

Seelen bauen

oder

Eine Einführung in neuronale Programmierung



Markus Rudolph



/ Lotes

Inhalt

- Motivation
- Literatur
- Klassische Theorien
- PSI-Theorie
 - Grundlagen
 - Gedächtnisschemata
 - Verhaltensprogramme
 - Komplexe Algorithmen
- Feedback? Angefixt?

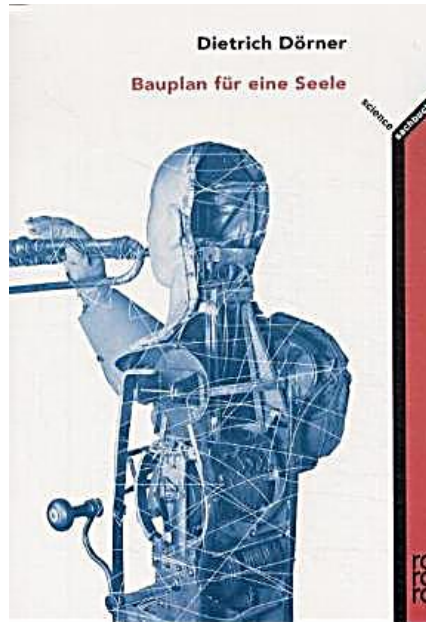


Motivation

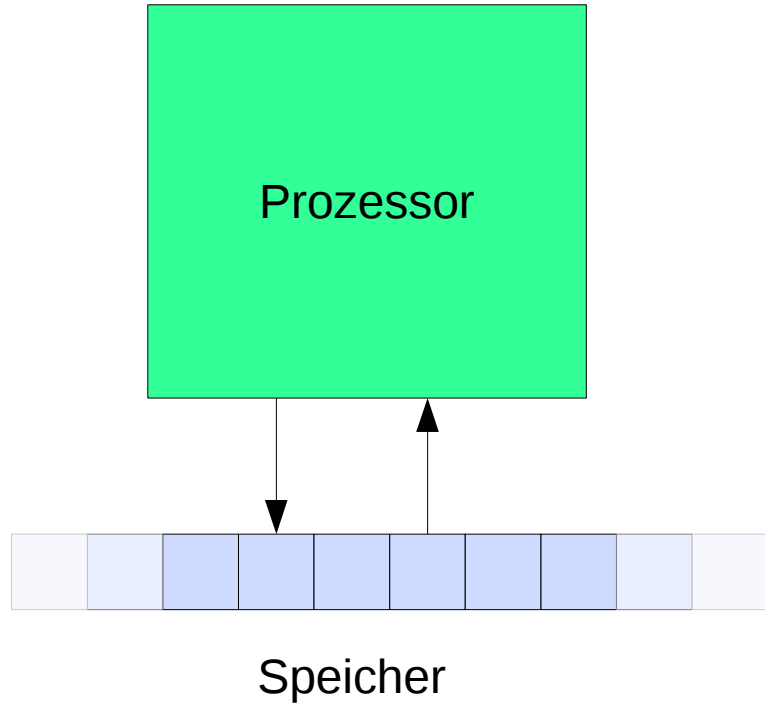
- Eine neuronale Rechentheorie, sowie Vor- und Nachteile dieser, erforschen.
- Hoffnung: Neurale Netze können intuitiv zusammengesetzt werden und Ergebnisse können erklärt werden
- Mitstreiter finden zum Bau neuronaler Programmierwerkzeuge

Literatur

- Diese Theorie wurde erstmals von Dietrich Dörner formalisiert.
- Unter PSI-Theorie betitelt, ist sie in folgenden Büchern erörtert.



Klassische Rechnerarchitektur



- Der typische Rechenzyklus eines Prozessors:
 - Lese Befehl aus Speicher
 - Dekodiere Befehl
 - Führe Befehl aus
 - Schreibe Ergebnis in den Speicher

Neuronale Netze

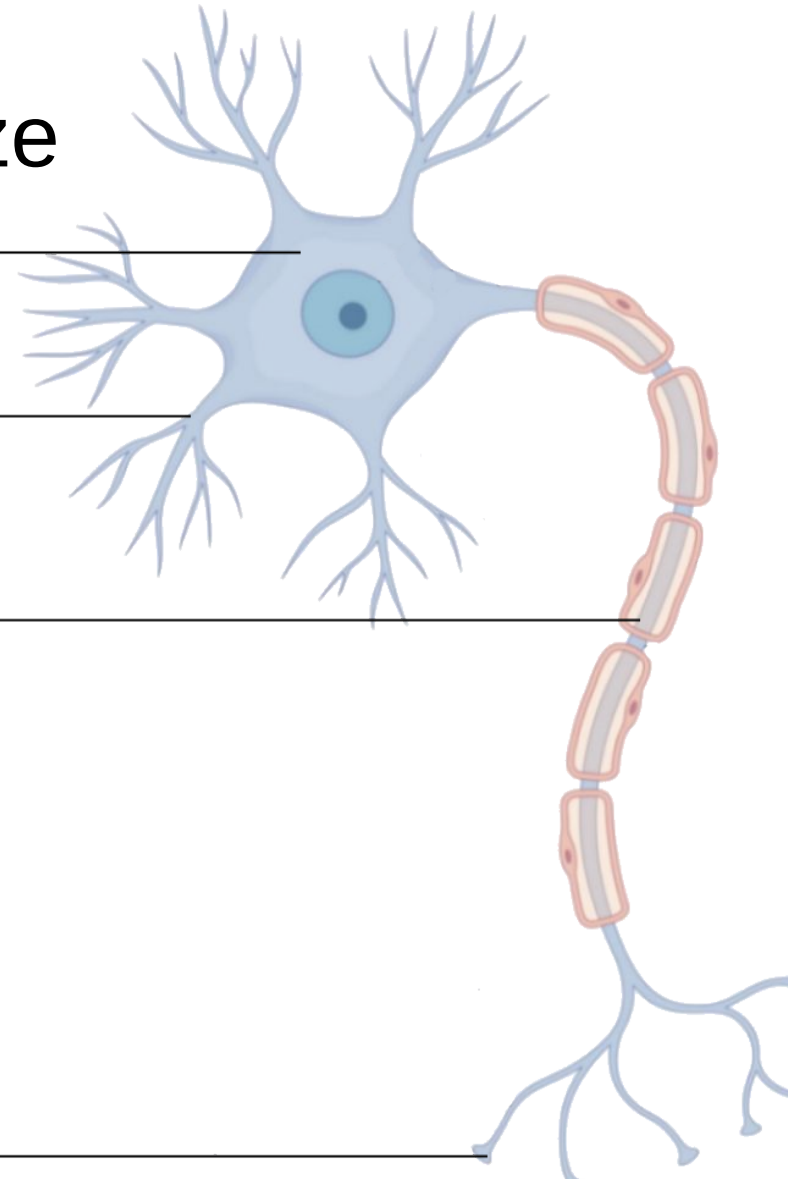
- Neuronen können Signale über ihr Axon feuern
- Dieses verteilt sich über seine Synapsen an andere Neuronen
- Ist die Summe der Eingangssignale größer als ein Schwellwert wird das Signal überhaupt gefeuert
- Synapsen gewichten das Signal und hängen sich an Dendriten der Nachbarneuronen

Neuron

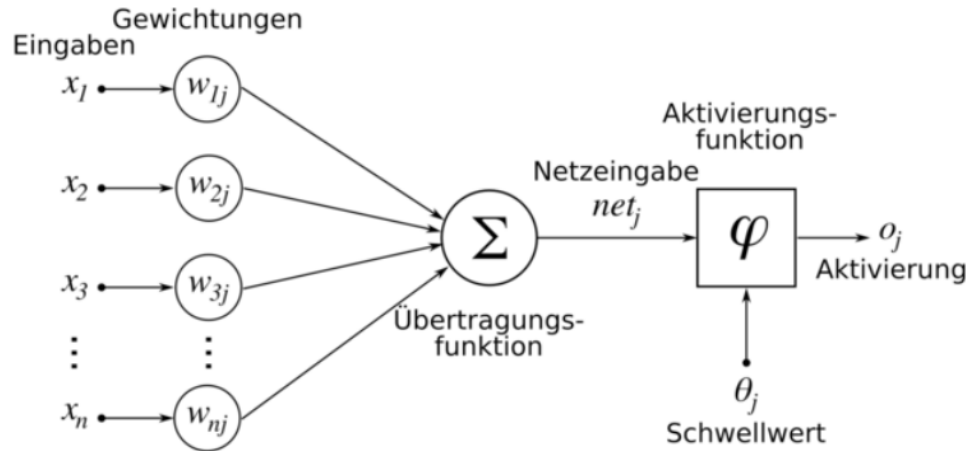
Dendriten

Axon

Synapsen



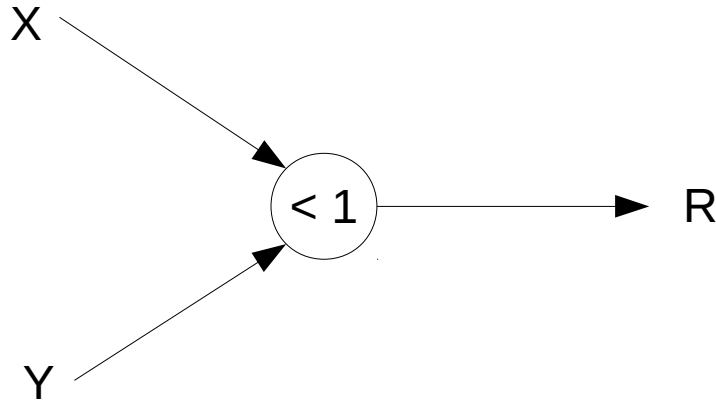
Künstliche neuronale Netze



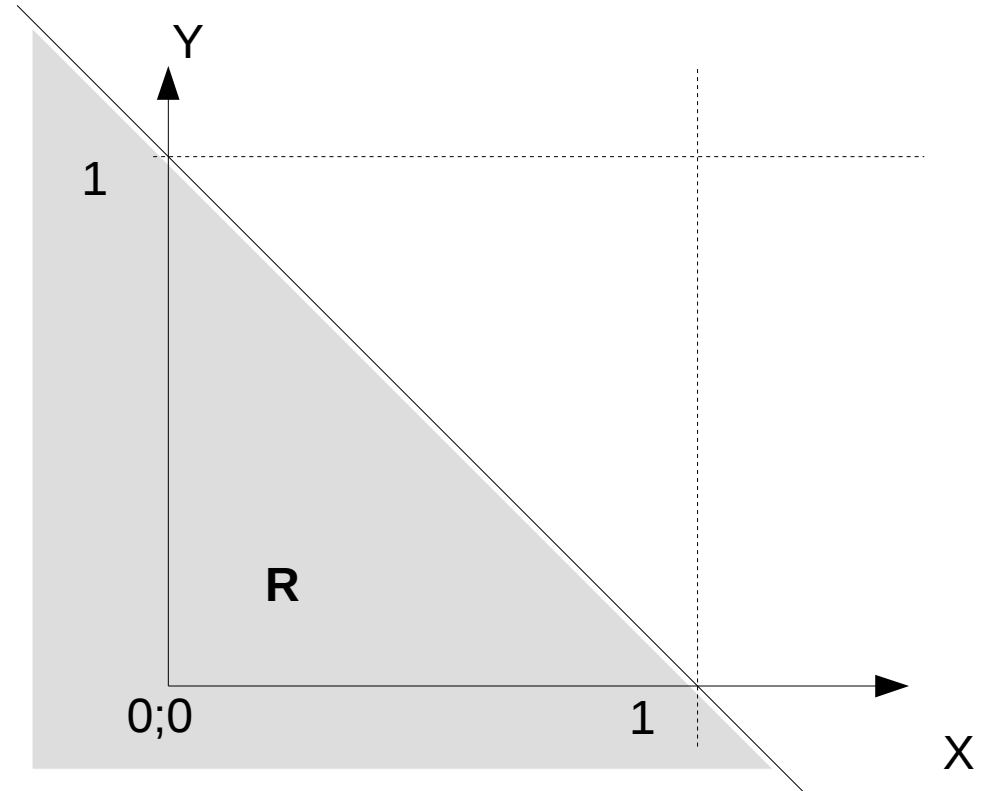
- Die Eingaben x_i für $1 \leq i \leq n$ werden mit dem Gewicht w_{ij} der i -ten Eingangssynapse von Neuron j multipliziert.
- Die Produkte werden addiert.
- Liegt die entstandene Summe über dem Schwellwert vom Neuron, wird das Neuron aktiv und feuert
- Sonst nicht

Künstliche neuronale Netze

Geometrische Interpretation

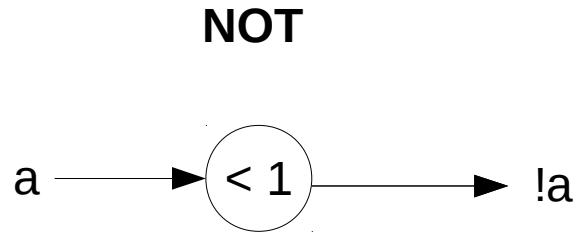
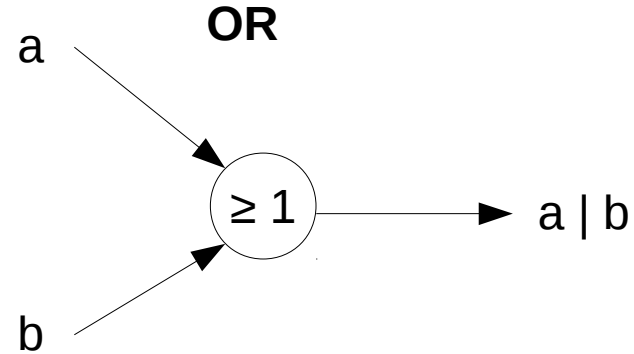
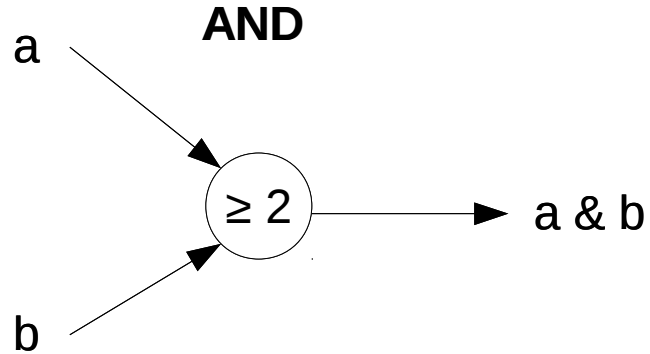


- zwei Eingaben X, Y; eine Ausgabe R
- entspricht der **Spaltung eines 2-dim. Raumes in zwei Halbräume**
- K Eingänge = K Dimensionen!



Logische Schaltungen

mit Neuronen*

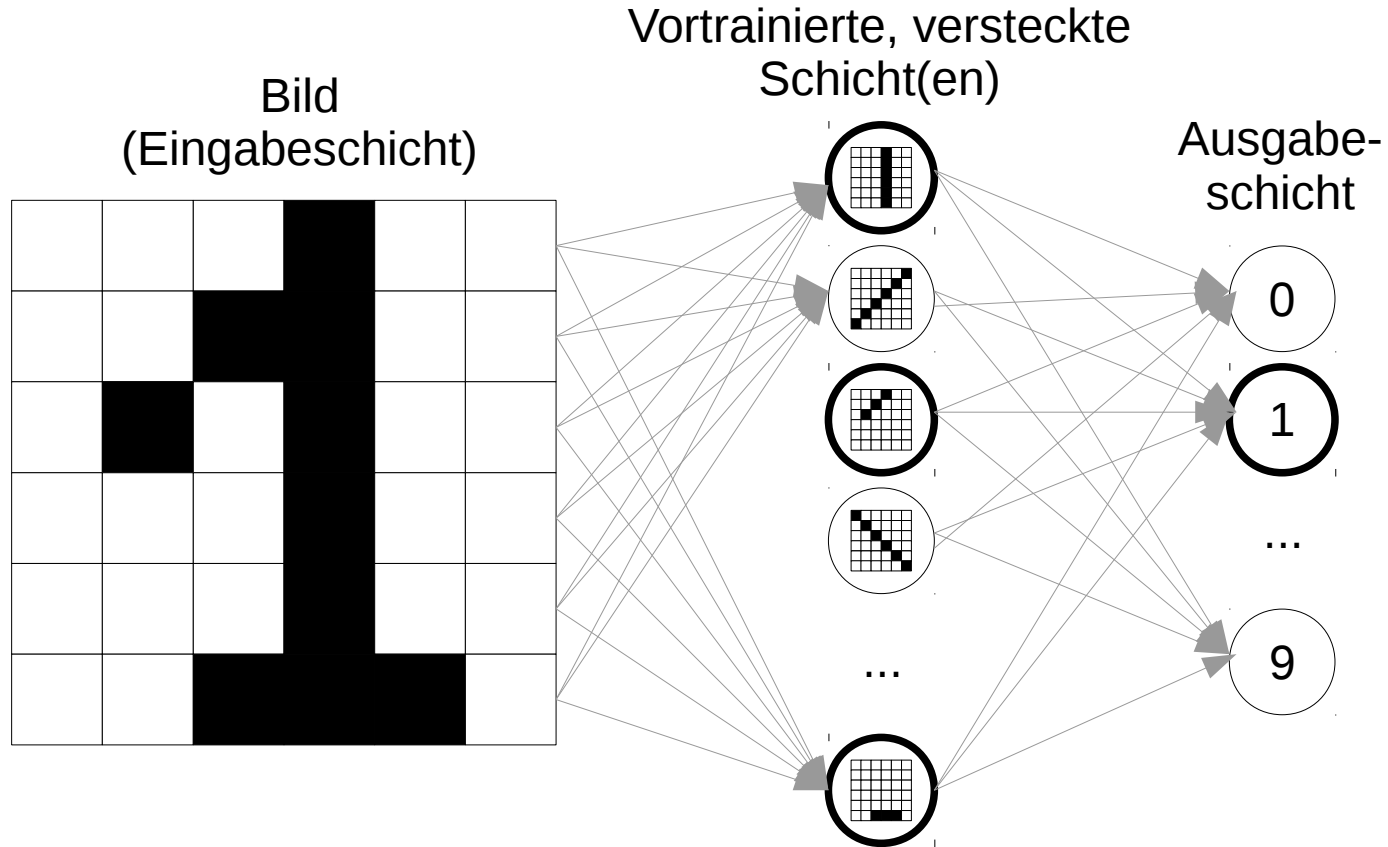


Turingvollständig! Obergeil!

* Natürlich unter der Annahme, dass Ein- und Ausgaben nur 0 oder 1 sind.

Klassische Anwendung

Mustererkennung



Typisches Vorgehen

- künstliche neuronale Netze werden zuerst trainiert und dann benutzt
- Beim Training werden Eingaben durch das Netz propagiert (Netz wird angewendet).
- Wenn die Ausgabe nicht passt, wird der Fehler zurückpropagiert (Synapsengewichte werden dabei geändert).

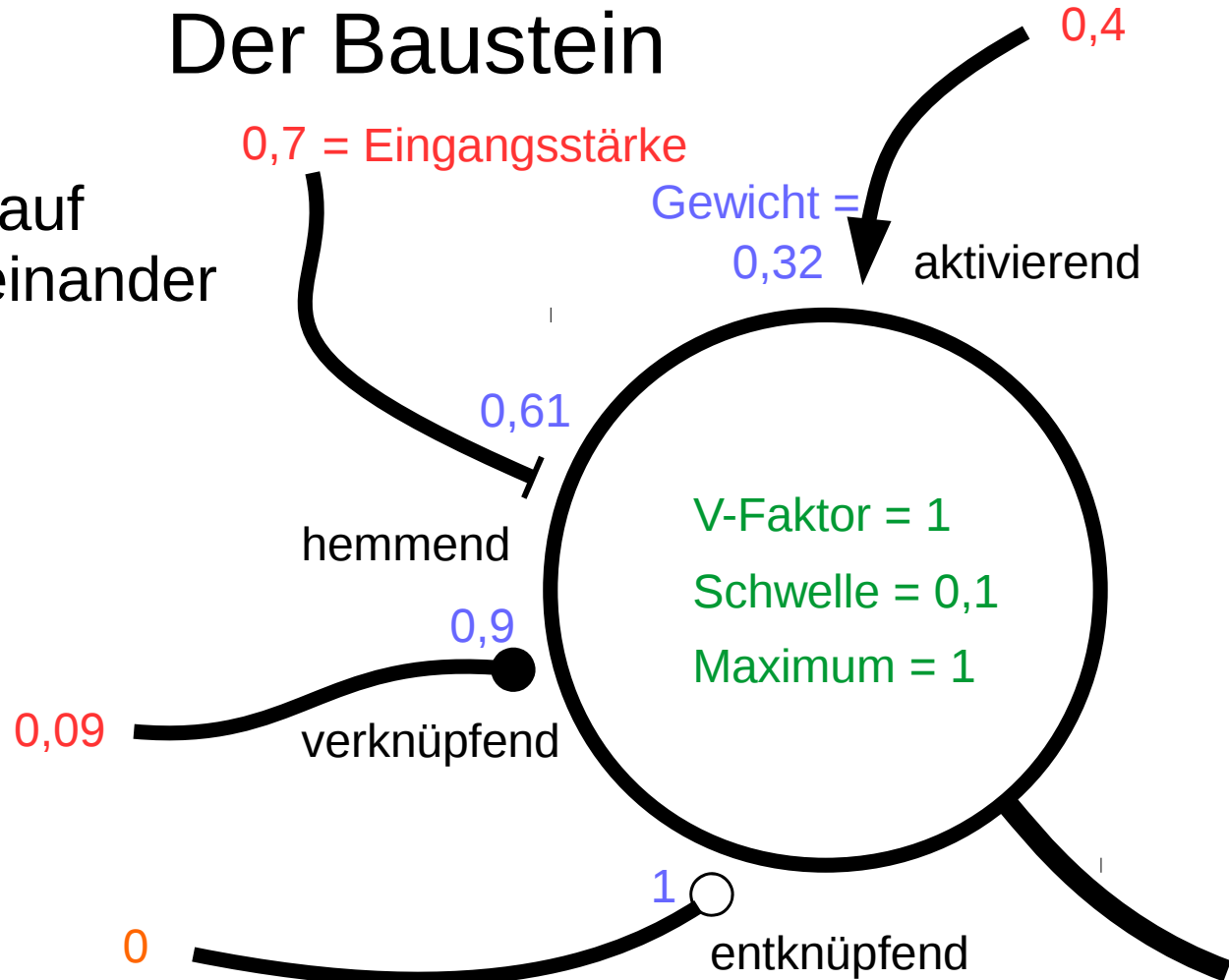
Ψ-Theorie

- modelliert psychische Prozesse ausschließlich mithilfe von Neuronen
- Greift dabei Themen auf wie
 - Motivation, Ziele, Bedarfsindikatoren
 - Handlungsregulation
 - Die Rolle der Sprache
- Diese Präsentation stellt nur die neuronale Programmierung vor (ich finde das alleine schon spannend genug!).

Der Baustein

Neuronen können auf genau 4 Arten miteinander verbunden sein:

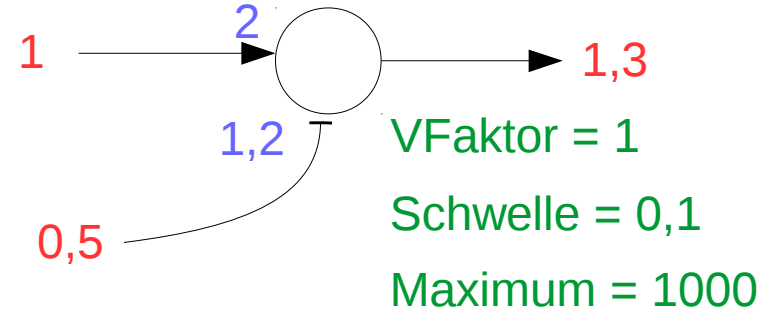
- Aktivierend
- Hemmend
- Verknüpfend
- Endknüpfend



Der Baustein

Aktivierend und hemmend

```
A :=  $\sum(a_i \cdot g_i)$   
if A > Schwelle then  
  begin  
    A := (A-Schwelle) · VFaktor  
    if A > Maximum then A := Maximum  
  end  
else  
  A := 0
```



Eingaben hemmender Neuronen werden von der Summe abgezogen. Das Gewicht wird also negativ interpretiert!

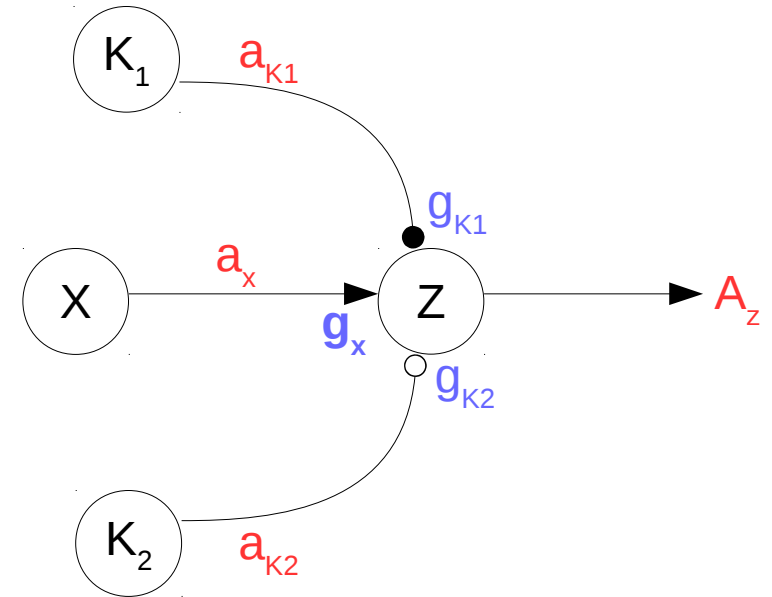
Der Baustein

Verknüpfend und Entknüpfend

Hier werden nur Gewichte anliegender (aktivierender und hemmender) Neuronen geändert.

```
Ak :=  $\sum(a_{ki} \cdot g_{ki})$   
if Ak > 0 then  
    gx :=  $(\sqrt{g_x} + a_x \cdot A_k \cdot A_z \cdot L)^2$   
else if Ak < 0 then  
    gx :=  $\sqrt{(g_x^2 + A_k \cdot A_z \cdot D)}$   
else if gx < T then  
    gx :=  $\sqrt{(g_x^2 - K)}$ 
```

Nach Dörner haben Standardneuronen ein maximales Gewicht von 1.



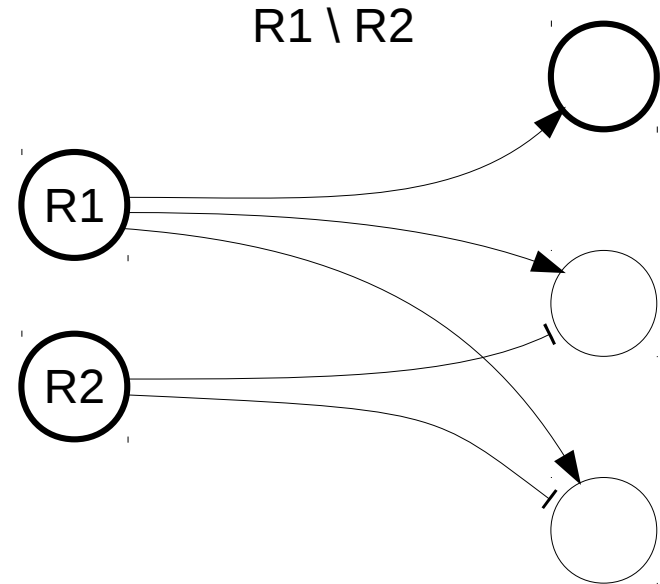
Always remember:
 $A = \sum(a_i \cdot g_i)$

Grundlegende Operationen

Mengenoperationen

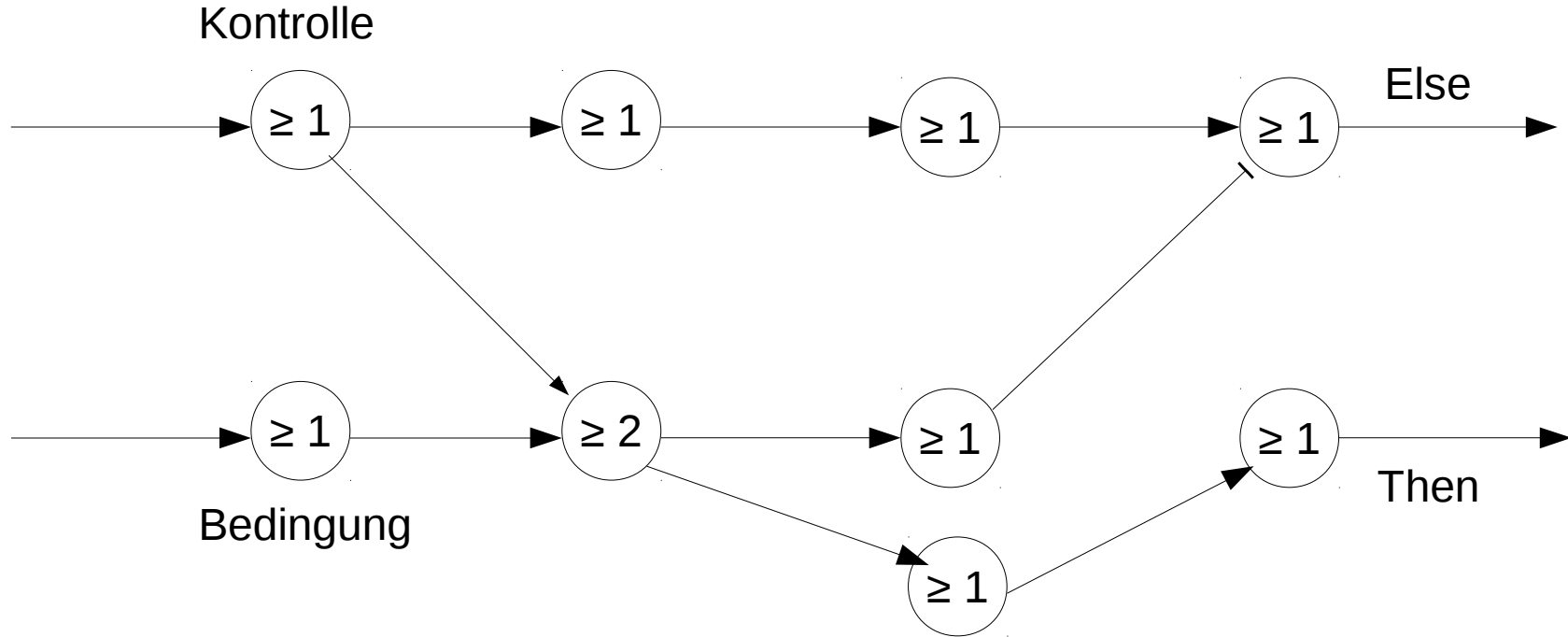
Beispiel: Differenzbildung

- benötigt nur einen Simulationsschritt
- das R steht für Registerneuron



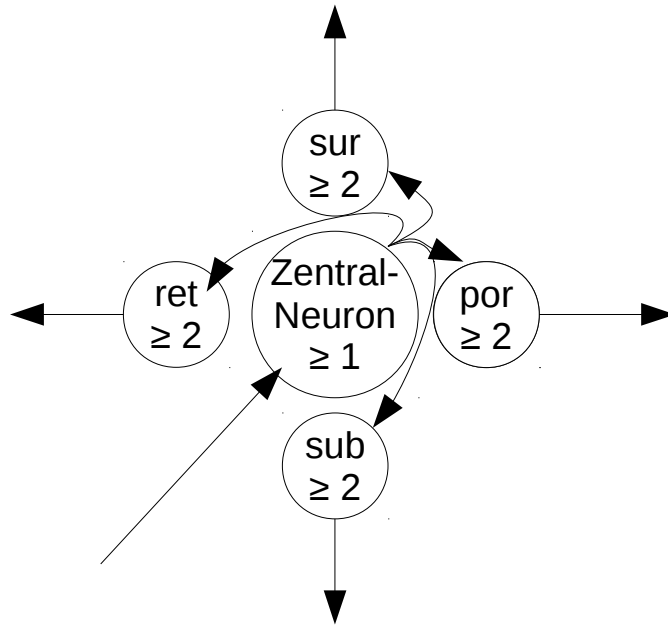
Grundlegende Operationen

Weiche



Level-up

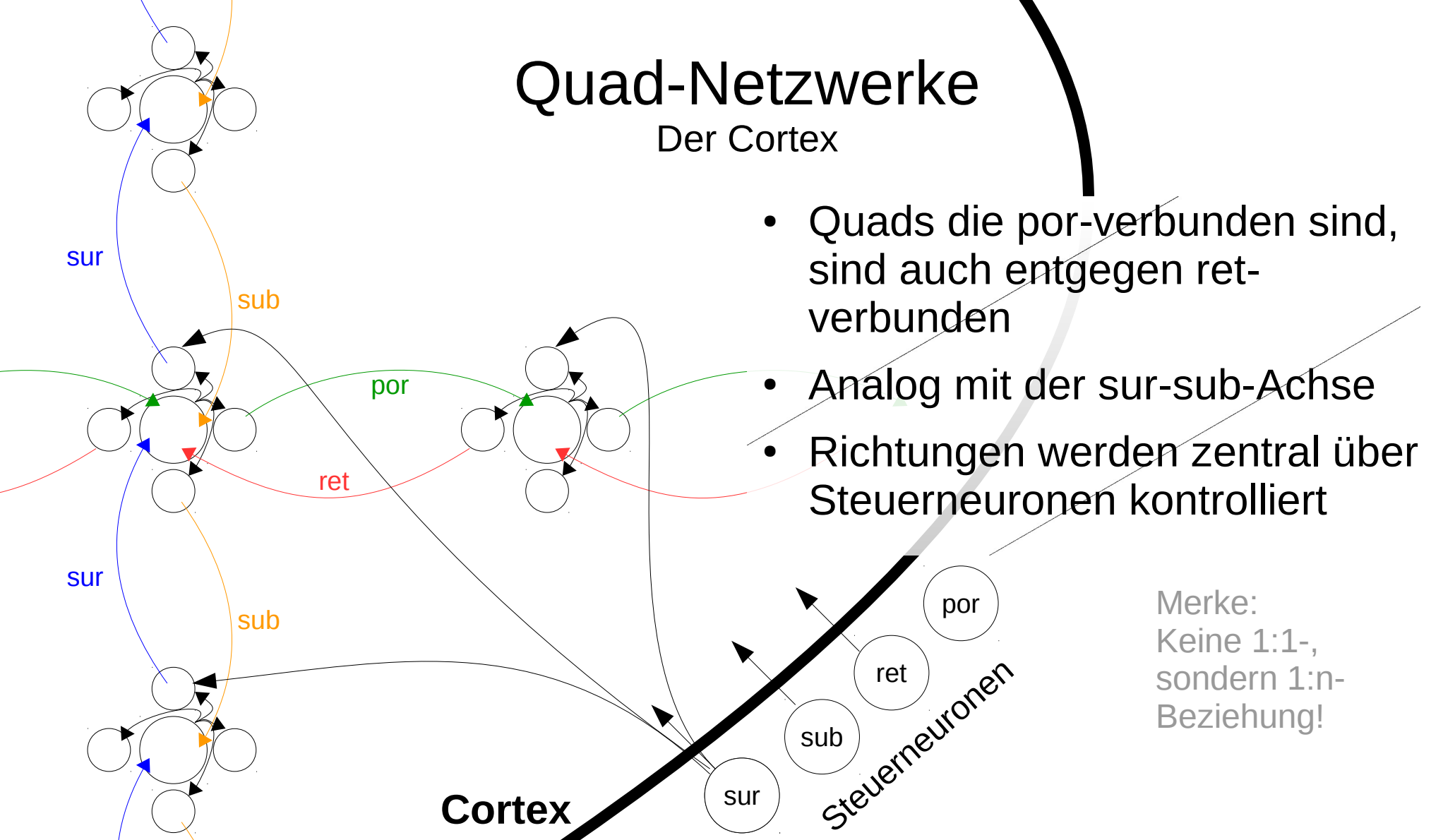
Das Quad



- Quads strukturieren das Gedächtnis
- die vier Richtungen stehen für
 - sur – aufwärts
 - sub – abwärts
 - ret – rückwärts
 - por – vorwärts
- Diese Achsen dienen der Verkettung z.B. in Teil-Ganzes- oder Abstraktion-Spezialisierungs-Relationen, sowie als Aufzählung von ähnlichen Teilen

Quad-Netzwerke

Der Cortex



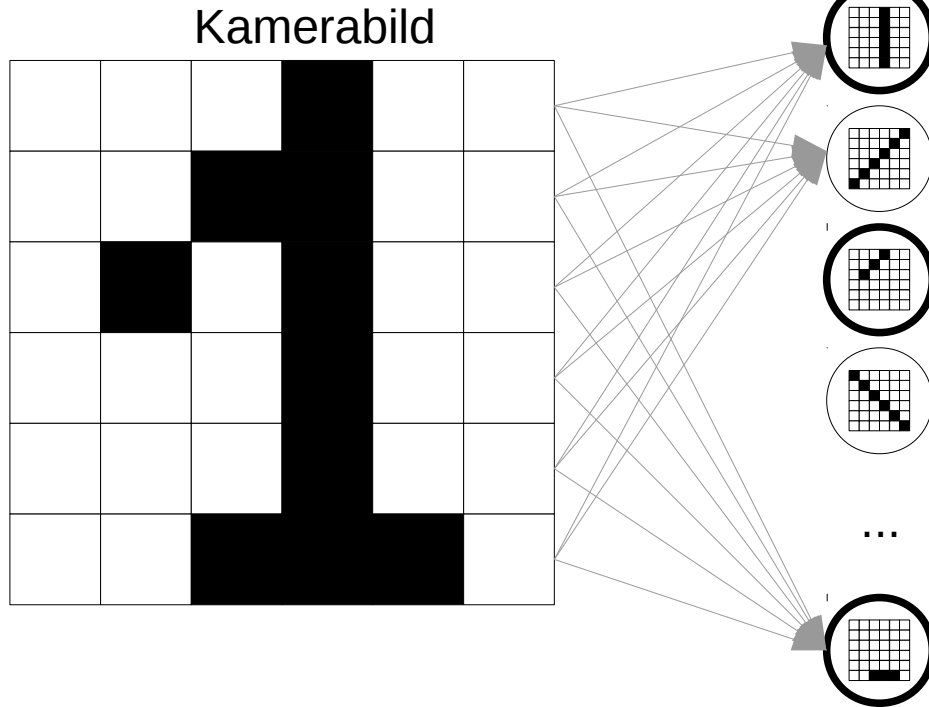
- Quads die por-verbunden sind, sind auch entgegen ret-verbunden
- Analog mit der sur-sub-Achse
- Richtungen werden zentral über Steuerneuronen kontrolliert

Merke:
Keine 1:1-,
sondern 1:n-
Beziehung!

Sensoren

Die Eingabeschnittstelle zur Umwelt

Elementare
Musterdetektoren

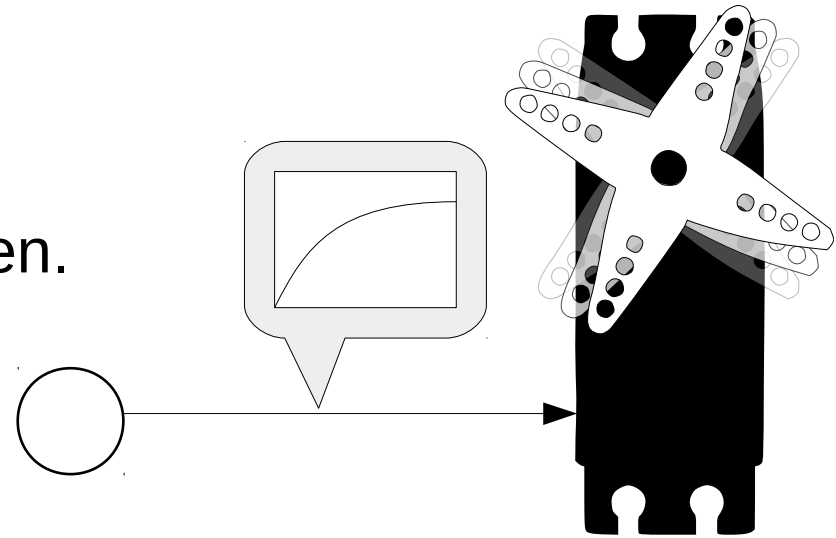


- Elementare Muster könnten, ähnlich wie die 3 Grundfarben, durch Kombination der Einzelmuster sämtliche anderen Muster repräsentieren
- Diese Musterdetektoren dienen als Blätter in sensorischen Schemata

Motoren

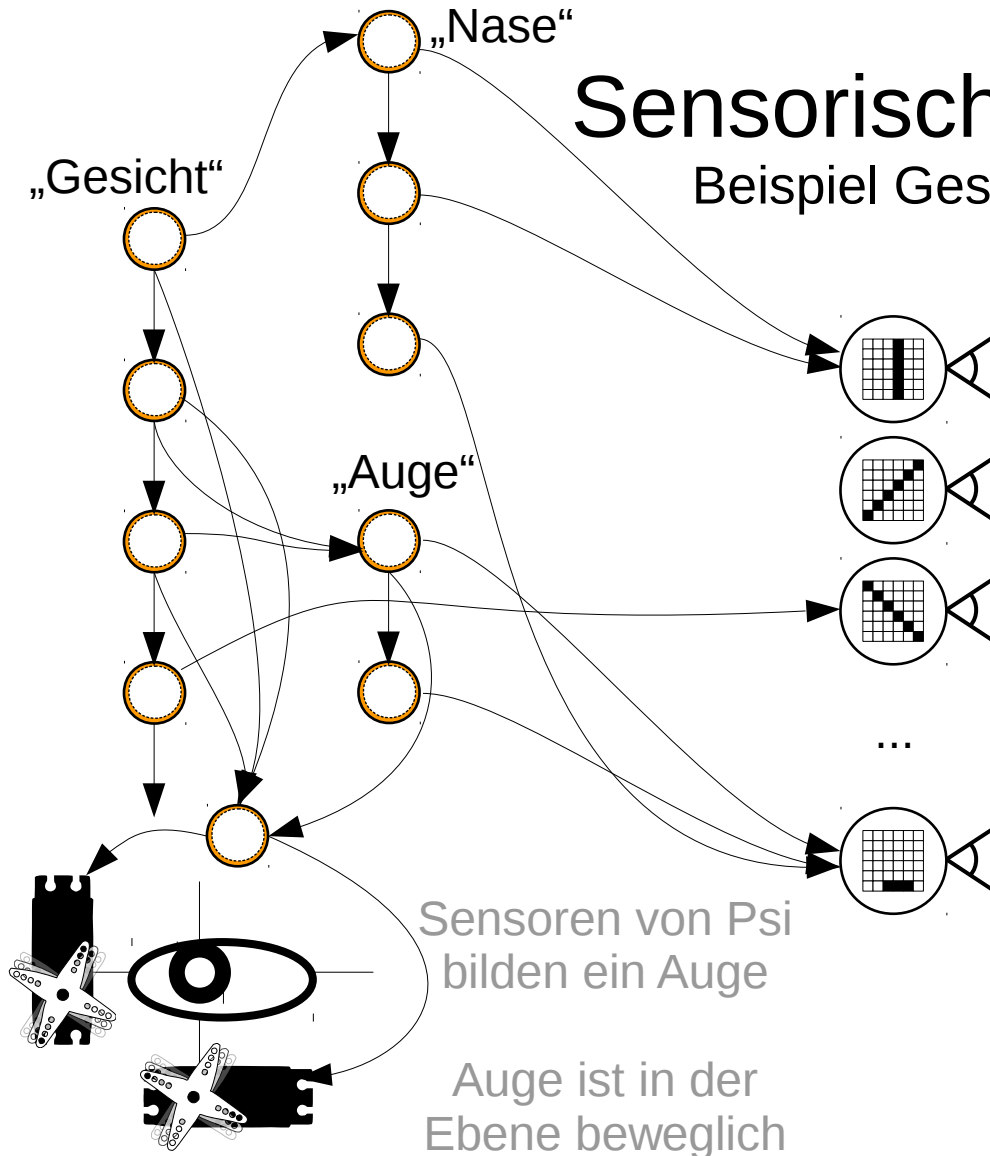
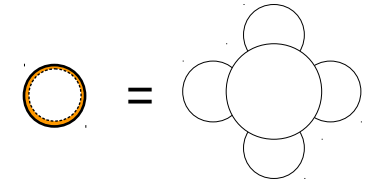
Die Ausgabeschnittstelle zur Umwelt

- Motoren werden von Neuronen gesteuert
- Die Stärke des Signals kann auf die Ausrichtungsstärke eines Servos abgebildet werden.
- Klassisch werden die Muskeln des Auges als Motoren aufgeführt (4 Richtungen: x/y)



Sensorische Schemata

Beispiel Gesichtserkennung



- Das Gesichtsschema teilt sich in weitere Unterschemata und elementare Musterdetektoren auf.
- Motorische Anreize positionieren Psi's Auge.
- Achtung! Bedenkt, dass jeder Pfeil auch ein Gegenstück hat!
- Man kann also durch Psi's Erinnerungen navigieren!

Und nun?

- Wir haben zu diesem Zeitpunkt ein Format um Eingaben zu speichern!
- Wir können Ausgabemodule wie Sensoren ansteuern!
- Es fehlt nur noch das Programm um diese beiden Welten zu verbinden!
- Programme in Psi sind auch Ketten aus Neuronen!
- eine große Herausforderung bei Psi ist es die besprochenen Strukturen anzulegen und abzufragen

Psychische Programme

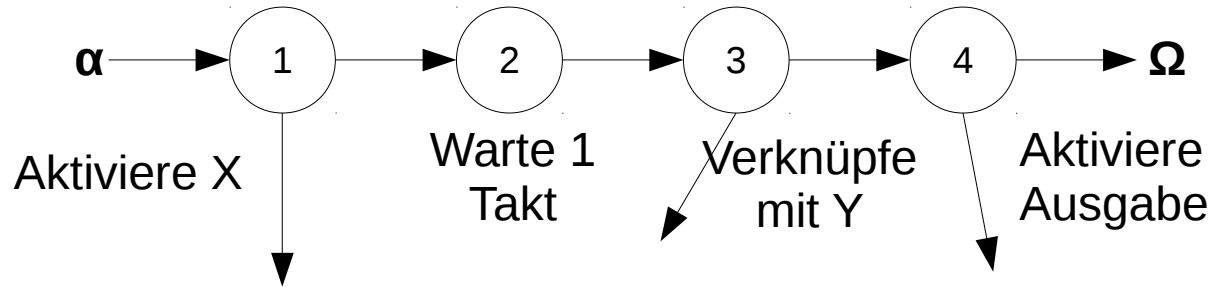
Die Maschinensprache von Psi

- Programme ausschließlich aus Neuronen gefertigt!
- Netz aus Neuronen = Assembler für einen Prozessor
- Psi besitzt auch eine formale Sprache = C als High-Level-Sprache
- Ein weiterer Schritt wäre die Wiederverwendung von Neuronenmodulen, ähnlich der Sprachen Verilog oder VHDL (=“Neuronales C++“).

Psychische Programme

Anweisungssequenzen

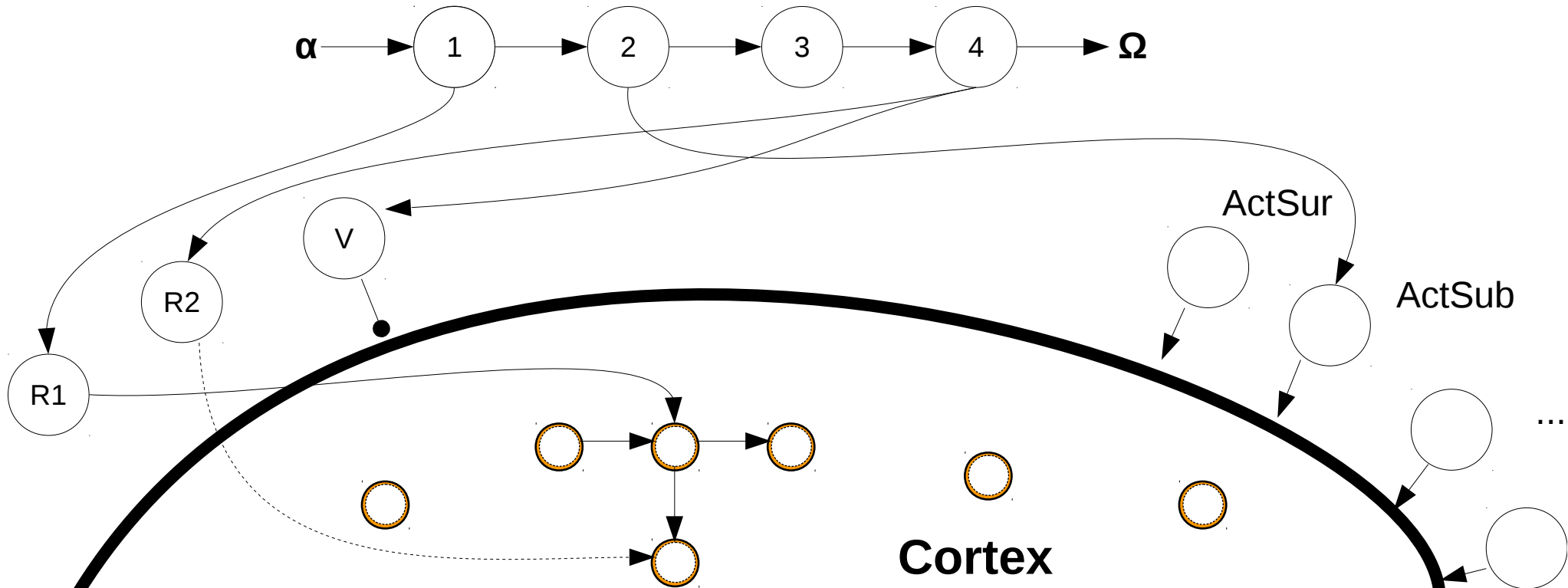
Eine Programmsequenz $a1 ; a2 ; a3 ; a4$ wird als Kette von Neuronen dargestellt



Psychische Programme

Zuweisungen

Es gibt sogar Variablen in Psi in Form von sogenannten Registerneuronen. Folgende Kette steht für „ $R2 := R1 \wedge \wedge_{\text{sub}}$ “.



Psychische Programme

Komplexe Algorithmen

- Im Mittelpunkt steht natürlich die Datenstruktur „Cortex“, die Quads mit jedem anderen Quad beliebig gewichtet verbinden kann.
- Algorithmen werden schnell unübersichtlich in Neuronenschreibweise.
- Algorithmen werden aber dadurch nicht weniger spannend ;-).

Psychische Programme

Wie geht es weiter?

- Dörner hat einige nützliche, neuronale Programme entwickelt (zu groß für die Präsentation)
- Seine Schilderungen sind so spannend, dass man gleich selber losprogrammieren will. => Es gibt aber kein Framework dafür! (nur ein Delphi-Projekt)
- Ein weiterer Nachteil wird sein, dass die Vernetzung so hochgradig ist, dass es schwer wird, Echtzeitsimulationen laufen zu lassen.

Herausforderungen

Was könnte man machen?

- programmieren erst spaßig, wenn man kurze Compiler-Anwortzeiten
 - Compiler auf Assemblerebene + schnellen Simulator/Debugger bauen
 - Compiler auf C und C++-Ebene folgen
 - Grafischer Editor (in Form eines Spieles)
 - Hardware oder Cloud-Dienst zur schnellen Ausführung entwickeln
 - Test-Frameworks, Package-System, Open-Source-Community, ...
 - Fiddles mit Berechnung bei AWS

Angesteckt?

Bei Kritik und Fragen nicht verzagen!

Schreibt mir eine Email an markhoernchen@gmx.de .

Ich hoffe ich konnte zeigen, wie faszinierend neuronale Programmierung ist.

Mich würde es freuen, wenn ihr gerne mitprogrammieren wollt!