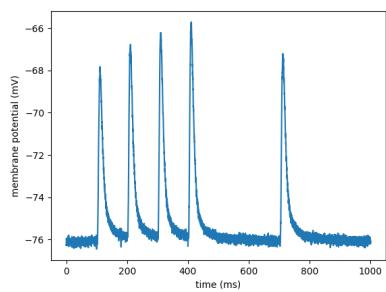


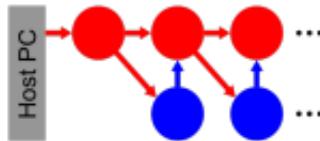
Abbildung 31: τ_{fac} : 200.0ms, U: 0.6, spike distance: 100.0ms, final spike distance: 300.0ms



5 Feed-Forward Netzwerke

Wir untersuchen ein neuronales Netz am Beispiel einer vorwärts gerichteten Synfire Chain. Der Aufbau erfolgt gemäß Abbildung ???. Die Kettenglieder sind Populationen und bestehen aus einer gewissen Anzahl an Neuronen. Jede Population ist jeweils mit der nächsten Population verbunden. Eine Gruppe (Spalte) feuert dabei immer synchron. Die inhibitorischen Neuronen verhindern ein Doppelfeuern der exzitatorischen.

Abbildung 32: Schematische Darstellung einer Synfire-Kette mit Feed-Forward-Hemmung. Erregende und hemmende Neuronen sind rot bzw. blau gefärbt



5.1 Aufgabe 1

Das Feuerverhalten der Synfire-Chain hängt maßgeblich von den synaptischen Gewichten ab. Um das in Abbildung ?? dargestellte Verhalten zu produzieren, haben wir folgende Gewichte verwendet:

Tabelle 5: Verbindungsgewichte Synfire-Chain

Gewicht	Wert
StimExcExc	$2 \cdot \text{pynn.minExcWeight}()$
StimExcInh	$2 \cdot \text{pynn.minExcWeight}()$
ExcExc	$2 \cdot \text{pynn.minExcWeight}()$
ExcInh	$1 \cdot \text{pynn.minExcWeight}()$
InhExc	$15 \cdot \text{pynn.minInhWeight}()$

Die exc-exc scheint die sensitivste Verbindung zu sein, da hier kleine Veränderungen des Gewichts einen sehr hohen Einfluss auf das Feuerverhalten der Synfire-Chain hatten. Die inhibitorischen Neuronen kann man mit folgenden Gewichtseinstellungen deaktivieren:

Tabelle 6: Verbindungsgewichte bei deaktivierter Inhibition

Gewicht	Wert
StimExcExc	$2 \cdot \text{pynn.minExcWeight}()$
StimExcInh	$0 \cdot \text{pynn.minExcWeight}()$
ExcExc	$2 \cdot \text{pynn.minExcWeight}()$
ExcInh	$0 \cdot \text{pynn.minExcWeight}()$
InhExc	$0 \cdot \text{pynn.minInhWeight}()$

Mit diesen Gewichten erhält man das in Abbildung ?? dargestellte Verhalten. Man sieht, dass die Hälfte aller Neuronen nun nicht mehr feuert sowie eine Auffächerung: Eine größere Anzahl an Neuronen feuert nun mehrfach.

Abbildung 33: Aktivität der Synfire-Kette einschließlich des Membranpotenzials des Neurons mit ID=0

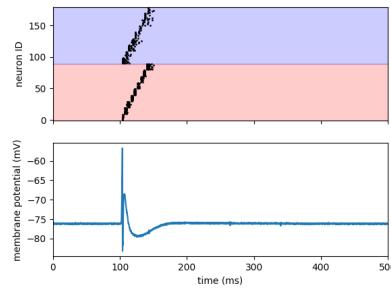
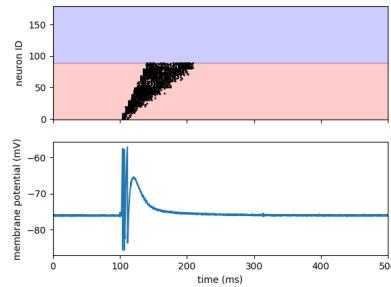


Abbildung 34: Aktivität der Synfire-Kette mit deaktivierter Inhibition



5.2 Aufgabe 2

Mit einer kleineren Anzahl an Neuronen je Population kann die Kettenlänge und die Dauer der Netzwerkaktivität maximiert werden. Eine maximale Dauer ergibt sich mit zwei exhibitorischen und einem inhibitorischen Neuron je Population bei einer Gesamtlänge von 64 Populationen. Durch Anpassen der Parameter wurde versucht der Auffächerung entgegenzuwirken. Das beste Ergebnis konnte mit folgenden Einstellungen erzielt werden und ist in Abbildung ?? dargestellt.

Tabelle 7: *Synfire-Chain maximaler Länge*

Gewicht	Wert
StimExcExc	$5 \cdot \text{pynn.minExcWeight}()$
StimExcInh	$4 \cdot \text{pynn.minExcWeight}()$
ExcExc	$3 \cdot \text{pynn.minExcWeight}()$
ExcInh	$2 \cdot \text{pynn.minExcWeight}()$
InhExc	$15 \cdot \text{pynn.minInhWeight}()$

Die starke Auffächerung der Kette lässt sich durch die starke Anregung bei fehlenden inhibitorischen Zellen erklären. Bis zum Ende der Refraktärzeit ist das anregende EPSP an der Zelle noch nicht abgeklungen, sodass diese nochmals feuert.

5.3 Aufgabe 3

Wir schließen die Synfire-Chain, indem wir die letzte Population an Neuronen mit der ersten verbinden. Das beobachtete Verhalten ist in Abbildung ?? dargestellt.

Abbildung 35: Synfire-Chain maximaler Länge

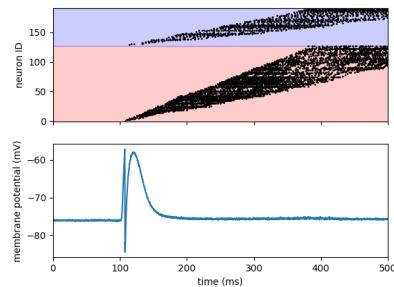
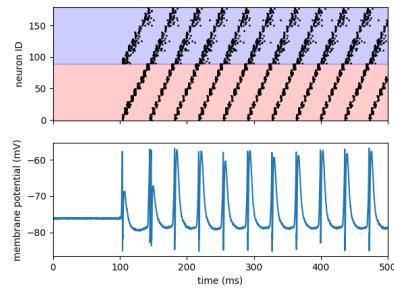
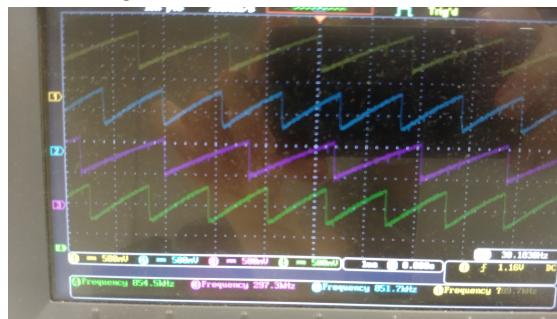


Abbildung 36: Aktivität der Synfire-Kette mit in einer Loop-Konfiguration, in der die letzte Population die erste stimuliert



In Abbildung ?? sieht man die zeitliche Differenz der eintreffenden PSPs auf den Membranen, woraus man gut das Timing des eintreffenden erregenden Stimulus und der Feedforward-Hemmung beobachten kann.

Abbildung 37: PSPs von 4 Hardware-Neuronen



6 Rekurrente Netzwerke

In diesem Versuchsteil untersuchen wir den Einfluss von zufälligen Verbindungen auf die neuronale Aktivität. Wir versetzen eine Population von einigen Neuronen in einen kontinuierlich feuernden Zustand.

6.1 Aufgabe 1

Abbildung 38: Feuerrate in Abhängigkeit der CVs für die ISI

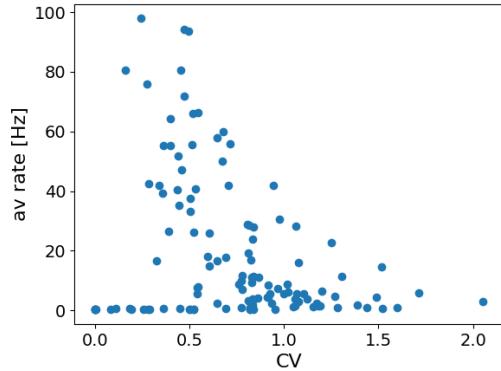
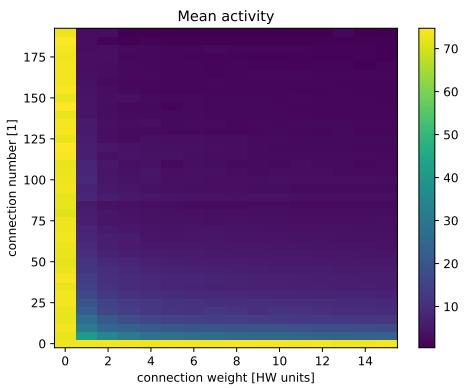


Abbildung 40: Mittlere Aktivität



6.3 Aufgabe 3

Schließlich wurde versucht die Feuerrate auf 25 Hz zu bringen. Das haben wir mit folgenden Paramtern am besten geschafft: $\omega = 2$ und $K = 10$. Bei Verwendung dieser Paramter erhielten wir eine Feuerrate von 24.8 Hz.

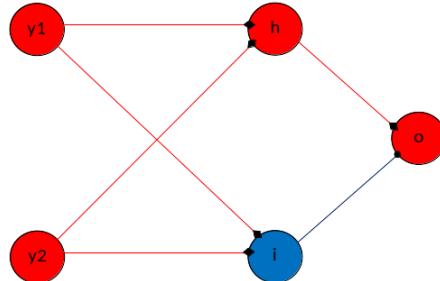
7 Eine einfache Berechnung - XOR

Im letzten Versuchsteil soll ein XOR-Gatter implementiert werden.

7.1 Aufgabe 1

Wir nehmen an, dass alle Neuronen funktionieren und jeder Eingangsimpuls löst genau eine Ausgangsimpuls aus. In einem perfekten System bräuchte man die hemmenden Neuronen also nicht, um zu verhindern, dass es ein Doppelfeuern geben kann. Man bräuchte sie nur, um zu verhindern, dass ein Signal weitergegeben wird, wenn beide Input-Neuronen y_1 und y_2 feuern.

Abbildung 41: Simples XOR-Gate



7.2 Aufgabe 2

Die Verbindungen zwischen der y und h -Schicht sind die sensitivsten Verbindungen. Hier müssen die inhibitorischen Neuronen, die auf h_1 und h_2 wirken, das Signal der exhibitorischen Neuronen perfekt unterdrücken, falls dieses auch Feuern. Durch eine größere Population könnten wir das Netzwerk robuster machen.

7.3 Aufgabe 3

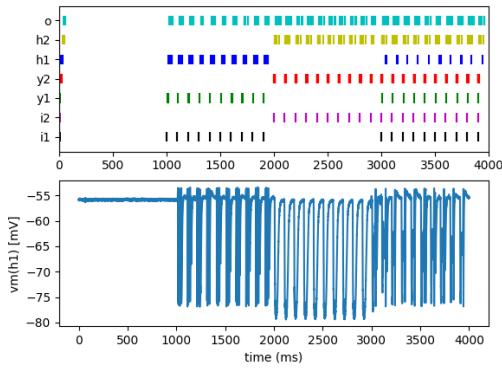
Mit den Paramtern

Tabelle 8: Gewichte für XOR-Gatter

Gewicht	Wert
I2Yw	2
Y2Hw	2
Y2Hi	15
H2Ow	4

erhalten wir das in ?? dargestellte Verhalten. Leider konnten wir nicht die richtigen Paramter finden um das richtige Verhalten eines XOR-Gatters zu reproduzieren. Wie man an Hand der Abbildung erkennt, scheint das Problem bei dem Gewicht $Y2Hi$ zu liegen. Sowohl bei y_1 als auch bei y_2 kommt ein Input-Spike an und diese beiden Spikes werden direkt an h_1 und h_2 (und damit an o) weitergereicht. Die Unterdrückung dieser beiden Spikes scheint mit unserem gewählten Parameter-Setting nicht zu funktionieren.

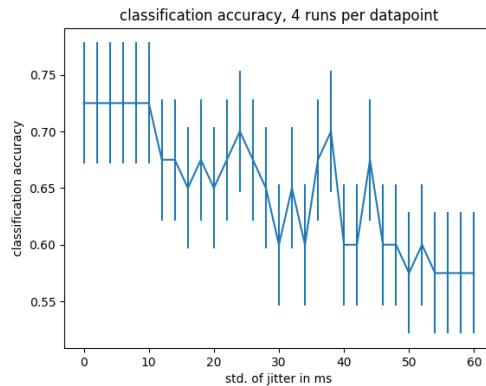
Abbildung 42: Verhalten des XOR-Gates mit Gewichten aus Tabelle 8



7.4 Aufgabe 4

Im letzten Aufgabenteil haben wir die Klassifizierungsrate (classification rate) mit Jitter (Störungen in der Frequenz) untersucht.

Abbildung 43: XOR-Gate mit Jitter



Man erkennt, dass mit höheren Jitter-Werten die Genauigkeit sinkt. Die Ursache hierfür sind verzögerte Signale, die den Sperrmechanismus verursacht durch die inhibitorischen Neuronen umgehen. Eine Lösung wäre, mehrere Signale auszusenden und das am häufigsten beobachtete Ereignis zu nehmen. Diesen Vorgang könnte man auch in den Versuch mitaufnehmen.