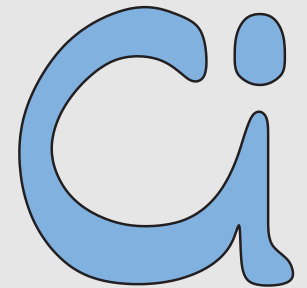


Neuronale Netze

Prof. Dr. Rudolf Kruse
Christoph Doell, M.Sc.
Christian Braune, M.Sc.

Computational Intelligence
Institut für Wissens- und Sprachverarbeitung
Fakultät für Informatik
kruse@iws.cs.uni-magdeburg.de



Spiking Neural Networks

Spiking Neural Networks

Deutsch: gepulste neuronale Netze

Momentan: die am häufigsten diskutierten KNN-Modelle

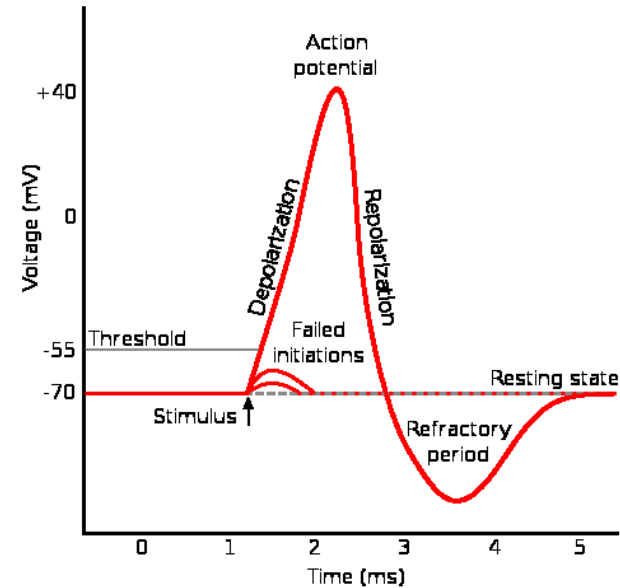
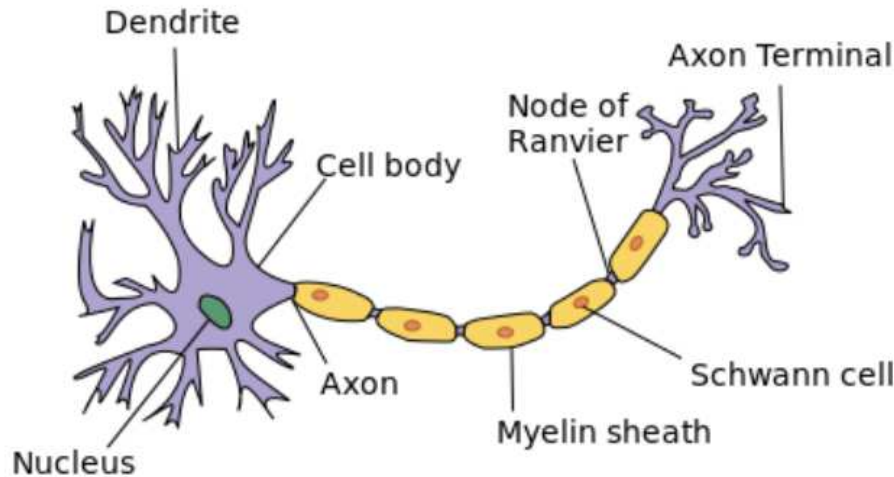
Erhöhen Realismus der neuronalen Simulation: berücksichtigen neben neuronalem und synaptischem Zustand auch Zeit

Idee: Neuronen eines SNN feuern *nicht* wie z.B. beim *MLP* bei jedem Propagationszyklus

Verschiedene Kodierungsmodelle, um Aktionspotential (*Spike Train*) als Folge zu behandeln: basierend auf Frequenz der Spikes oder Zeitspanne zwischen Spikes

Ziel: Mit welchem Kode (welcher Sprache) kommunizieren Neuronen?

Messung von Pulsen



Neuronales Signal besteht aus kurzen elektrischen Pulsen

Mögliche Beobachtung mittels feiner Elektrode nahe des Zellkörpers oder des Axons

Pulse (*Aktionspotentiale/Spikes*): Amplitude von 100mV und Dauer von 1-2ms

Aussehen des Pulses ändert sich bei der Propagation entlang des Axons nicht.

Spike Trains

Kette von Aktionspotentialen eines einzelnen Neurons heißt *Spike Train*

Sequenz stereotypischer Ereignisse, die in regelmäßigen Intervallen eintreten

Da alle Spikes eines Neurons gleich aussehen, enthält Form des Aktionspotentials keine Information

Vermutung: Anzahl und Zeitpunkte der Spikes spielen eine Rolle

Spikes eines Trains sind meistens leicht voneinander zu trennen

Synapsen und Neuronale Dynamik

Synapse: Ort, wo Axon eines (präsynaptischen) Neurons Dendrit/
Zellkörper eines (postsynaptischen) Neurons berührt

Ankunft eines Aktionspotentials führt zu biochemischer Reaktion

Rezeptoren der Synapse registrieren chemisches Signal und wandeln
es in elektrischen Impuls um

Effekt eines Spikes auf postsynaptisches Neuron kann mittels
Elektrode gemessen werden

sie misst Potentialdifferenz $u(t)$ (Membranpotential) zwischen
Zellinnerem und -äußerem

Ohne Spike: Neuron "ruht" bei konstantem Potential ($\approx 65mV$)

Mit Spike: Potential ändert sich und kehrt zurück in Ruhezustand

Bei positivem Wechsel: erregend (exzitatorisch),
bei negativem: hemmend (inhibitorisch)

Mathematische Beschreibung

Gegeben: Zeitreihe $u_i(t)$ des Membranpotentials vom Neurons i

In Ruhe: $u_i(t) = u_{rest}$

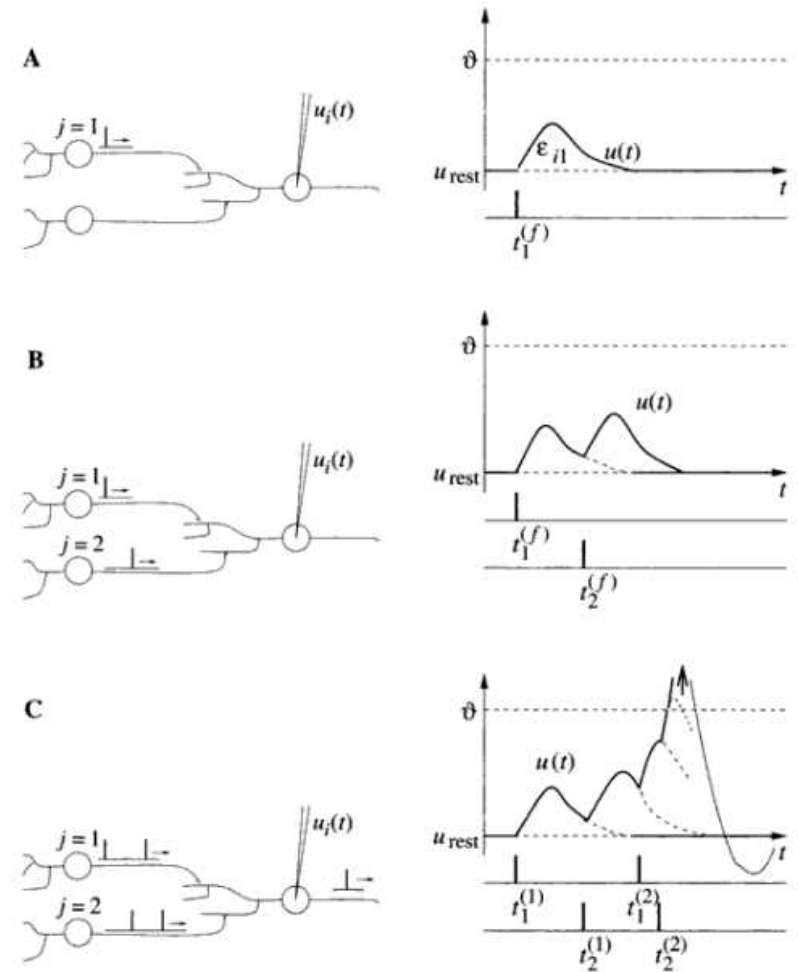
Bei $t = 0$ feuert präsynaptisches Neuron j

Für $t > 0$, Antwort von i :

$$u_i(t) - u_{rest} = \epsilon_{ij}(t)$$

$\epsilon_{ij}(t)$: postsynaptisches Potential

Falls $\epsilon_{ij}(t) > 0$, dann exzitatorisch (EPSP),
sonst inhibitorisch (IPSP)



Schwellenwert und Aktionspotential

Nun zwei präsynaptische Neuronen $j = 1, 2$, die beide Spikes an i senden

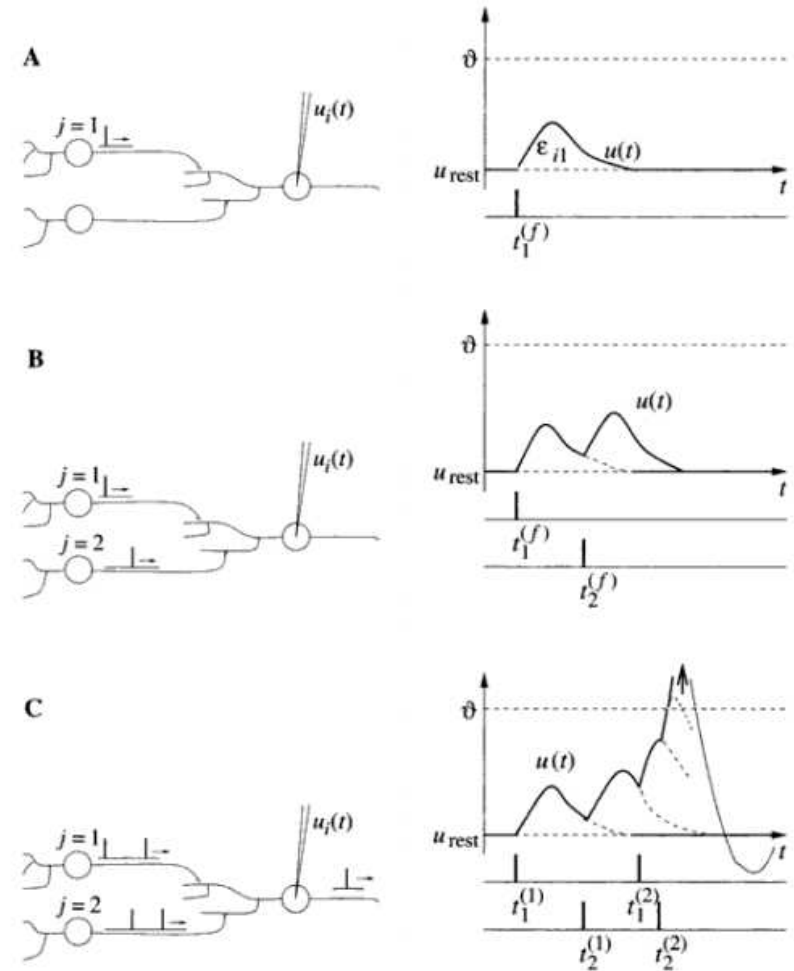
Neuronen j feuern bei $t_j^{(1)}, t_j^{(2)}$

Jeder Spike evoziert ein PSP ϵ_{ij}

Mit wenigen Spikes, Änderung des Potentials:

$$u_i(t) = \sum_j \sum_f \epsilon_{ij}(t - t_j^f) + u_{rest}$$

Bei vielen Spikes in kurzer Zeit: Linearität verschwindet



Schwellenwert und Aktionspotential

Bei Erreichen des kritischen Wertes θ wird Aktionspotential (100mV) in i angeregt

Dieses wird über Axon an andere Neuronen weitergeleitet

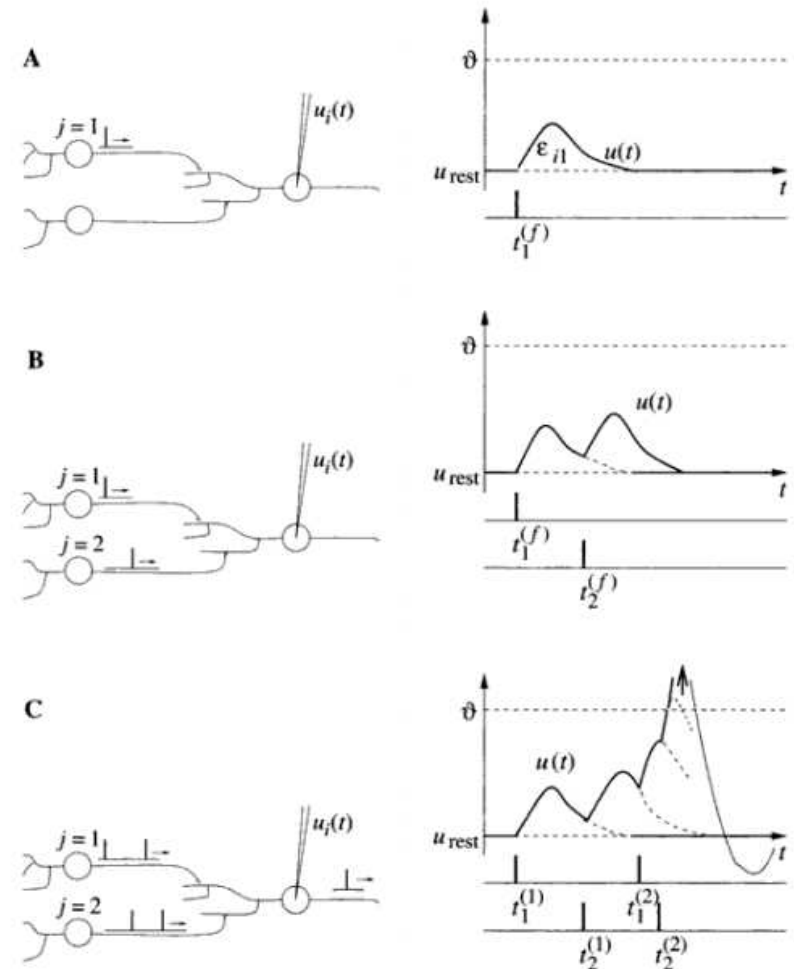
Nach Puls: $u_i(t) < u_{rest}$

Einzelne EPSPs haben 1mV

θ ist ca. 20-30mV über u_{rest}

4 Spikes reichen normalerweise nicht für postsynaptisches Aktionspotential

Normalerweise ca. 20-50 Spikes in kurzer Zeit erforderlich



Phänomenologisches Neuronenmodell

Kritische Spannung für Aktionspotential: Schwellenwert θ

Wenn $u_i(t) \geq \theta$, dann feuert i einen Spike zum Zeitpunkt $t_i^{(f)}$

Annahme: Spikes haben stets ähnliches Aussehen

Trajektorie des Membranpotentials während Spike kann deswegen über zeitlichen Verlauf $\eta(t - t_i^{(f)})$ beschrieben werden

Definition eines einfachen Modells

Variable u_i beschreibt momentanen Wert des Membranpotentials des Neurons i

$$u_i(t) = \eta(t - \hat{t}_i^{(f)}) \sum_j \sum_f \epsilon_{ij}(t - t_i^{(f)}) + u_{rest}$$

$\hat{t}_i^{(f)}$: letzte Feuerzeit von i , also

$$\hat{t}_i^{(f)} = \max\{t_i^{(f)} \mid t_i^{(f)} < t\}$$

Es wird immer dann gefeuert, wenn u_i Wert θ überschreitet

$$u_i(t) = \theta \text{ und } \frac{d}{dt}u_i(t) > t \Rightarrow t = t_i^{(f)}$$

ϵ_{ij} beschreibt Antwort von i

η beschreibt Aussehen des Spikes

Definition eines einfachen Modells

$$u_i(t) = \eta(t - \hat{t}_i^{(f)}) \sum_j \sum_f \epsilon_{ij}(t - t_i^{(f)}) + u_{rest}$$

mit

$$\hat{t}_i^{(f)} = \max\{t_i^{(f)} \mid t_i^{(f)} < t\}$$

Wichtig hier: es interessiert nur Potential*differenz* („Abstand“ zu u_{rest})

Mittels Vereinfachung $u_{rest} := 0$: $u(t)$ ist Abstand zum Ruhepotential

Dieses Modell nennt man SRM₀ (Spike Response Model), (Gerstner, 1995)

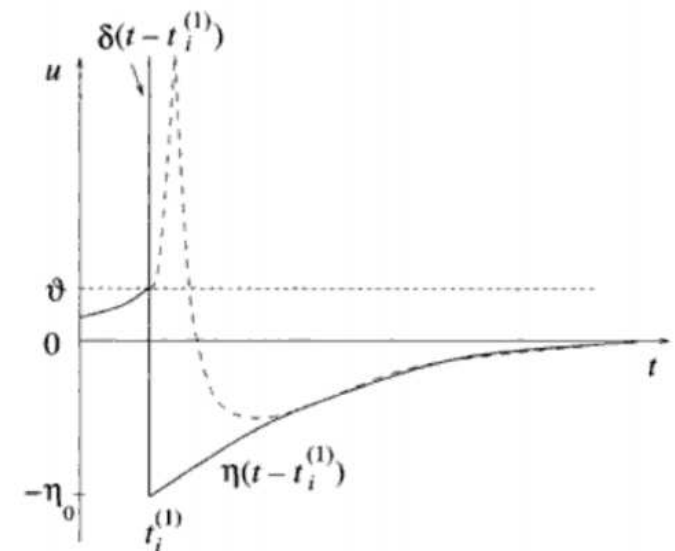
Beispiel: Formeller Pulse

η kann beispielsweise einen Rechteckpuls beschreiben, auf den ein negatives Spike Potential folgt:

$$\eta(t - t_i^{(f)}) = \begin{cases} 1/\Delta t & \text{für } 0 < t - t_i^{(f)} < \Delta t \\ -\eta e^{-\left(\frac{t-t_i^{(f)}}{\tau}\right)} & \text{für } \Delta t < t - t_i^{(f)} \end{cases}$$

η_0, τ und $\Delta t > 0$ sind freie Parameter

für $\Delta t \rightarrow 0$ nähert sich der Rechteckpuls dem Diracpuls δ an (siehe Abb.)



Formeller Spike Train

Feuerzeit: Zeitpunkt, wann Neuron Aktionspotential emittiert

Feuerzeiten des Neurons i : $t_j^{(f)}$ mit $f = 1, 2, \dots$

Rein formell: Spike Train S von i ist Sequenz der Feuerzeiten

$$S_i(T) = \sum_f \delta(t - t_i^{(f)})$$

$\delta(x)$ ist Dirac-„Funktion“ mit

$$\delta(x) = 0 \text{ für } x \neq 0 \text{ und } \int_{-\infty}^{+\infty} \delta(x) dx = 1$$

Spikes werden somit zu Zeitpunkten reduziert

Einschränkungen des Modells SRM₀

SRM₀ ist sehr stark vereinfacht

Es ignoriert viele Aspekte der neuronalen Dynamik

Ganz besonders: postsynaptische Potentiale haben alle gleiches Aussehen unabhängig vom Zustand des Neurons

Weiterhin: Dynamik des Neurons i hängt nur von seiner letzten Feuerzeit \hat{t}_i ab

Ausgefeiltere Modelle existieren, werden in dieser Vorlesung nicht behandelt

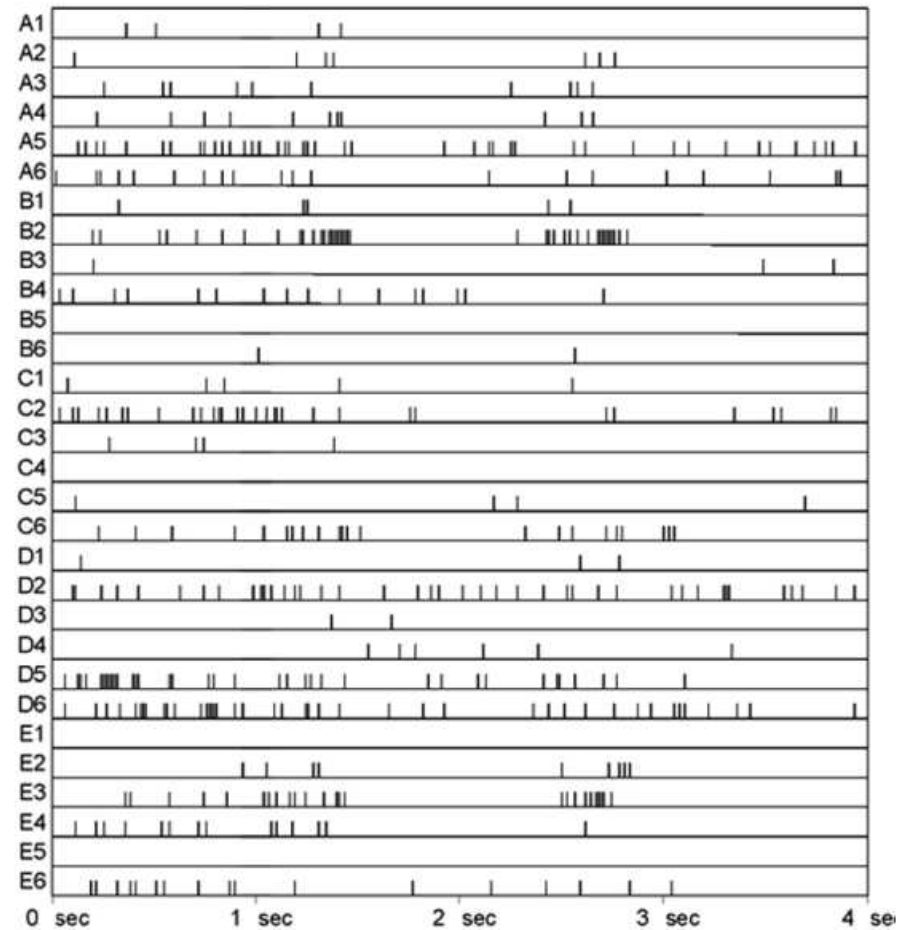
Das Problem der neuronalen Kodierung

Gehirne von Säugetieren beinhalten mehr als 10^{10} dicht gepackte Neuronen

Sie sind untereinander zu einem komplizierten Netzwerk verschaltet

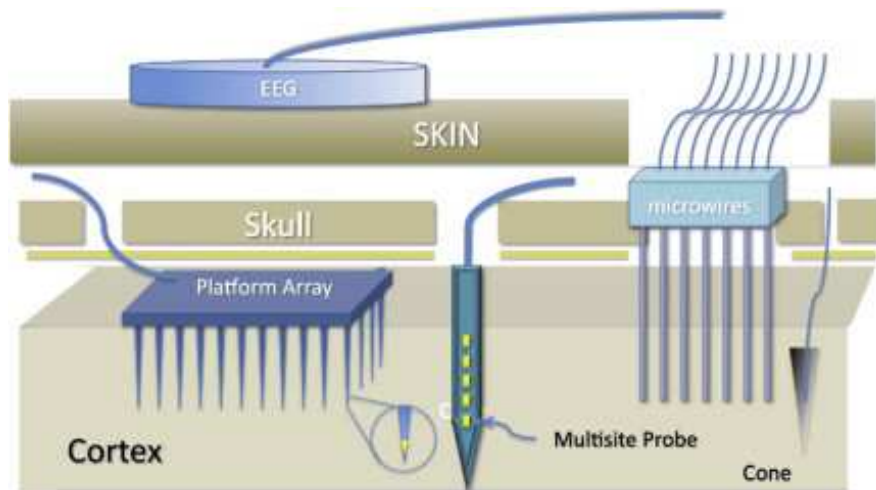
In jedem noch so kleinen Volumen des Kortex werden Tausende von Spikes pro Millisekunde emittiert

Welchen Code nutzen Neuronen zur Informationsübertragung?



Spike Train von 30 Neuronen im Affenkortex

Multi-Elektroden Aufnahmegeräte [Borgelt *et al.*, 2013]

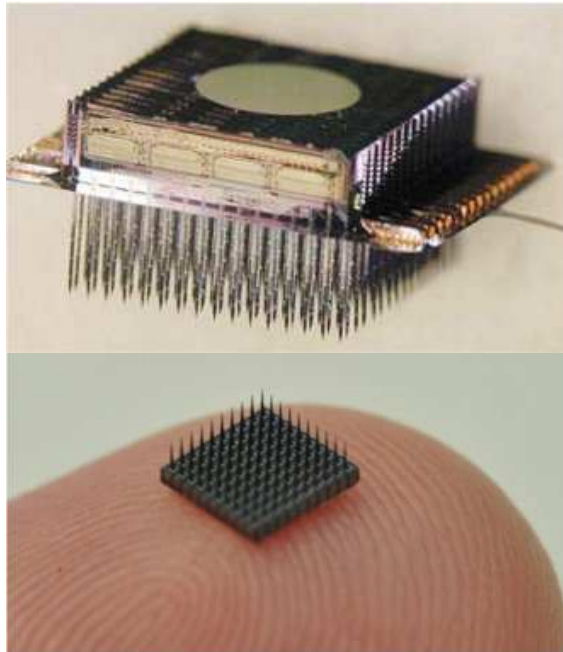


Quelle: John P. Donoghue

Unterschiedliche Arten von Aufnahme-geräten wurden in den letzten Jahren entwickelt und sind heutzutage im Einsatz.

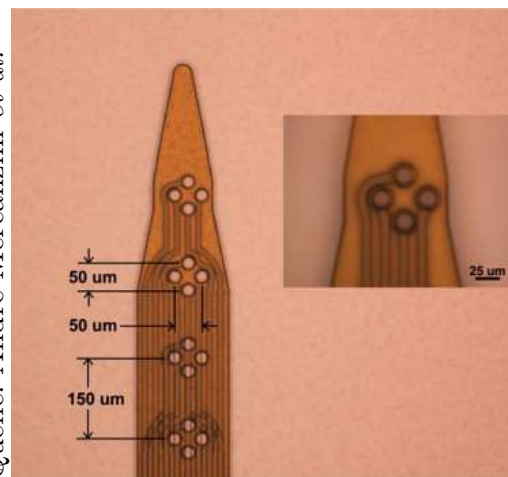
Nachteil dieser Geräte:
Sie müssen implantiert werden.

Vorteile: Sehr hohe Zeit- und Raum-
auflösung, kleinste Schwankungen im
elektr. Potential können erkannt werden.

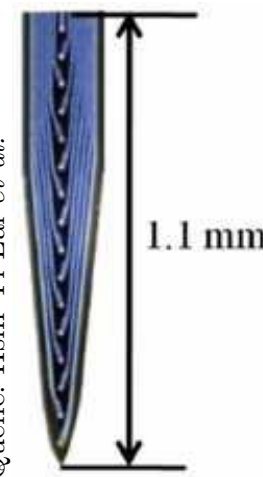


Quellen: Je-Min Yoo *et al.*; J.S. Brumberg *et al.*

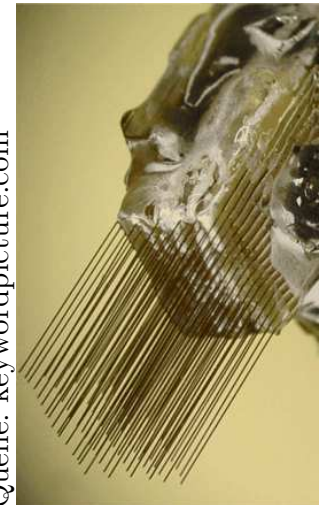
Quelle: André Mercanzini *et al.*



Quelle: Hsin-Yi Lai *et al.*



Quelle: keywordpicture.com



Generative Modelle

Bisherige Betrachtung (SRM₀) rein deskriptiv

Gewinnung von Daten durch MEAs nur *in vivo* möglich

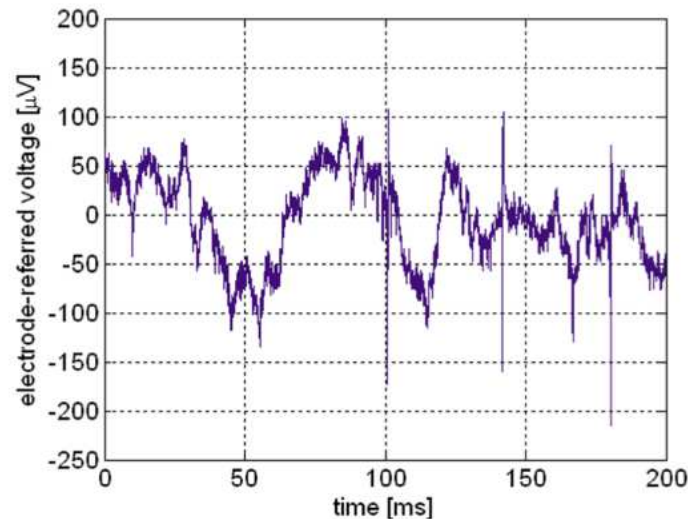
⇒ mögliche ethische Probleme

Alternativ (vgl. Potter *et al.*): *in vitro* Aufnahmen: embryonale Hirnzellen (zumeist von Ratten) werden in Nährlösung direkt auf den Elektroden gezüchtet

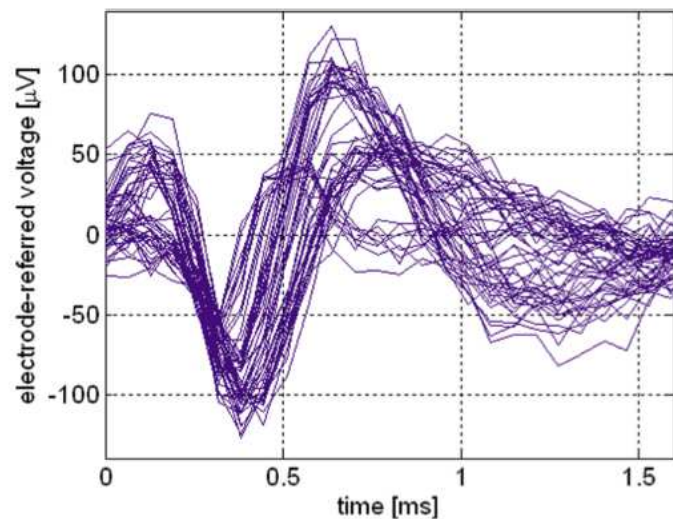
Feuernde Neuronen (eingefärbt, Farbstoff reagiert auf das Kalzium in den Neuronen)



Spike Sorting



source: Reid R. Harrison



source: Reid R. Harrison

Eine echte Aufnahme des elektrischen Potentials beinhaltet auch das lokale Potential (*local field potential*), das von benachbarter Aktivität bestimmt wird.

Das LFP kann durch einen Hochpassfilter ($\approx 300\text{Hz}$) entfernt werden

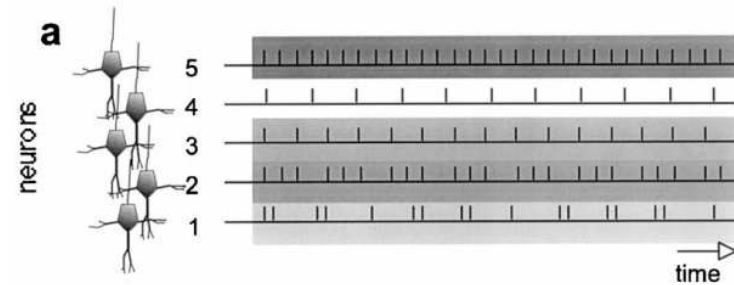
Spikes werden durch Schwellwertüberschreitung detektiert.

Richtet man alle Spikes aus (siehe untere Abb.), können unterschiedliche Neuronen erkannt werden, die von der selben Elektrode aufgenommen wurden.

Diesen Vorgang nennt man **spike sorting**.

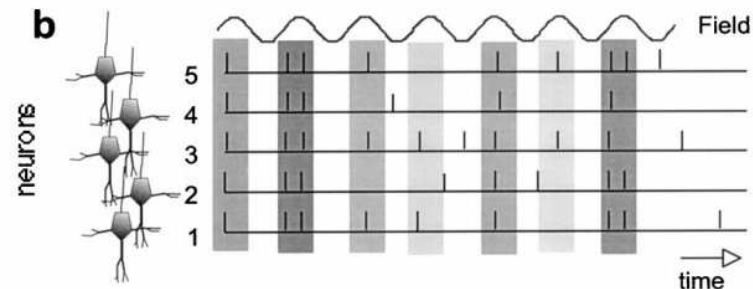
Frequency Coding

Neuronen generieren Spikes mit unterschiedlicher Frequenz als Reaktion auf unterschiedliche Stimuli; je stärker der Stimulus, desto höher die Frequenz.



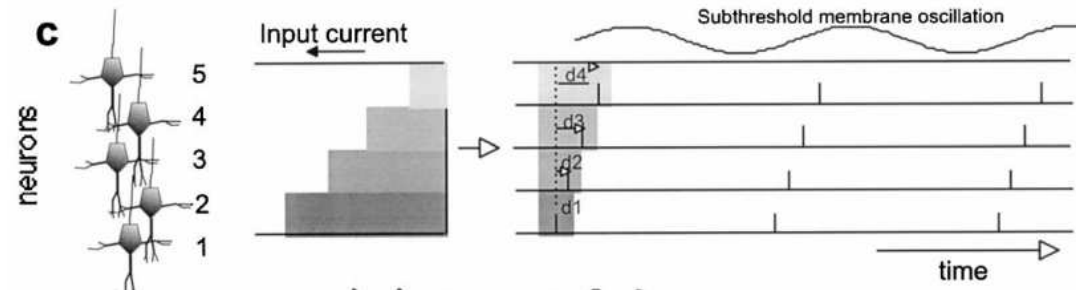
Temporal Coincidence Coding

Je enger Koinzidenzen beieinander liegen, desto höher ist die Stimulusintensität. Neuronen sind in Ensembles organisiert, die koinzident feuern.



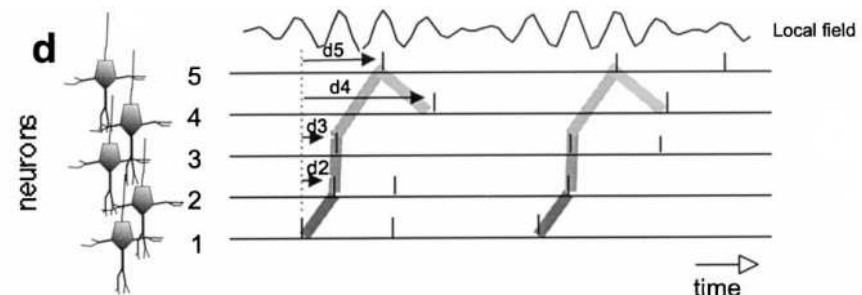
Delay Coding

Der Eingabestimulus wird in ein Verzögerungssignal umgewandelt (möglicherweise relativ zu einem Referenzsignal). Ein stärker stimuliertes Neuron erreicht den Depolarisierungsschwellwert eher und emittiert einen Spike früher.



Spatio-Temporal Coding

Neuronen emittieren Spikes in einer bestimmten Reihenfolge als Antwort auf einen Stimulus. Die Sequenz wird sowohl durch temporale als auch lokale Eigenschaften des Stimulus sowie die Struktur der Netzwerkes bestimmt.



Simulation von Spike Trains

Einfachstes Modell: *Poisson-Prozess*

Feuerzeiten von i hängen nur von der verstrichenen Zeit seit letztem Spike ab (wie bei SRM₀) und der Feuerrate des Neurons (in Hz)

Interspike-Intervalle sind exponential verteilt: $f(x; \lambda) = \begin{cases} \lambda e^{-\lambda x} & x \geq 0 \\ 0 & x < 0 \end{cases}$

Durch Gedächtnislosigkeit: Neuronen können beliebig schnell hintereinander feuern

Ignorieren der *refractory period*, einfache Lösung: Gamma-Prozess (ISIs sind gammaverteilt)

Vorteile des Poisson-Prozess: Koinzidenzen können extrem einfach simuliert werden

Mutter-Prozess pro Ensemble aus dem Spikes kopiert werden, Reduzierung des „Rauschens“ liefert Poisson-Prozess mit den selben Eigenschaften

Simulation von Spike Trains

Poisson- und Gammaprozesse per se stationär (d.h. konstante Feuerrate)

Simulation von nicht-stationären Prozessen durch Warp-Funktion

Schwankungen der Feuerrate werden in einem Feuerratenprofil dargestellt

Diese Funktion wird genutzt, um die Zeitachse des stationären Prozesses zu verformen (*warping*)

Entstehender Spike Train weiß über die Zeit *ungefähr* das gewünschte Feuerratenprofil auf.

Inverse Methode kann auch genutzt werden, um nicht-stationäre Prozesse mit Mitteln für stationäre Prozesse zu untersuchen

Unwarping durch Warpen mit dem inversen Feuerratenprofil, Ergebnis ist ein stationärer Prozess

Weitere Simulatoren

CoDi (**C**ollect and **D**istribute)

Basiert auf zellulären Automaten (ähnlich wie „Game of Life“)

Neurogrid

Platine mit 256x256 Siliziumneuronen, die über onBoard-RAM konfiguriert werden können und so ein neuronales Netz simulieren. Wird im *Human Brain Project* verwendet.

SpiNNaker (**S**piking **N**eural **N**etwork **A**rchitecture)

ARM-basierter Parallelprozessor (mehrere Millionen Prozessoren!) zur mit dem Ziel ein Neuronales Netz von bis zu einer Milliarde Neuronen zu simulieren (Human Brain Project).

Brian (Spiking Neural Network Simulator)

Python-basierter Simulator, der auch größere Netzwerke schnell simulieren kann. Implementiert unterschiedliche Arten von Modellen (Integrate-and-Fire, Hodgkin-Huxley) und erlaubt es, die einzelnen Parameter sehr flexibel einzustellen.