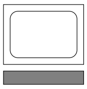
Lecture 7 – Miniprojects

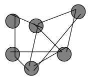
→ -Associative Memory
-Navigation and Reinforcement learning
-Neuron Modeling

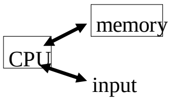
Systems for computing and information processing

Brain

Computer





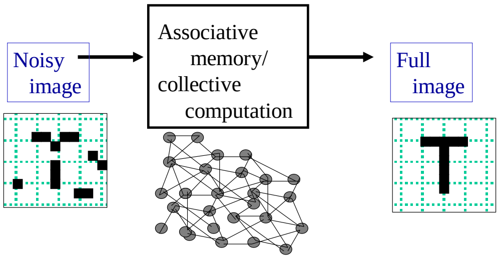


Distributed architecture

(10^{10} proc. Elements/neurons)
No separation of
processing and memory

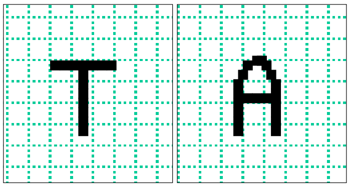
Von Neumann architecture
1 CPU
(10^{10} transistors)

- recognize/understand images:
pattern recognition



Brain-style computation

Associative memory



Prototype $\vec{p}^1$

Prototype $\vec{p}^2$

interactions

$$w_{ij} = \frac{1}{N}\sum_\mu p_i^\mu p_j^\mu$$

Sum over all prototypes

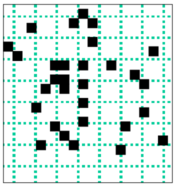
dynamics

$$S_i\,(t+1) = \text{sgn}\left[\sum_j w_{ij} S_j\right]$$

Sum over all interactions with i

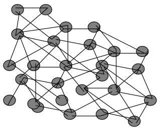
Hopfield model

Associative memory



Prototype $\vec{p}^1$

Interacting neurons

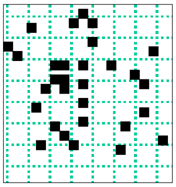


Finds the closest prototype i.e. maximal overlap (similarity) m^μ

Computation
- without CPU,
- without explicit memory unit

Hopfield model

Associative memory



Prototype $\vec{p}^1$

For comparison with Spin S_j=+/-1

$$m^\mu = \tfrac{1}{N}\sum_j p_j^\mu S_j$$

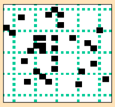
+/-1

Task: Find the prototype with maximal overlap

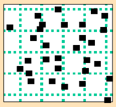
Blackboard:
-Overlap as similarity
-Overlap of random patterns
-Overlap equation

Exercise now: Associative memory

Q: How many prototypes can be stored?



Prototype $\vec{p}^1$



$\vec{p}^2$

Next lecture:10h15

Random patterns

Interactions $w_{ij} = \sum_\mu p_i^\mu p_j^\mu$

Dynamics $S_i\ (t+1)\ =\ \text{sgn}\left[\sum_j w_{ij} S_j\ \ t\ \right]$

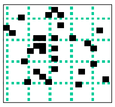
$\Pr\{S_i\ (t+1) = +1\} = 0.5\{1 + g\left[\sum_j w_{ij} S_j\ \ t\ \right]\}$

Derive macroscopic equation

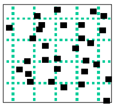
$m^\nu(t+1) = \frac{1}{N}\sum_i \text{sgn}\left[m^\nu(t) + p_i^\nu \sum_\mu p_i^\mu m^\mu\ \ t\ \right]$

$m_i^\nu(t+1) = ?$

Q; How many prototypes can be stored?



Prototype $\vec{p}^1$



Prototype $\vec{p}^2$

Random patterns

Interactions (1) $w_{ij} = \sum_\mu p_i^\mu p_j^\mu$

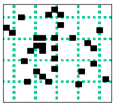
Dynamics (2) $S_i\ (t+1) = \text{sgn}\ \sum_j w_{ij} S_j$

Minimal condition: pattern is fixed point of dynamics

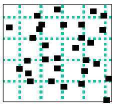
-Assume we start directly in one pattern
-Pattern stays

Attention: Retrieval requires more (pattern completion)

Exercise series 5: How many prototypes can be stored?



Prototype $\vec{p}^1$



Prototype $\vec{p}^2$

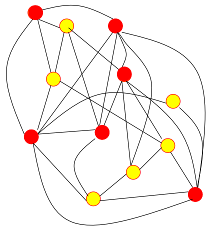
Random patterns
with mean activity a=0
(50 percent active neurons)

Interactions (1) $w_{ij} = \sum_\mu p_i^\mu p_j^\mu$

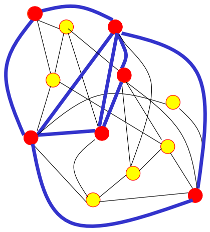
Dynamics (2) $S_i\ (t+1) = \text{sgn}\ \sum_j w_{ij} S_j$

Idea:
-rewrite equations
-map to random walk!

Hebbian Learning and Associative Memory



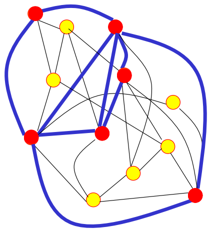
Hebbian Learning



item memorized

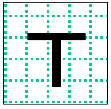
Hebbian Learning

Recall:
Partial info



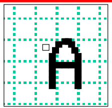
item recalled → *Associative memory*

Exercise series 5: learning of prototypes



Prototype

$\vec{p}^1$



Prototype

$\vec{p}^2$

interactions

$$(1)\quad w_{ij} = \tfrac{1}{N}\sum_{\mu} p_i^{\mu} p_j^{\mu}$$

Sum over all prototypes

a) Show that (1) corresponds to a rate learning rule

$$(2)\quad \frac{d}{dt} w_{ij} = a_2^{corr} (\nu_j^{pre} - \vartheta)(\nu_i^{post} - \vartheta)$$

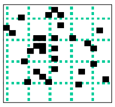
Assume that weights are zero at the beginning;

Each pattern is presented (enforced) during 0.5 sec (One after the other).

note that $p_j^{\mu} = \pm 1$ *but* $\nu_j \geq 0$

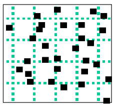
b) Compare with: $\frac{d}{dt} w_{ij} = a_0 + a_1^{pre}\nu_j^{pre} + a_1^{post}\nu_i^{post} + a_2^{corr}\nu_j^{pre}\nu_i^{post} + \ldots$

Miniproject-1: learn always!



Prototype

$\vec{p}^1$



Prototype

$\vec{p}^2$

Random pattern
(50 percent active neurons)

Interactions (1)

$$w_{ij} = \tfrac{1}{N}\sum_{\mu} p_i^{\mu} \quad p_j^{\mu}$$

Dynamics (2) $S_i\,(t+1) = \text{sgn}\left[\sum_j w_{ij}\left(S_j(t) - a\right)\right]$

Q: How many prototypes can be stored?
can we keep on learning?
is it useful to forget?

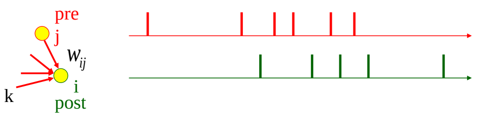
Miniprojects: -teams of 2
-select one of three miniprojects
-research report

Lecture 7 – Miniprojects

✓ -Associative Memory
→ -Navigation and Reinforcement learning
-Neuron Modeling

Unsupervised vs. reinforcement learning

review: Hebbian Learning

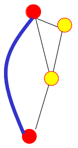


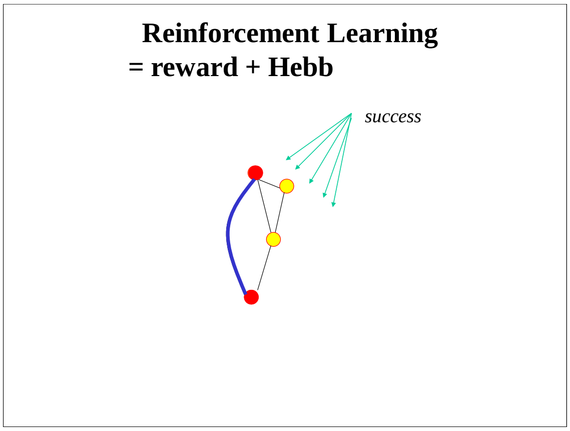
*When an axon of cell **j** repeatedly or persistently takes part in firing cell **i**, then j's efficiency as one of the cells firing i is increased*

Hebb, 1949

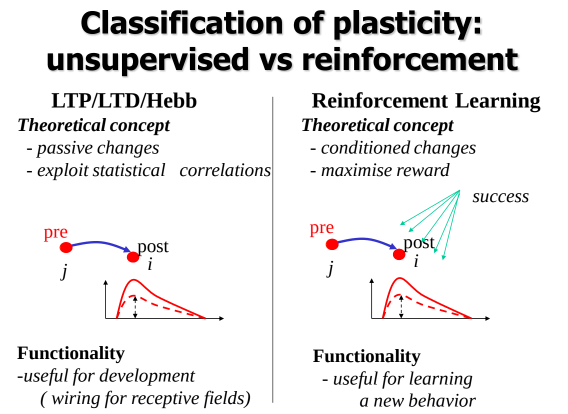
- local rule
- simultaneously active (correlations)

Hebbian Learning
= unsupervised learning





Reinforcement Learning = reward + Hebb



Classification of plasticity: unsupervised vs reinforcement

LTP/LTD/Hebb

Theoretical concept

- passive changes

- exploit statistical correlations

Functionality

-useful for development

(wiring for receptive fields)

Reinforcement Learning

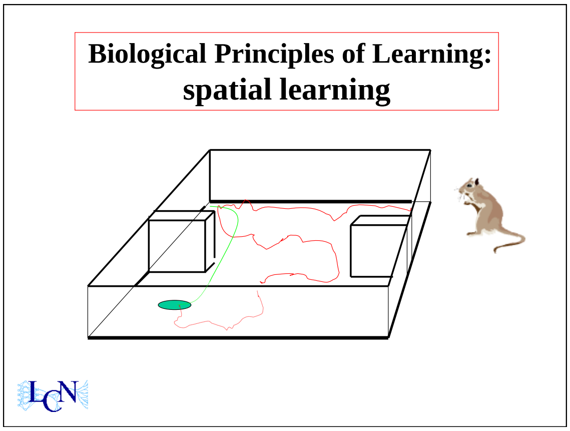
Theoretical concept

- conditioned changes

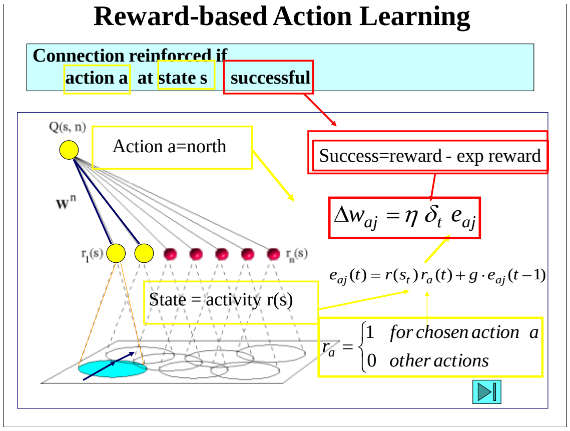
- maximise reward

Functionality

- useful for learning a new behavior



Biological Principles of Learning: spatial learning



Reward-based Action Learning

Connection reinforced if action a at state s successful

Action a=north

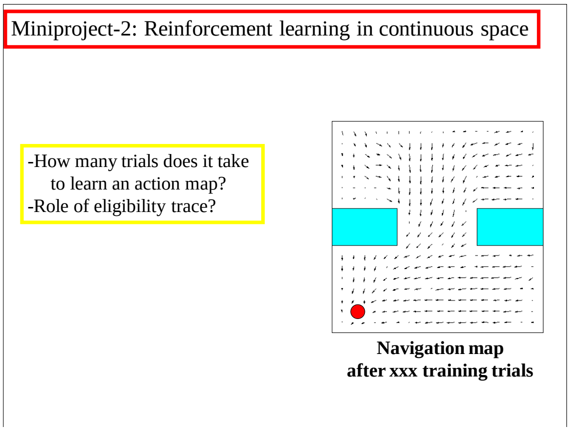
Success=reward - exp reward

$$\Delta w_{aj} = \eta\, \delta_t\, e_{aj}$$

$$e_{aj}(t) = r(s_t)\, r_a(t) + g \cdot e_{aj}(t-1)$$

State = activity r(s)

$$r_a = \begin{cases} 1 & \text{for chosen action } a \\ 0 & \text{other actions} \end{cases}$$



Miniproject-2: Reinforcement learning in continuous space

-How many trials does it take to learn an action map?

-Role of eligibility trace?

Navigation map after xxx training trials

Miniprojects: -teams of 2

-select one of three miniprojects

-research report

Lecture 7 – Miniprojects

√ -Associative Memory

√ -Navigation and Reinforcement learning

→ -Neuron Modeling

Detailed neuron models:
Hodgkin-Huxeley model

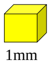
Review of Hodgkin-Huxley models

Dendrite model and cable equation

Synaptic input (conductance input)

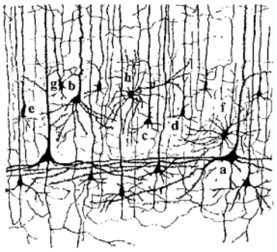
Wulfram Gerstner
http://lcn.epfl.ch

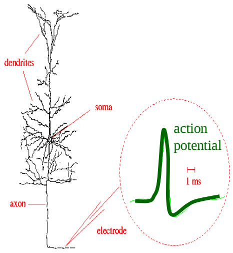
moodle.epfl.ch



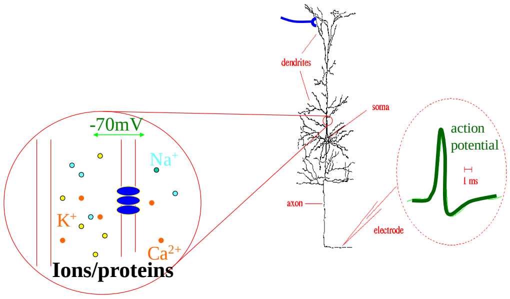
10 000 neurons
3 km wires

Signal:
action potential (spike)





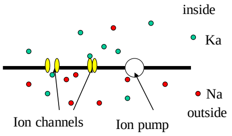
Hodgkin-Huxley type models

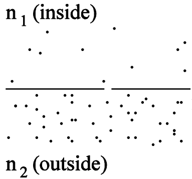


Hodgkin-Huxley Model

Ion density

$$n \propto e^{-\frac{E}{kT}}$$



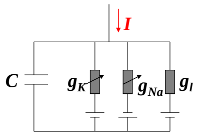


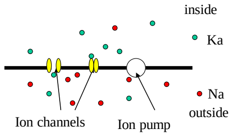
$$\Delta u = u_1 - u_2 = \frac{kT}{q} \ln \frac{n(u_1)}{n(u_2)}$$

Reversal potential

Concentration difference ⇔ voltage difference

Hodgkin-Huxley Model

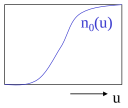


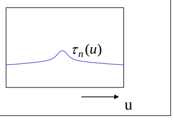


stimulus

$$C\frac{du}{dt} = \underbrace{-g_{Na} m^3 h (u - E_{Na})}_{I_{Na}} \underbrace{- g_K n^4 (u - E_K)}_{I_K} \underbrace{- g_l (u - E_l)}_{I_{leak}} + I(t)$$

$$\frac{dm}{dt} = -\frac{m - m_0(u)}{\tau_m(u)}$$



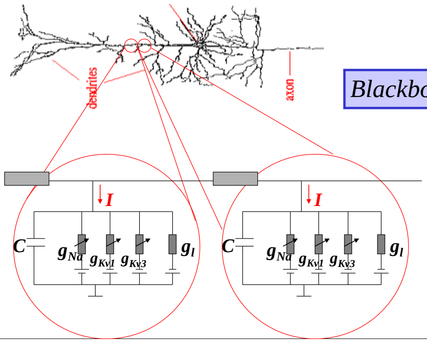


Compartmental models

Hodgkin-Huxley models
with spatial structure

Wulfram Gerstner
http://lcn.epfl.ch/

Spatial structure, Dendrite model, axon model



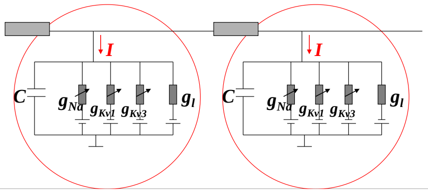
Blackboard

Compartmental model and cable equation

Blackboard

$$C\frac{du_n}{dt}+\sum_{ion} I_n^{ion} = I_n(t)$$

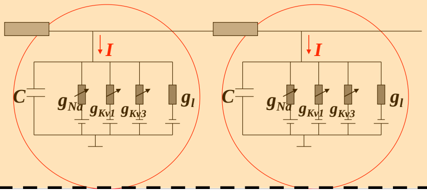
$$C\frac{du_n}{dt}+\sum_{ion} I_n^{ion} = I_n^{stim}(t)+i_n^{axial}-i_{n+1}^{axial}$$



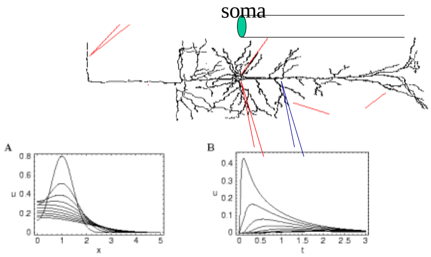
Exercise now:

- derive cable equation
- derive solution of equation

Next lecture:11h45



Cable equation – semi-infinite



Stimulate dendrite, measure at soma

Detailed neuron models:
Hodgkin-Huxeley model

Review of Hodgkin-Huxley models

Dendrite model and cable equation

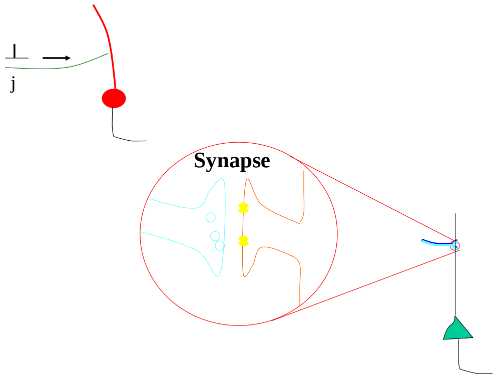
Synaptic input

Other ion currents

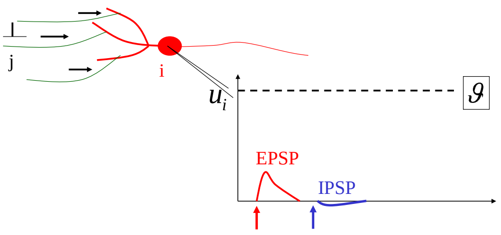
Wulfram Gerstner
http://lcn.epfl.ch

moodle.epfl.ch

Model of Synaptic input: conductance change



Model of Synaptic input: conductance change



$$I^{stim}(t) = -I^{syn}(t) = \underbrace{\sum_{k,f} g_e(t-t_k^f)[u-E_e]}_{\text{EPSC}} + \underbrace{\sum_{k',f'} g_i(t-t_{k'}^{f'})[u-E_i]}_{\text{IPSC}}$$

Detailed neuron models:
Hodgkin-Huxeley model

Review of Hodgkin-Huxley models

Dendrite model and cable equation

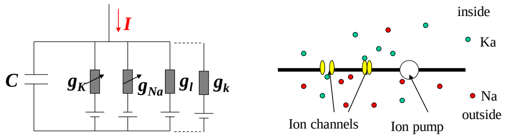
Synaptic input

Other ion currents – NEURON simulator

Wulfram Gerstner
http://lcn.epfl.ch

moodle.epfl.ch

Conductance-based neuron Model



stimulus

$$C\frac{du}{dt} = -\sum_k I_k^{ion} + I(t)$$

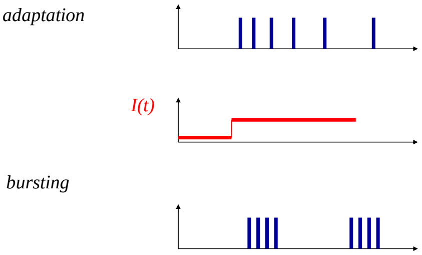
$$I_k^{ion} = g_k\, m^p h^q$$

$$\frac{dm}{dt} = -\frac{m-m_0(u)}{\tau_m(u)}$$

$$\frac{dh}{dt} = -\frac{h-h_0(u)}{\tau_h(u)}$$

Conductance-based neuron Model

Phenomenology:



Miniproject-3: Detailed neuron modeling

Hodgkin-Huxley standard

Other ion currents

Dendritic compartment

Synaptic input

Other ion currents – NEURON simulator

Detailed neuron models:
Hodgkin-Huxeley model

Review of Hodgkin-Huxley models

Dendrite model and cable equation

Synaptic input

→ Other ion currents – NEURON simulator

Wulfram Gerstner
http://lcn.epfl.ch

moodle.epfl.ch

The end