



Hochschule **RheinMain**
University of Applied Sciences
Wiesbaden Rüsselsheim

COMPUTERSPIELE MIT EVOLUTI- ONÄREN ALGORITHMEN GEWIN- NEN

Fachseminar “Machine Learning”

Letztes Update: 13. Januar 2016

Harald Heckmann

Studienbereich Informatik
Hochschule RheinMain



GLIEDERUNG

1. Einleitung
2. Neuronale Netze
3. NEAT
4. Super Mario mit NNs und NEAT gewinnen

EINLEITUNG

EINLEITUNG

Das Ziel dieser Präsentation ist es ein grundlegendes Verständnis über neuronale Netze und genetische Algorithmen aufzubauen und mit diesem Wissen anhand eines Beispiels zu verstehen, wie Computerspiele mit optimierten neuronalen Netzen gewonnen werden können.

NEURONALE NETZE

Definition

Neuronale Netze sind eine vereinfachte Nachahmung des Lern- und Denkverhaltens lebender Organismen, die als Computerprogramm realisiert werden können um so ein selbstständig lernendes System zu erzeugen.

- ▶ Neuronale Netze werden verwendet, wenn ein System (selbstständig oder mit einem Lehrer) etwas lernen soll
- ▶ Das Neuronale Netz besteht aus Knoten (Neuronen) und Kanten (Synapsen)
- ▶ Die Kanten sind Gewichtet, die Gewichtung gibt an wie stark und auf welcher Art und Weise (stimulierend, hemmend) die Signale wirken
 - ▶ Im Gewicht der Kanten ist das Wissen und somit das Verhalten des Netzes gespeichert

EVOLUTIONÄRE ALGORITHMEN

Definition

Evolutionären Algorithmen liegen die darwinischen Prinzipien als Grundsatz vor. Diese Algorithmen optimieren neuronale Netze mittels stochastischen und metaheuristischen Verfahren, welche durch Beobachtungen der natürlichen Evolution definiert werden.

- ▶ Diese Algorithmen gehören von der Arbeitsweise her üblicherweise zu der Kategorie „trial and error“ (Versuch und Irrtum, Ausprobieren).

GENETISCHE ALGORITHMEN

Definition

Genetische Algorithmen sind spezielle Evolutionäre Algorithmen, welche Lösungen zu Optimierungsproblemen liefern. Die Arbeitsweise der Algorithmen ist geprägt durch die natürliche Evolution. Die neuronalen Netze können durch Vererbung, Selektion, Kreuzung und Mutation geformt werden.

NEAT

Definition

NEAT ist eine Methode die dazu dient die Struktur eines neuronalen Netzes zu optimieren und neue Lösungen zu finden, wobei der Suchraum der Lösungen möglichst klein gehalten werden soll. Dieser Methode liegt ein genetischer Algorithmus zu Grunde.

NEAT setzt auch Gewichte an Kanten, optimiert diese jedoch nicht. Die Ideen von NEAT lauten:

- ▶ Eine Logik zum Verbinden der Knoten (Gene) um neue Lösungswege zu erzeugen
- ▶ Ein Mechanismus zum Schutz der strukturellen Innovation
- ▶ Der Suchraum der Lösungen soll minimal gehalten werden

REINFORCEMENT LEARNING

Definition

Reinforcement learning wird durch ein Lernproblem charakterisiert. Immer wenn das System auf Rückmeldung der Umgebung angewiesen ist, ist Reinforcement Learning angebracht. Dabei werden neuronale Netze (Lösungen) mittels einer Fitness bewertet, die angibt wie gut ein System gelernt hat.

- ▶ Die Fitness spiegelt den Erfolg hinsichtlich einer bestimmten Aufgabe eines neuronalen Netzes wieder.

Beispiel: Bestimmung der Fitness

In einem Rennspiel ist die Fitness höher desto schneller man ins Ziel kommt

SUPER MARIO

Ziel des Spiels

Ziel des Spiels Super Mario ist es, die Spielfigur Mario so schnell wie möglich in das Ziel zu führen und auf dem Weg möglichst viele Münzen zu sammeln und Gegner zu zerstören.

- ▶ Mario ist eine Spielfigur die zu Fuß unterwegs ist und sich ducken, springen und rennen kann
- ▶ Mario kann Verstärkungen aufsammeln, die sein Durchhaltevermögen erhöhen als auch die Fähigkeit bieten, Projektile abzufeuern, die (hauptsächlich) dazu benutzt werden sich vor Gegnern zu schützen.
- ▶ Mario hat zu Beginn 3 Leben und die Berührung eines Gegners sowie das Fallen in eine Grube können einen Verlust eines Lebens bedeuten
- ▶ Wurden alle Leben verloren, so ist das Spiel vorbei

NEURONALE NETZE

TRAININGSPHASE

- ▶ In der Trainingsphase lernt das Neuronale Netz mit Trainingsmaterial, wobei die Gewichte der Synapsen angepasst werden

Das Training kann folgendermaßen erfolgen:

- ▶ supervised - das gewünschte Ergebnis ist bekannt
- ▶ unsupervised - das gewünschte Ergebnis ist nicht bekannt (hier kann NEAT verwendet werden)

TESTPHASE

- ▶ Nach der Trainingsphase folgt die Testphase
- ▶ Den Inputneuronen werden Reize übermittelt und die Reaktion des Systems darauf wird geprüft

Folgende Reize gilt es zu unterscheiden:

- ▶ Ausgangsreize - Hierbei ist von Interesse, ob das neuronale Netz das Trainingsmaterial richtig erfasst hat
- ▶ neue Reize - Hierbei ist von Interesse, ob das Neuronale Netz darüber hinaus in der Lage gewesen ist eine Generalisierung des Trainingmaterials zu vollziehen

LERNREGELN

- ▶ Um die Gewichte der Synapsen anzupassen braucht man Regeln
- ▶ Es gibt verschiedene Lernregeln für neuronale Netze, die bestimmen welches Aktivitätslevel Synapsen haben
- ▶ Die Hebb-Regel ist eine (einfache) Lernregel und wird im weiteren Verlauf als einfaches Beispiel verwendet

Beispiel: Hebb-Regel

$$\Delta w_{ij} = a_i \cdot a_j \cdot \epsilon$$

Δw_{ij} = Gewichtsveränderung der Synapse zwischen Neuron i und j

a_i = Aktivitätslevel des empfangenden Neurons

a_j = Aktivitätslevel des sendenden Neurons

ϵ = positiver Lernparameter

NEAT

STRUKTUR

- ▶ Ein durch NEAT entstandenes und optimiertes neuronales Netz wird Genotype (Genome) genannt und kann als Phenotype (Netzwerke) dargestellt werden
- ▶ Ein Genom ist ein neuronales Netz und beinhaltet alle Informationen die erlernt wurden
- ▶ Die Phenotype sind lediglich (eine baumartige) Darstellung dieser Genome

GENOTYP (GENOM)

Ein Genom ist eine Struktur die sich aus zwei weiteren Strukturen zusammensetzt:

- ▶ Knotengene (node genes) - Neuronen
- ▶ Kantengene (connection genes) - Synapsen

Ersteres enthält Informationen bezüglich des Types des Genes (z.B. Sensor, Output). Zweiteres enthält Informationen über die Verbindung dieser Gene.

GENOTYP (GENOM)

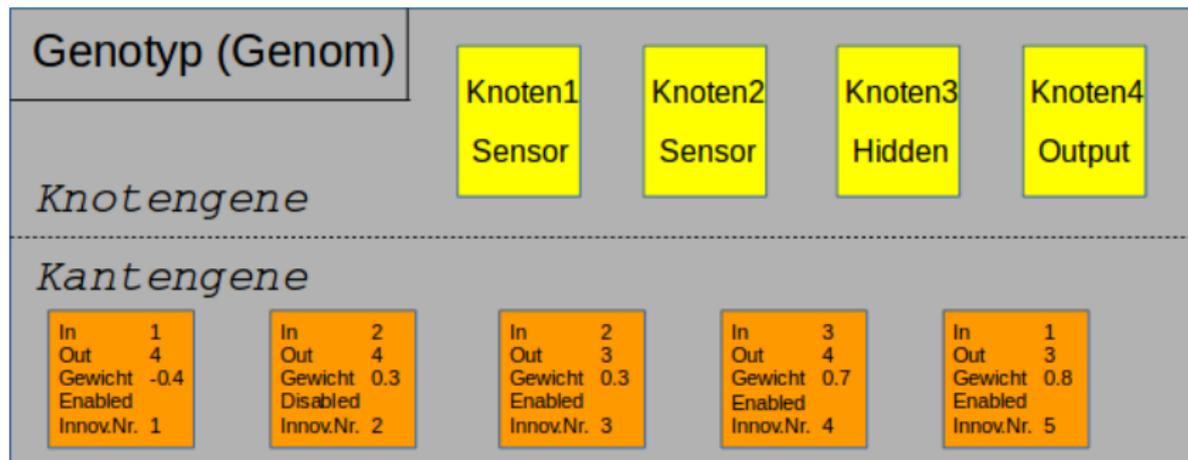


Abbildung: Genotyp (Genom)

PHENOTYP (NETZWERK)

Das Genom kann mit gerichteten Graphen dargestellt werden:

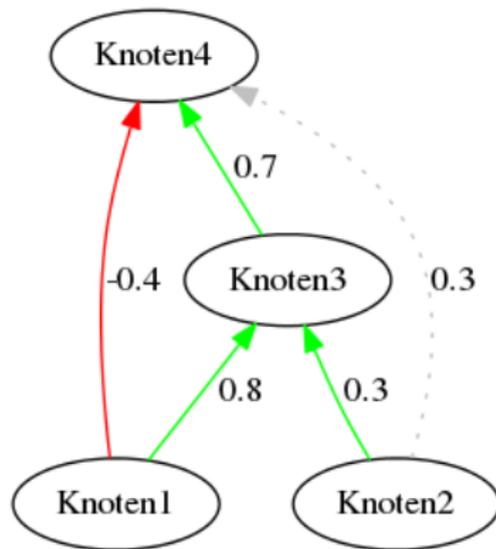


Abbildung: einziges Netzwerk zum Genom aus Abb. 1

GENETISCHE ENKODIERUNG

Eine Kante enthält folgende Informationen:

- ▶ In: Knoten, an dem die Kante anfängt
- ▶ Out: Knoten, an dem die Kante endet
- ▶ Gewicht: Gewichtung der Kante, welche den Einfluss der Verbindung ausdrückt
- ▶ Enabled/Disabled: Aktiv / Inaktiv
- ▶ Innov.Nr.: Innovationsnummer um die Historie zu verfolgen, dazu später mehr

HISTORIE DER GENE

- ▶ Es gibt einen globalen Innovationszähler
- ▶ Jedes Mal, wenn ein neues Gen erscheint, wird die Innovationsnummer des Gens auf den aktuellen Wert des Zählers gesetzt und dieser wird inkrementiert
- ▶ Dies passiert bei Mutationen
- ▶ Durch die Innovationsnummer ist klar, welche Historie ein Gen hat
- ▶ Mittels der Innovationsnummern der Gene findet eine Speziesordnung der entsprechenden Genome statt
- ▶ Mitglieder einer Spezies sind sich Ähnlich und für Kreuzungen untereinander geeignet

MUTATIONEN

1. „Add connection mutation“

- ▶ Zwei bisher unverbundene Knotengene werden verbunden
- ▶ Eine neue Kante entsteht
- ▶ Das Gewicht der Kante wird zufällig gewählt

2. „Add node mutation“

- ▶ Ein Knoten (Gen) wird zwischen einer bestehenden Verbindung eingefügt
- ▶ Die betroffene Verbindung wird deaktiviert und es entstehen zwei neue Verbindungen
- ▶ Die Kante die zum neuen Knoten führt erhält ein Gewicht von 1.0
- ▶ Die Kante die vom neuen Knoten wegführt erhält das Gewicht der Kante die zuvor an dieser Stelle war und aufgeteilt wurde

BEISPIEL: ADD NODE MUTATION

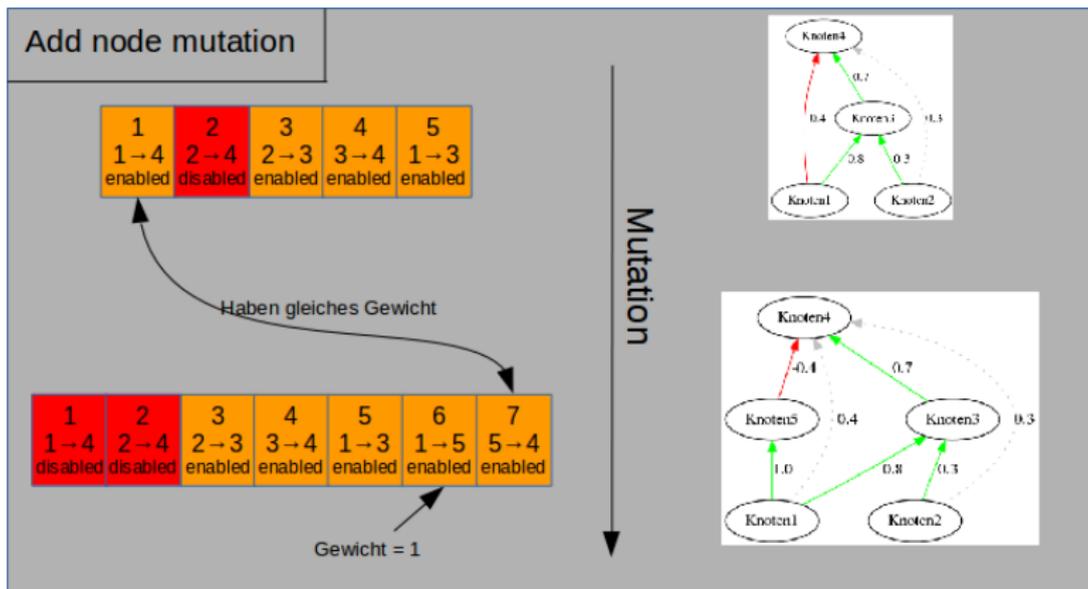


Abbildung: Knotenmutationen durch NEAT

BEISPIEL: ADD CONNECTION MUTATION

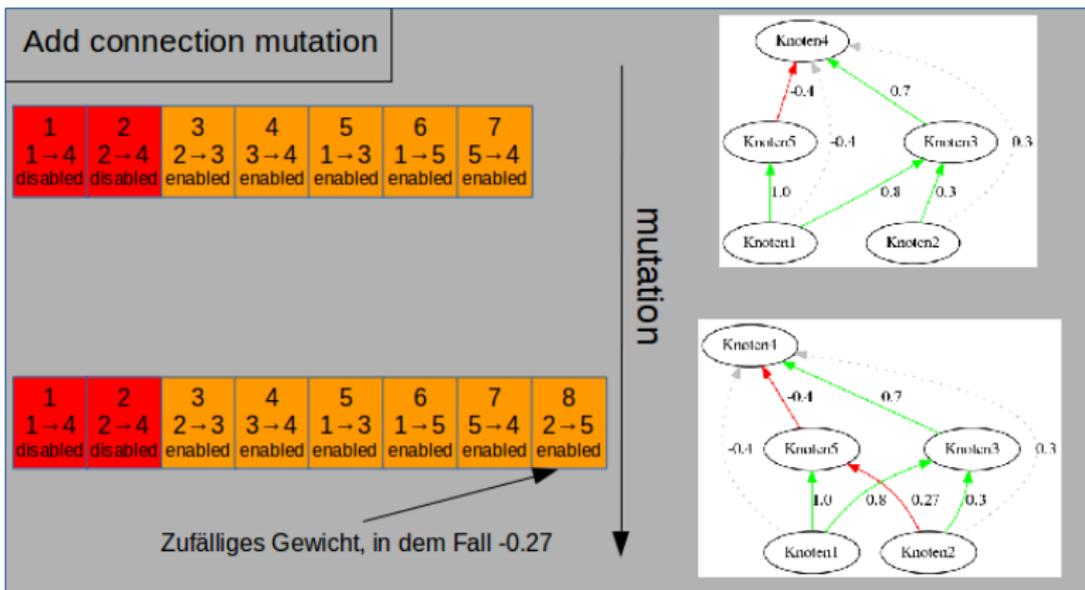


Abbildung: Kantenmutationen durch NEAT

PAARUNGSZUSTÄNDE

- ▶ Paarungszustände werden bei Kreuzungen benötigt, um zu entscheiden welche Gene aus den zu kreuzenden Genomen für den Nachkommen übernommen werden
- ▶ Paarungszustände spielen auch bei der Speziesbildung eine entscheidende Rolle (dazu gleich mehr)

Gene können folgende Paarungszustände haben:

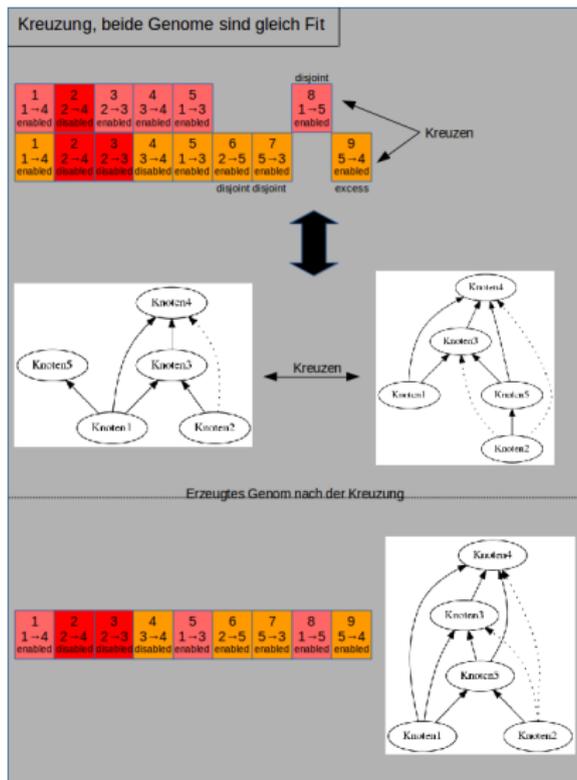
- ▶ matching genes (M) - beide Gene aus den beiden zu paarenden Genomen haben die selbe Innovationsnummer
- ▶ disjoint genes (D) - Gene, die keinen Partner und eine Innovationsnummer kleiner als die größte Innovationsnummer des kleineren Genomes haben
- ▶ excess genes (E) - Gene, die keinen Partner und eine Innovationsnummer größer als die größte Innovationsnummer des kleineren Genomes haben

KREUZUNG

Nach folgenden Regeln werden zwei Genome gekreuzt:

- ▶ Matching genes (M) werden zufällig aus einem der beiden Genome gewählt
 - ▶ Dabei unterscheiden sich lediglich der Aktivierungszustand und das Gewicht der Gene (M)
- ▶ Disjoint genes (D) und excess genes (E) werden aus beiden Genomen übernommen, wenn beide Genome etwa gleich Fit sind
- ▶ Disjoint genes (D) und excess genes (E) werden aus dem fitteren Genom übernommen, wenn beide Genome nicht gleich Fit sind

KREUZUNG - BEISPIEL



SPEZIEN

Das neuronale Netz, dessen Struktur verändert wurde oder erst entstanden ist, kann selber nicht die topologischen Innovationen aufrecht erhalten.

Damit Genome nicht

1. wegen zu geringer Fitness zu früh entfernt werden
2. durch zu rasantes Wachstum die gesamte Population übernehmen

Werden Genome in Spezien eingeteilt.

BESTIMMUNG DER MITGLIEDER EINER SPEZIES (1)

Dafür werden folgende Werte benötigt:

- ▶ Kompatibilitätsdistanz δ zwischen zwei Genomen
- ▶ Kompatibilitätsschwellwert δ_s einer Spezies, frei wählbar

Die Kompatibilitätsdistanz δ wird wie folgt bestimmt:

$$\delta = \frac{c_1 \cdot E}{N} + \frac{c_2 \cdot D}{N} + c_3 \cdot \overline{W}$$

E = Anzahl der excess genes

D = Anzahl der disjoint genes

\overline{W} = Durchschnittliche Gewichtung der matching genes

N = Anzahl der Gene im größeren Genom

c_1, c_2, c_3 = Gewichtung, frei wählbar

BESTIMMUNG DER MITGLIEDER EINER SPEZIES (2)

- ▶ Nach jeder Testphase wird die Kompatibilitätsdistanz jeden Genoms (A) zu genau einem zufälligen Genom jeder anderen Spezies gebildet
- ▶ wenn $\delta < \delta_s$ gilt wird Genom (A) der entsprechenden Spezies zugeordnet
- ▶ wurden keine geeignete Spezies für Genom (A) gefunden, so wird eine neue Spezies erzeugt

BESTIMMUNG DER GRÖSSE EINER SPEZIES (1)

- ▶ Damit eine Populationsübernahme einer Spezies ausgeschlossen ist, wird regelmäßig die Größe aller Spezies reguliert
- ▶ Durch die Bestimmung der neuen Größe stellt sich auch heraus, wie viele Kreuzungen innerhalb der Spezies durchgeführt werden und wieviele Genome (mit der geringsten Fitness) verworfen werden

Bestimmung der „adjusted fitness“ (angepasste Fitness) für jedes Genom jeder Spezies:

BESTIMMUNG DER GRÖSSE EINER SPEZIES (2)

$$f'_i = \frac{f_i}{\sum_{j=1}^{N_j} sh(\delta(i,j))}$$

f'_i = angepasste Fitness des Genoms i

f_i = alte Fitness des Genoms i

j = anderes Genom aus der selben Spezies wie i

$sh(\delta(i,j))$ = sharing function, gibt 0 zurück wenn $\delta(i,j) > \delta_s$,
ansonsten 1

BESTIMMUNG DER GRÖSSE EINER SPEZIES (3)

- ▶ Anschließend wird die durchschnittliche angepasste Fitness aller Genome berechnet
- ▶ Nun stehen alle Werte zur Verfügung um die neue Größe der Spezies zu berechnen
- ▶ somit kann auch berechnet werden, wie viele Kreuzungen durchgeführt werden müssen und wie viele Genome verworfen werden

Bestimmung der neuen Größe der Spezies:

BESTIMMUNG DER GRÖSSE EINER SPEZIES (4)

$$N'_j = \frac{\sum_{i=1}^{N_j} f_{ij}}{\bar{f}}$$

N'_j = neue Größe der Spezies

N_j = alte Größe der Spezies

f_{ij} = angepasste Fitness des Individuums i in Spezies j

\bar{f} = Durchschnittliche angepasste Fitness der gesamten Population

SUPER MARIO MIT NNS UND NEAT GEWINNEN

VORBEREITUNGEN UND START (1)

Bevor das Programm mit der Bearbeitung der neuronalen Netze startet, muss es einige Variablen initiieren

Zähler:

- ▶ Generationszähler = 0
- ▶ Innovationszähler = 1

Parameter:

- ▶ gewünschtes Ergebnis: Mario ist im Ziel (und optional: die Fitness ist maximal)
- ▶ Maximale Dauer bis zum erwünschten Ergebnis, hier unendlich
- ▶ Anzahl der Startgenome, hier 100
- ▶ Anzahl der Knotenmutationen pro Training = 5% (Wert frei wählbar)
- ▶ Anzahl der Kantenmutationen pro Training = 25% (Wert frei wählbar)

VORBEREITUNGEN UND START (2)

- ▶ Zeitliche Verhältnis von Trainings- und Testphase pro Generation, hier 80%/20%
- ▶ Fitnessbedingung, in dem Fall:
 - ▶ Distanz richtung rechts vom Startpunkt aus
 - ▶ Geschwindigkeit in der Mario nach rechts kommt
 - ▶ Anzahl der gesammelten Münzen
 - ▶ Anzahl der eliminierten Gegner
 - ▶ Höhe in der durch das Ziel gesprungen wurde

VORBEREITUNGEN UND START (3)

Datenspeicher:

- ▶ Lookuptable für Innovationsnummern der Kantengene und der Knotengene, welche diese verbinden
- ▶ Genommengen
- ▶ Spezienmengen

Nach der Definition all dieser Werte werden (in diesem Fall) 100 zufällige minimale Genome erstellt und die erste Generation kann beginnen.

START (1)

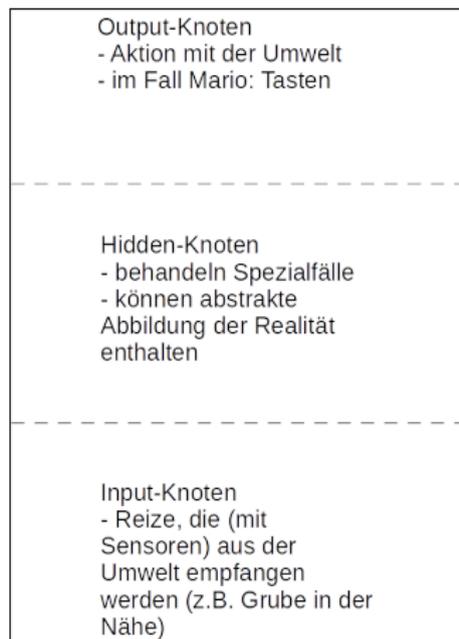
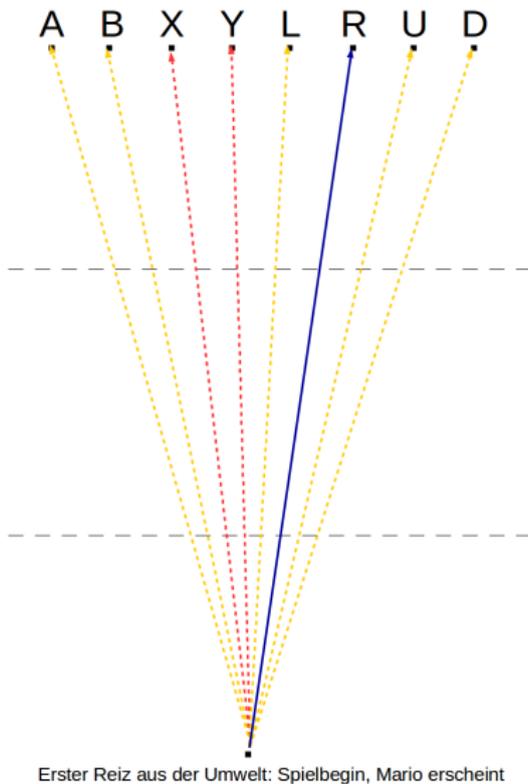


Abbildung: Bedeutung der Ebenen in Super Mario

START (2)



Erster Reiz aus der Umwelt: Spielbeginn, Mario erscheint

TRAININGSPHASE

Die Trainingsphase wird in diesem Fall (80/20) vier mal durchgeführt

Pseudocode: Trainingsphase einer Generation

```

1  ————— Trainingsphase —————
2  Fuehre Kantenmutationen fuer
3  x% Zufallsknoten in
4  jedem Genom durch
5
6  Fuehre Knotenmutationen fuer
7  y% Zufallskanten in
8  jedem Genom durch
9
10 # Kein Teil aus NEAT:
11 Trainiere die Gewichte der
12 Kanten, z.B. mit der Hebb-Regel
  
```

NEUE REIZE AUS DER UMWELT

Im Laufe der Präsentation wird immer das selbe Genom als Beispiel genommen, es entwickeln sich jedoch (zu Beginn) hundert Genome gleichzeitig. Wenn während des Trainings ein neuer Reiz aus der Umwelt entsteht (z.B. Undurchdringbares Objekt), wird ein neuer Sensor-Knoten erstellt

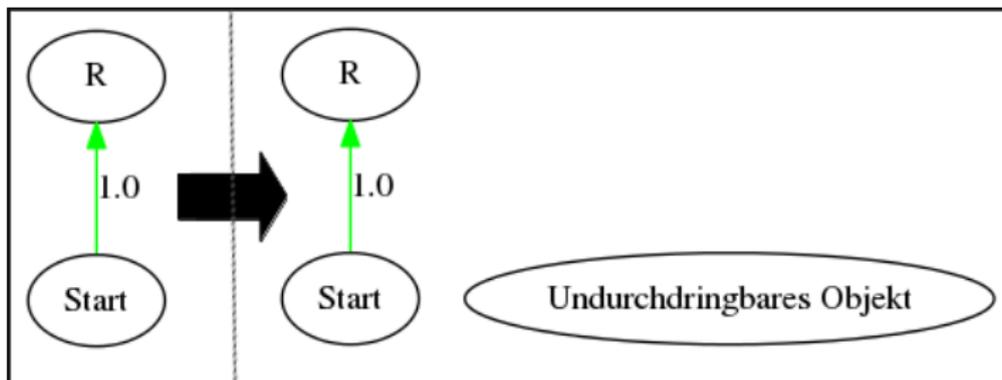


Abbildung: Erster Reiz aus der Umwelt, 7 Output-Knoten wurden in dieser Darstellung weggelassen

KANTENMUTATIONEN (ADD CONNECTION MUTATION)

- ▶ Es wurden 5% Kantenmutationen pro Training definiert
- ▶ Momentan besteht das neuronale Netz aus 2 Sensor-Knoten und 8 Output-Knoten
- ▶ 5% von $16-1=15$ Möglichen Kanten sind 0.75, aufgerundet 1 Kante
- ▶ Die beiden zu verbindenden Knoten werden zufällig gewählt

KANTENMUTATIONEN (VORHER)

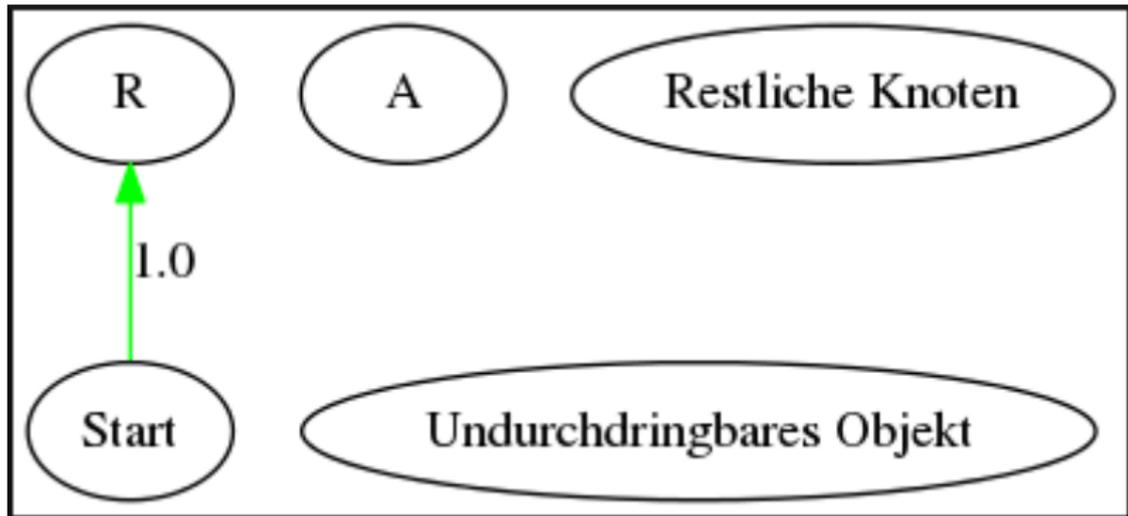


Abbildung: Super Mario Phenotyp vor der Kantenmutation

MÖGLICHE KANTENMUTATIONEN (NACHHER)

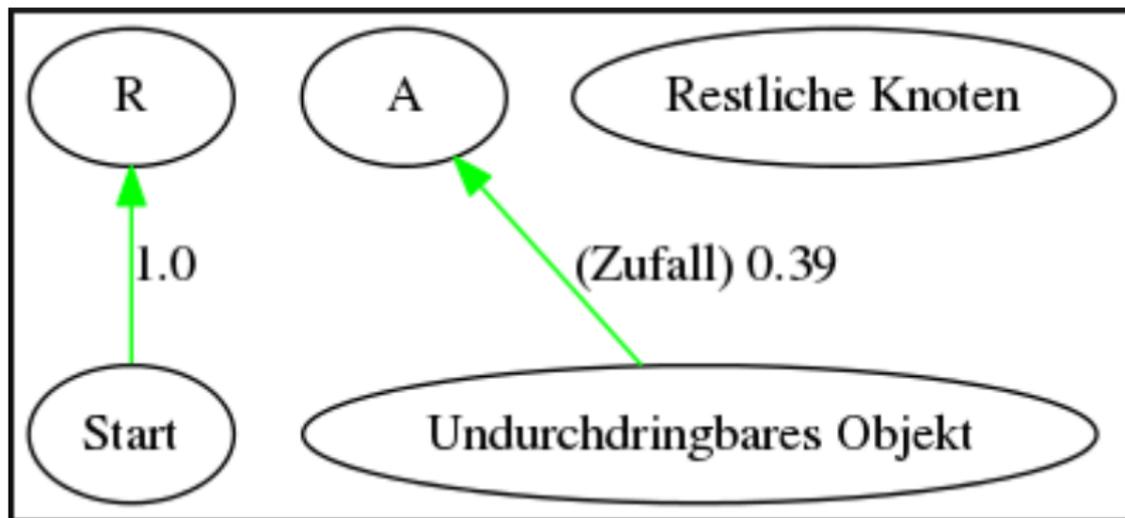


Abbildung: Super Mario Phenotyp nach der Kantenmutation

KNOTENMUTATIONEN (ADD NODE MUTATION)

- ▶ Es wurden 25% Knotenmutationen pro Training definiert
- ▶ Momentan besteht das neuronale Netz aus 2 Kanten
- ▶ 25% von 2 Kanten die aufgeteilt werden können sind 0.5 Kanten, aufgerundet 1 Kante
- ▶ Die zu zerteilende Kante wird zufällig gewählt
- ▶ Der neue Knoten ist ein Hidden-Knoten und stellt eine Abstraktion der Umwelt dar

KNOTENMUTATION (VORHER)

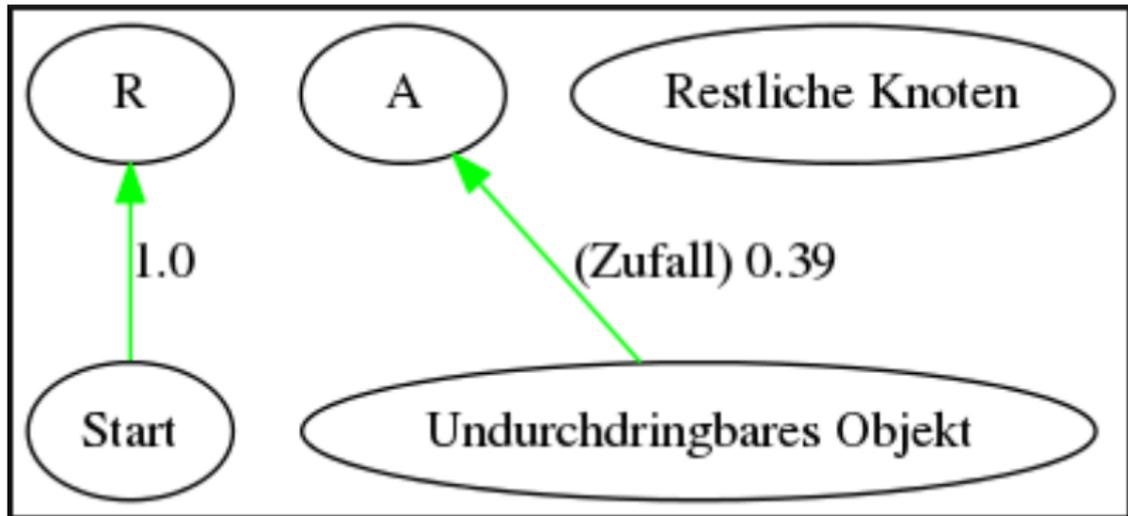


Abbildung: Super Mario Phenotyp vor der Knotenmutation

(MÖGLICHE) KNOTENMUTATION (NACHER)

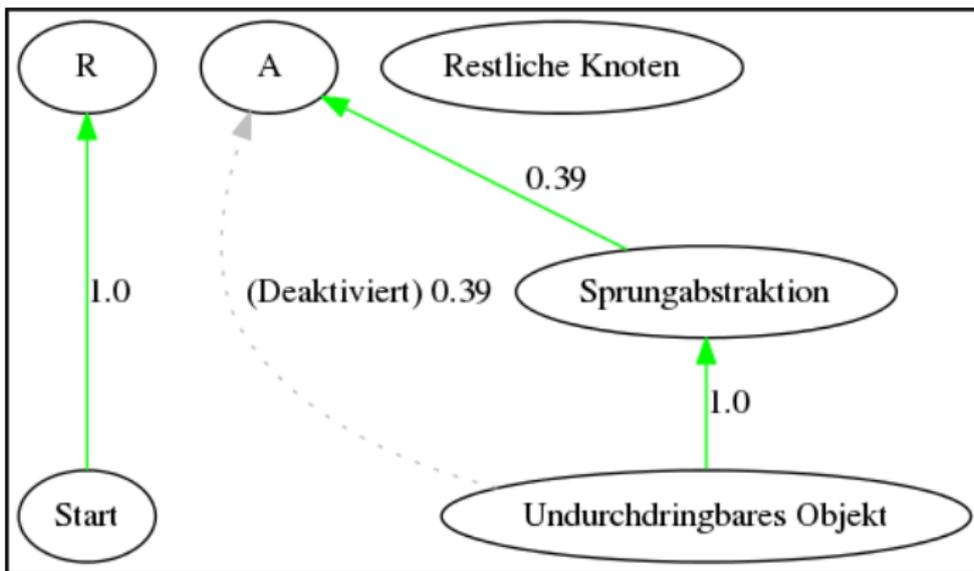


Abbildung: Super Mario Phenotyp nach einer möglichen Knotenmutation

BEISPIEL: ABSTRAKTION DURCH HIDDEN-KNOTEN

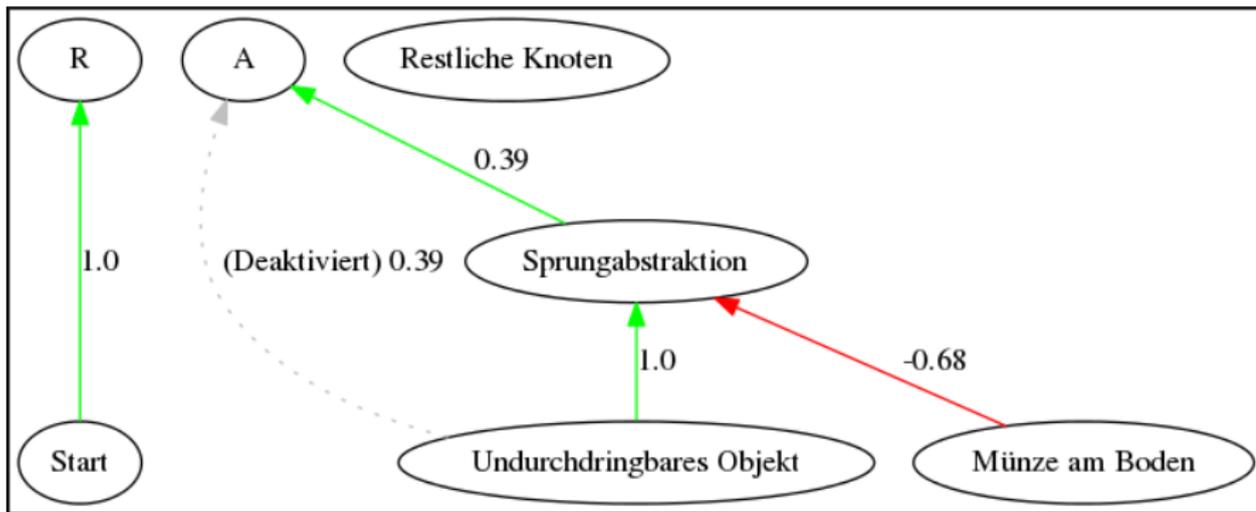


Abbildung: Sinnvolle Abstraktion durch Hidden-Knoten

TESTPHASE

Pseudocode: Testphase einer Generation

```

1  ————— Testphase —————
2  – Evaluiere Fitness fuer alle Genome
3  # Den folgenden Teil kann man nach Testphase
4  # oder vor Trainingsphase durchfuehren
5  – Ordne alle Genome einer Spezies zu
6  – Bestimme die angepasste Fitness aller Genome
7  – Bestimme die durchschnittliche
8  angepasste Fitness aller Genome
9  – Bestimme mit den errechneten Werten
10 die neuen Speziengroessen
11 – Kreuze und loesche Genome so, dass die
12 vorgegebene Speziengroesse erreicht wird
13 – Inkrementiere den Generationszaehler
  
```

SPEZIENZUORDNUNG

- ▶ Nach den im Kapitel „NEAT“ beschriebenen Kriterien werden Spezien gebildet und Genome den Spezien zugeordnet
- ▶ Die Spezienzuordnung der Genome wird in diesem Beispiel nicht berechnet
- ▶ Die neue Größe der Spezien wird in diesem Beispiel nicht berechnet
- ▶ Für Spezien die größer werden, werden nur die beste (fittesten) $x\%$ der Mitglieder gekreuzt
- ▶ Für Spezien die kleiner werden, werden die besten (fittesten) $x\%$ der Genome gekreuzt und für jede Kreuzung werden 2 der am wenigsten Fitten Genome der Spezies entfernt

BEISPIELKREUZUNG

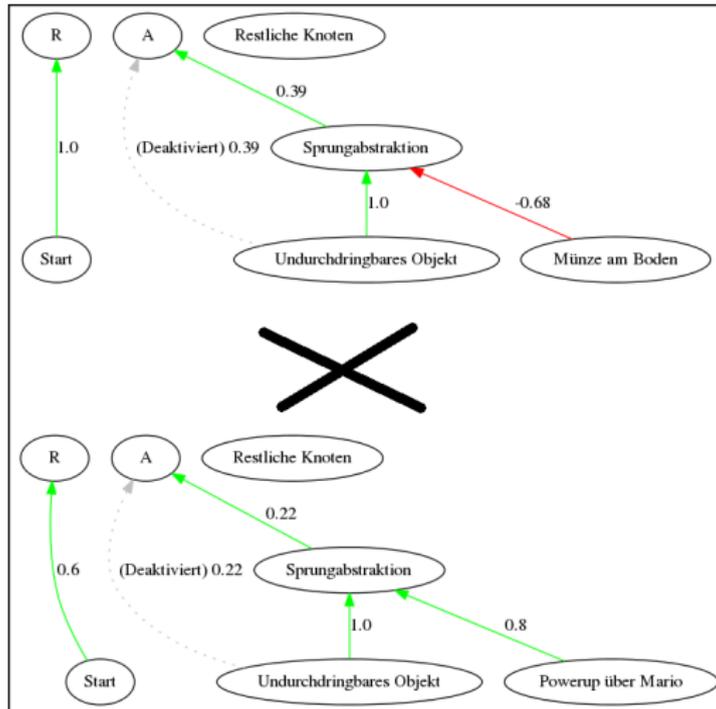


Abbildung: Beispielkreuzung

KREUZUNG ZWEIER GLEICH FITTER GENOME

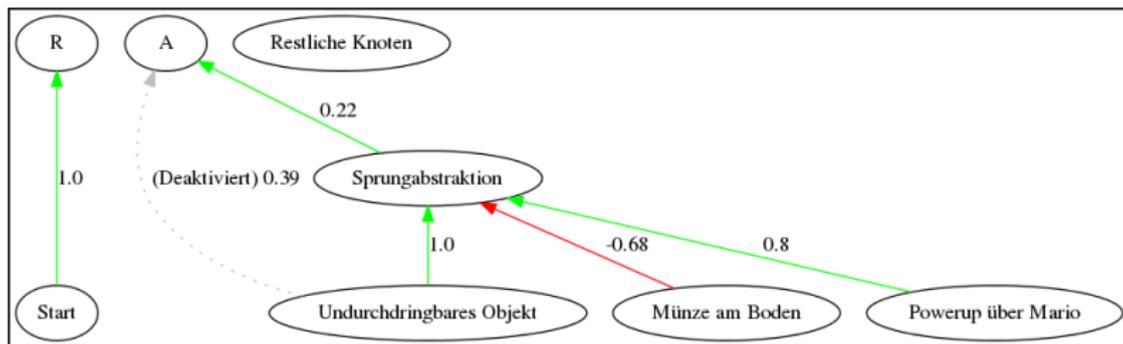


Abbildung: Ein möglicher Nachkomme der Kreuzung, wenn beide Genome gleich Fit sind

KREUZUNG ZWEIER UNGLEICH FITTER GENOME

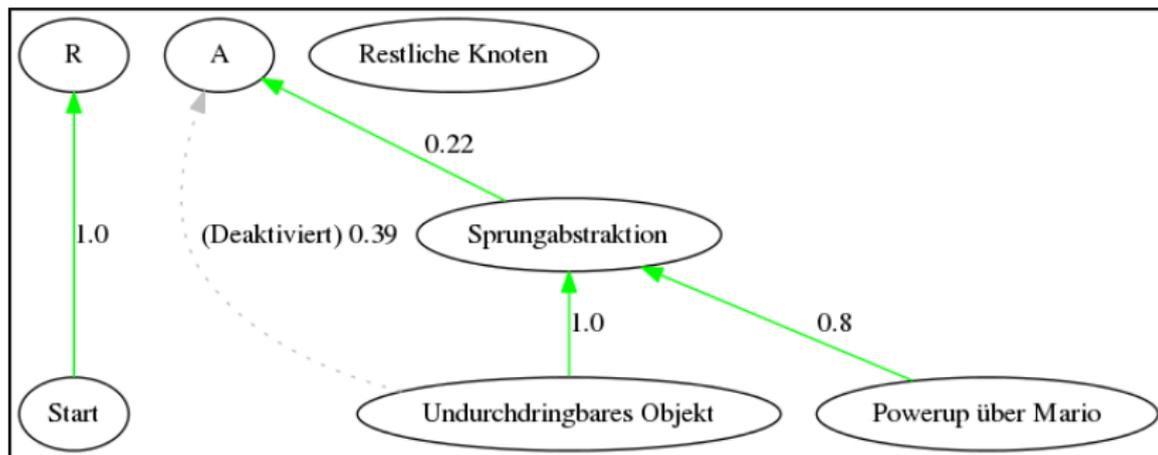


Abbildung: Ein möglicher Nachkomme der Kreuzung, wenn beide Genome ungleich Fit sind

FRAGEN, ANMERKUNGEN, KONTAKT

Für Fragen und Anmerkungen stehe ich gerne jetzt zur Verfügung sofern noch Zeit dafür übrig ist

- ▶ Alternativ können Sie mir bei Fragen oder Anmerkungen eine E-Mail an harald.heckmann@student.hs-rm.de schreiben
- ▶ Dieser Foliensatz basiert auf dem HSRM Theme von Benjamin Weiss.
- ▶ Das HSRM Theme steht unter der **GNU Public License**. Es darf also weitergegeben und modifiziert werden, sofern die Lizenzart beibehalten wird.