

# Kopplung von Neuronen

Katharina Ritter, Friedrich Bach, Felix Tabbert, Walter  
Tewes, Matthias Walther

WWU Münster

12.06.2012

# Inhalt

Einführung

Lighthouse-Modell

Numerische Ergebnisse

Schlussbemerkungen

# Unterschiede zum 1–Neuronenmodell

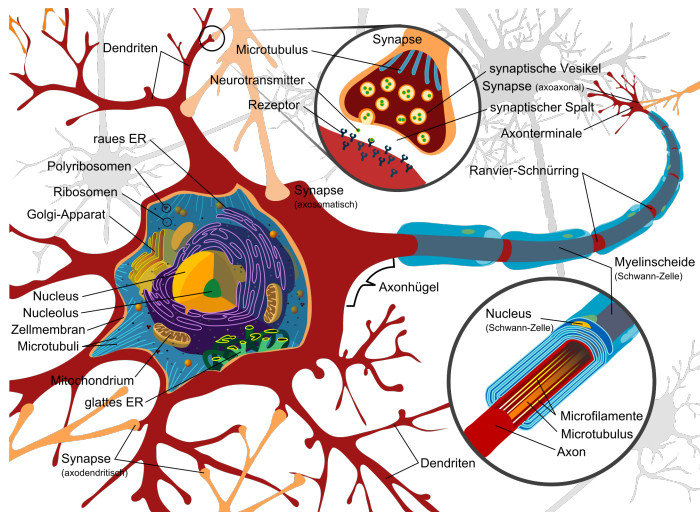
- ▶ betrachten nun ein Netz aus  $N$ –Neuronen
- ▶ diese können auf unterschiedlichste Weisen miteinander verbunden sein (zufällig, benachbart, . . .)
- ▶ Modell möglichst stark vereinfachen
- ▶ Auswirkungen beobachten

# Ziele

Wir interessieren uns für

- ▶ die Wechselwirkung der Neuronen aufeinander
- ▶ insbesondere die Synchronisation der Neuronen ("phase-locked")
- ▶ die Bedingungen, unter denen die Neuronen sich synchronisieren

# Das Neuron – ein komplexes System



[3]

# Modellbildung

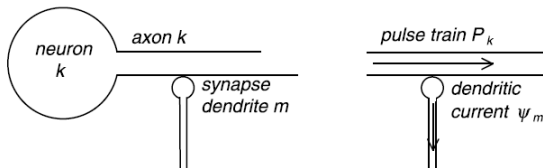
Dynamik im neuronalen Netz lässt sich reduzieren auf:

- ▶ Dendritischer Strom

$$\dot{\Psi}_m(t) = aP_k(t) - \gamma\Psi_m(t)$$

- ▶ Phase

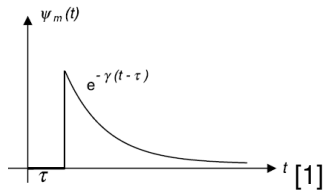
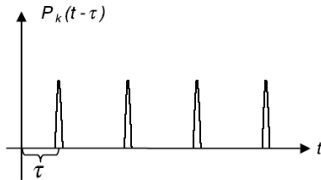
$$\dot{\Phi}_m(t) = \Psi_m(t) + p_{ext,m}(t)$$



[1]

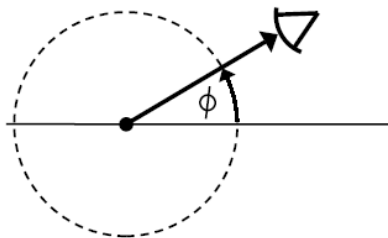
# Dendritischer Strom

$$\dot{\Psi}_m(t) = aP_k(t) - \gamma\Psi_m(t)$$



# Phase

Idee: Neuron feuert  $2\pi$ -periodisch



[1]

Über das Axon laufende Pulse

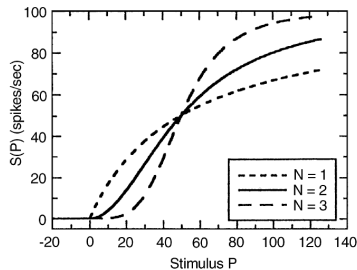
$$P_k(t) = f(\Phi_k(t)) = \sum_{\nu} \delta(\Phi_k(t) - 2\pi\nu) \dot{\Phi}_k(t_{\nu})$$



# Naka-Rushton-Relation

Abhängigkeit: Spike-Rate Neuron  $\leftrightarrow$  Input

$$S(P) = \frac{rP^N}{\Theta^N + P^N}$$

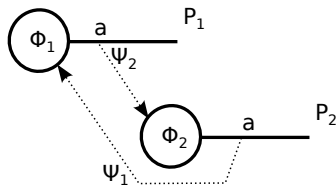


[1]

Für Phase gilt also

$$\dot{\Phi}_m = S(\Psi_m(t) + p_{ext,m}(t))$$

## 2 Neuronen



Neuron 1

$$\dot{\Psi}_1(t) = af(\Phi_2(t)) - \gamma\Psi_1(t)$$

$$\dot{\Phi}_1(t) = c\Psi_1(t) + p_{ext,1}$$

Neuron 2

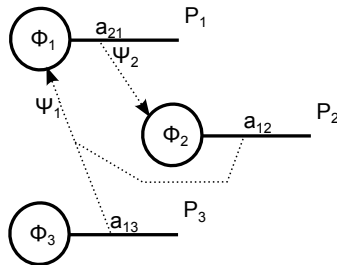
$$\dot{\Psi}_2(t) = af(\Phi_1(t)) - \gamma\Psi_2(t)$$

$$\dot{\Phi}_2(t) = c\Psi_2(t) + p_{ext,2}$$

# N Neuronen

$$\dot{\Psi}_m(t) = \sum_k a_{mk} P_k(t) - \gamma \Psi_m(t)$$

$$\dot{\Phi}_m(t) = S(\Psi_m(t) + p_{ext,m}(t))$$

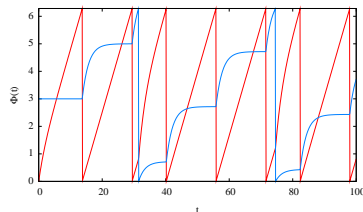


# Modellierung des pulse-trains

Zur Erinnerung (zwei Neuronen):

$$\dot{\Psi}_i(t) = a f(\Phi_j(t)) - \gamma \Psi_i$$

$$\dot{\Phi}_i(t) = \Psi_i(t) + p_{ext,i}$$



- ▶ Ziel: Simulation der DGLn mit RK4.
- ▶ Frage: Wie ist  $f(\Phi(t)) = \sum_{\nu \in \mathbb{Z}} \delta(\Phi(t) - 2\nu\pi) \dot{\Phi}(t) = \sum_{t_\nu} \delta(t_\nu)$  zu modellieren?
- ▶ Lösung: Modelliere  $\delta(t_\nu)$  als normierte Rechteckfunktion mit Breite  $h$  (Zeitschrittweite des RK4 Verfahrens).

## Simulation des *phase-locked state*

Lösung für  $p_{ext,1} = p_{ext,2} =: p$

Vergleich mit Theorie:

- ▶ Erwartet: identischer, konstanter Abstand der Peaks

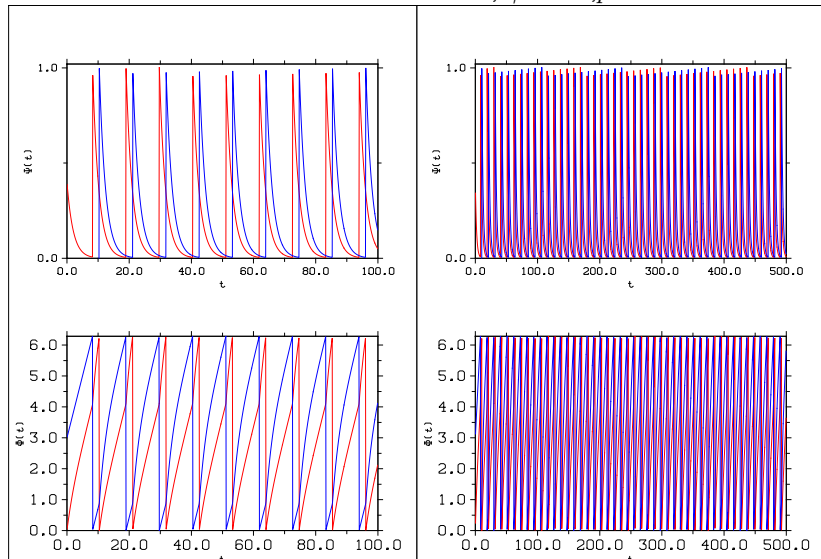
$$\Delta_1 = \Delta_2 = \frac{1}{p} \left( 2\pi - \frac{a}{\gamma} \right)$$

- ▶ Aus der Numerik:

$p$	numerisch	analytisch
0.4	10.708000	10.707963

**Tabelle** : phase-locked state mit  $a = 1.0$  und  $\gamma = 0.5$

Phase locked state für  $t = 100$  und  $t = 500$ ,  $\gamma = 0.5$ ,  $p = 0.4$



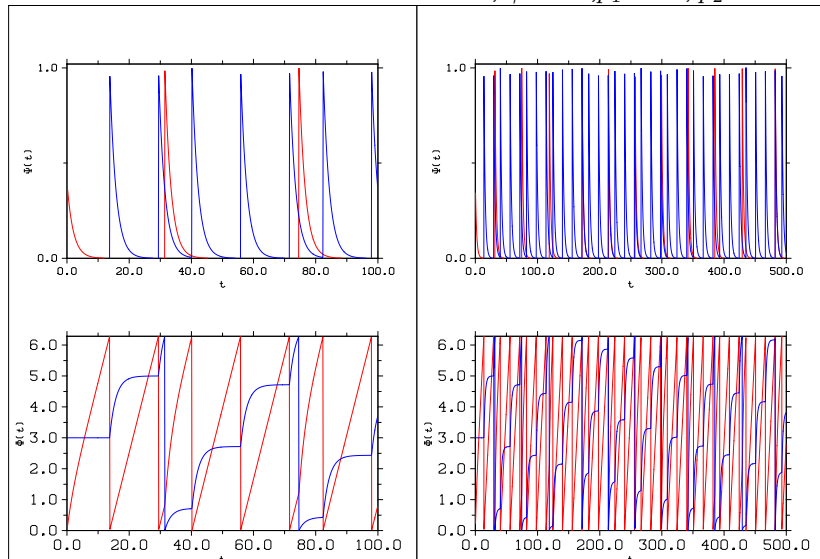
## Simulation des *Frequency Pulling*

Betrachte nun  $p_{ext,1} \ll p_{ext,2}$ :

Vergleich mit Theorie:

- ▶ Erwartet: Anregung des 1. Neurons durch das Zweite
- ▶ Für  $|\gamma\Delta| \ll 1$  erwartete Frequenzen:  $\omega_i = 2\pi \frac{2\pi p_{ext,i} + a p_{ext,j}/\gamma}{4\pi^2 - a^2/\gamma^2}$
- ▶ Endliche Frequenz bei verschwindender externer Anregung eines der Neuronen

Phase locked state für  $t = 100$  und  $t = 500$ ,  $\gamma = 0.5$ ,  $p_1 = 0.4$ ,  $p_2 = 0.0$





# Erweiterung des Modells auf $N$ Neuronen

Zur Erinnerung:

$$\dot{\Psi}_m(t) = \sum_k a_{mk} f(\Phi_k(t)) - \gamma \Psi_m$$

$$\dot{\Phi}_m(t) = \Psi_m(t) + p_{ext,m}$$

$a_{mk}$  Wechselwirkungsmatrix.

- ▶ Ziel: Erweiterung des Codes in  $N$  Dimensionen.
- ▶ Ausgabe: Dynamik des Systems  $\rightarrow$  Animation einer lin. Kette.
- ▶ Auftreten von Strukturen.
- ▶ Normierung bleibt nicht erhalten, weitere Überlegungen notwendig.

# Erweiterung des Modells auf $N$ Neuronen

Zur Erinnerung:

$$\dot{\Psi}_m(t) = \sum_k a_{mk} f(\Phi_k(t)) - \gamma \Psi_m$$

$$\dot{\Phi}_m(t) = \Psi_m(t) + p_{ext,m}$$

$a_{mk}$  Wechselwirkungsmatrix.

- ▶ Ziel: Erweiterung des Codes in  $N$  Dimensionen.
- ▶ Ausgabe: Dynamik des Systems  $\rightarrow$  Animation einer lin. Kette.
- ▶ Auftreten von Strukturen.
- ▶ Normierung bleibt nicht erhalten, weitere Überlegungen notwendig.

# Zusammenfassung

Ziel: Modellierung eines neuronalen Netzes  
Verwendetes Modell: Lighthouse-Modell

Bisher: schwerpunktmäßige Betrachtung von zwei gekoppelten Neuronen

- ▶ Kennenlernen des Modells
- ▶ Klärung von Fragen der numerischen Umsetzung
- ▶ Vergleich mit analytischen Ergebnissen

Jetzt: Übergang zu  $N$ -Neuronen

# Zusammenfassung

Ziel: Modellierung eines neuronalen Netzes  
Verwendetes Modell: Lighthouse-Modell

Bisher: schwerpunktmäßige Betrachtung von zwei gekoppelten Neuronen

- ▶ Kennenlernen des Modells
- ▶ Klärung von Fragen der numerischen Umsetzung
- ▶ Vergleich mit analytischen Ergebnissen

Jetzt: Übergang zu  $N$ -Neuronen

# Ausblick

Bisher: Stromdynamik in statischem Neuronennetz

Nächster Schritt: Simulation neuronaler Plastizität

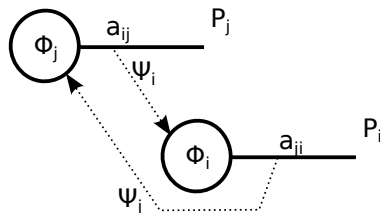
Neuronale Plastizität: Veränderung der Struktur des Neuronennetzes

- ▶ Entwicklungsbedingt
- ▶ Aktivitätsabhängig

In diesem Fall: Aktivitätsabhängige Veränderungen  
Mögliche Ursachen: Lernerfolg

# Synaptische Plastizität nach Hebb

- ▶ Kausales Feuern führt zur Verstärkung der synaptischen Verbindung
- ▶ Akausales Feuern schwächt die synaptische Verbindung



Dies wird mathematisch durch eine zeitabhängige Kopplungsmatrix  $a_{ij}(t)$  realisiert.

# Quellen



Haken, H.: *Brain dynamics: an introduction to models and simulations*. Springer Verlag, 2008.



Chen, C. and Jasnow, D.: *Phys. Rev. E* **81** 011907 (2010)

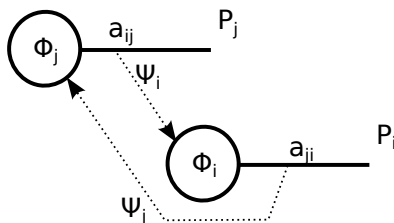


`commons.wikimedia.org`

Vielen Dank für Ihre  
Aufmerksamkeit!



# Mathematische Modellierung der Zeitabhängigkeit



$$\dot{a}_{ij} = \Delta A_i P_i - r a_{ij} B_j P_j$$

$$\dot{A}_i = (1 - A_i) P_j - \frac{A_i}{\tau_A}$$

$$\dot{B}_j = (1 - B_j) P_i - \frac{B_j}{\tau_B}$$