C:\Users\haller\Desktop\Logo_HFU.tif

Ersetzen Sie alle [ ]-Klammern durch korrekte Daten.

PS: Diese Gedankenblasen sollten vor der Abgabe natürlich alle entfernt werden!

[Art der Arbeit]

in

[Studiengang]

[Titel der Arbeit] Neuroevolution benutzerdefinierter mehrbeiniger Kreaturen

[Untertitel der Arbeit]

**Achtung**: Diese Vorlage kann Fehler enthalten! Bitte prüfen Sie die Einhaltung der Richtlinie nochmal selbst nach. Für Hinweise auf Fehler in der Vorlage bin ich Dankbar!

|  |  |
| --- | --- |
| Referent: | [Name] |
| Korreferent: | [Name] |
| Vorgelegt am: | [Datum] |
| Vorgelegt von: | [Name] |
|  | [Matrikelnummer] |
|  | [Straße und Hausnummer]  [PLZ und Stadt] |
|  | [HFU-Emailadresse] |

# Vorwort

Das Vorwort ist optional. Wenn Sie kein Vorwort in Ihrer Arbeit verwenden wollen können Sie dieses Kapitel entfernen!

# Abstract

[Englisch, 100 -120 Worte]

[Deutsch, 100 – 120 Worte]

# Inhaltsverzeichnis

[Vorwort I](#_Toc331872775)

[Abstract III](#_Toc331872776)

[Inhaltsverzeichnis V](#_Toc331872777)

[Abbildungsverzeichnis VII](#_Toc331872778)

[Tabellenverzeichnis IX](#_Toc331872779)

[Abkürzungsverzeichnis XI](#_Toc331872780)

[1. Einleitung 1](#_Toc331872781)

[2. Grundlagen 3](#_Toc331872782)

[3. [Eigene Kapitel] 5](#_Toc331872783)

[4. Ausblick 7](#_Toc331872784)

[5. Fazit 9](#_Toc331872785)

[Literaturverzeichnis 11](#_Toc331872786)

[Stichwortverzeichnis 13](#_Toc331872787)

[Eidesstattliche Erklärung 15](#_Toc331872788)

[A. [Anhang] 17](#_Toc331872789)

# Abbildungsverzeichnis

Das Abbildungsverzeichnis muss vor der Finalen Abgabe entfernt werden, wenn keine Abbildungen in die Arbeit eingefügt wurden.

# Tabellenverzeichnis

Das Tabellenverzeichnis muss vor der Finalen Abgabe entfernt werden, wenn keine Tabellen in die Arbeit eingefügt wurden.

# Abkürzungsverzeichnis

|  |  |
| --- | --- |
| [Abkürzung]  KGS  NEAT | [Ausgeschriebene Abkürzung]  Knochen-Gelenk-System  Neuroevolution of Augumented Topologies |
| ZMax | höchste Z-Position, die von der Kreatur am Ende der Simulation erreicht wurde |
|  |  |
|  |  |
|  |  |
|  |  |
|  |  |
|  |  |

# Einleitung

## Ausgangssituation

In dieser Thesis sollen mehrbeinige Kreaturen in einer Simulation selbstständig lernen Laufen, ohne dass ihnen Laufmuster gezeigt werden. Die Kreaturen laufen in Intervallen von 20 Sekunden, bevor sie auf ihre Startposition und Startpose zurückgesetzt werden. Entscheidend für die Qualität des Lernens einer Kreatur ist die höchste Z-Position, die von der Kreatur am Ende der Simulation erreicht wurde (im weiteren bezeichnet als).

Im Rahmen dieser Arbeit lernen die Kreaturen durch den genetischen Algorithmus NEAT Laufen. Es werden die Auswirkung auf das Lernverhalten verschiedener Konfigurationen und Parameter des NEAT-Algorithmus‘, die Inputs/Sensoren, (Outputs), Entlinearisierung, Fitness-Funktion und Hyperparameter betreffen untersucht.

Des weiteren wird ein Editor implementiert, in dem Kreaturen benutzerdefiniert gebaut und anschließend mit ihnen die Simulation gestartet werden können. Die Kreaturen werden aus Knochen, Gelenken und Rotationslimits gebaut und dadurch definiert. Es können von jedem Glied aus in 25 Richtungen neue Glieder gebaut werden, sowie deren Länge festgelegt werden ebenso können die Rotationslimits auf den jeweiligen X- und Y-Achsen eingestellt oder gänzlich blockiert werden. Es ist möglich sie zu serialisieren und deserialisieren, da sie in einem eigenen Format, dem KGS, gespeichert werden.

### 1.2 Zielsetzung

Ziel dieser Thesis ist es, herauszufinden, welche Konfigurationen bei mehreren sich signifikant voneinander unterscheidenden Kreaturen im KGS zu dem besten Lernerfolg führt. Dieser wird an dem Durchschnitt der ZMax‘ über die verschiedenen Kreaturen bestimmt.

Auch soll die Frage beantwortet werden, ob es möglich ist, einen Editor zu Realisieren, indem lauflernfähige dreidimensionale Kreaturen benutzerdefiniert erstellt werden können.

* Gelingt es, einen (prototypen) Editor für …

# Grundlagen

## NN

2.1.1 Neuronen/ Perceptronen

2.2 Genetische Algorithmen/ NEAT

2.2.0 Minimal Structure/ Ausgangszustand

2.2.1 Genome

2.2.2 Mutation

2.2.3 Speciation

## Physiksimulation

### Unity Physics Engine

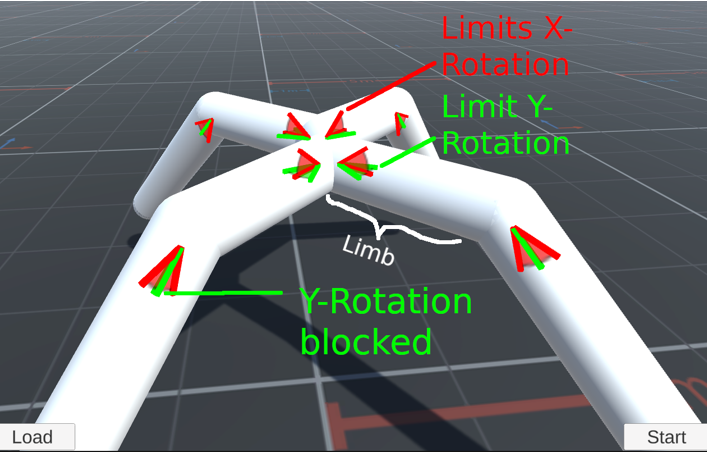
* Rigidbodies
* Configurable Joints
* (Joint Drive)

2.2.2 BodyParts (kann in Grundlagen oder in Implementierung sein)

# [Eigene Kapitel] Implementierung

## Editor

### Interface & Steuerung (auch Konzeptionelle)



### KGS System

(de-)Serialization

## 3.1.4 Editor-Kreatur zu Trainings-Kreatur

-Massen, ArticulatedJoints (settings aus BP MLAgents, BodyParts)

KGS System, DockingBalls, Problem Ausrichtung Docking Balls

## Training

Für die Implementierung gibt es an mehreren Stellen verschiedene Konfigurationen. Die Standard-Konfiguration zeichnet den Ausgangszustand aus, die Variations-Konfiguration zeichnet den zu testenden Zustand aus.

* NEAT Genom pro Kreatur-Instanz
* Einige Kreaturen pro Welle
* NEAT Outputs
  + -1 bis 1
  + Standard-Config.:
    - Steuern Zielrotationen pro Glied
      * -1: kleinster möglicher Winkel
      * 1: größter möglicher Winkel
  + Variations-Config:
    - Steuern Zielrotationen pro Glied
      * -1: kleinster möglicher Winkel
      * 1: größter möglicher Winkel
    - Steuern maximale Kräfte pro Glied
* NEAT Inputs
  + Triviale und vorhersehbare Probleme
    - Negative Werte erreichen
    - Wertebereich -1 bis 1 ausschöpfen
    - Lernrelevante Einflüsse kodieren
  + Lösungen in dieser Thesis: Verschiedene Sensoren aus der Simulation
  + Standard-Konfig.:
    - Distanz zum CoM für quantifizierbaren Bezug zur Gewichtsverteilung des gesamten KGS pro Kreatur
  + Variations-Konfig:
    - Faktor der momentanen Rotation für Bezug zur Pose des gesamten KGS pro Kreatur
  + Delinearisierung
    - Standard-Konfig.:
      * Tangens Hyperbolikus
    - Variations-Konfig:
      * Keine
* Wellen
  + Nach Dauer einer Welle -> nächste Generation, nächste Welle
  + UI für die MaxWaveTime
* Fitness
  + Standard-Konfig.:
    - Fitness = Z-Position
  + Variations-Konfig:
    - Kontinuierlich Z-Geschwindigkeit auf Fitness addieren
    - Kontinuierlich Z-Position auf Fitness addieren
    - Kontinuierlich X-Position von Fitness abziehen
    - Wenn Z-Pos zu negativ, abziehen
    - Kontinuierlich Winkel zwischen Body und forward von Fitness abziehen
* Hyperparameter
  + Standard-Konfig.:
    - Crossing aus
  + Variations-Konfig:
    - Crossing an

# [Eigene Kapitel] Methode

### 4.1 Data Collecting

- Konfigurationen

- inspirationen für konfigurationen

* Default



* RotationalFactor

Ein Bild, das Text enthält.

Automatisch generierte Beschreibung

* LinearInputs

Ein Bild, das Text enthält.

Automatisch generierte Beschreibung

* CrossingOn

Ein Bild, das Text enthält.

Automatisch generierte Beschreibung

* AlsoPunishX

Ein Bild, das Text enthält.

Automatisch generierte Beschreibung

- Welche Test Kreaturen und warum?

- Verschiedene Beinanzahl

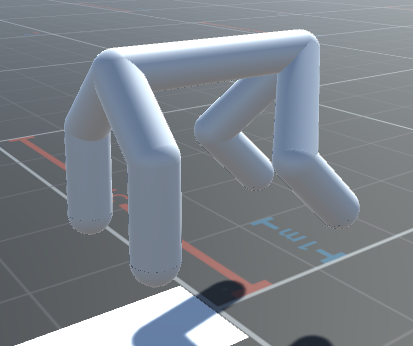
- Angelehnt an die Tierwelt

Spinne

Ein Bild, das Outdoorobjekt enthält.

Automatisch generierte Beschreibung

Hund

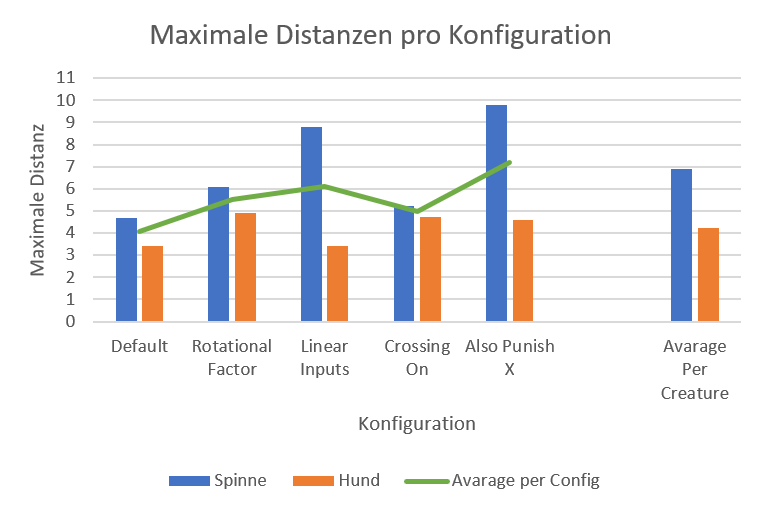


Test runs:

* Jede Konfiguration mit jeder Kreatur getestet
* Mehrere Batches pro Kreatur um Zufall zu verringern
* Pro Batch 5 Minuten Simulationszeit
* Pro Batch unterschiedlich viele Wellen
* Pro Welle wird die maximal-Distanz gespeichert

# [Eigene Kapitel] Resultate

Hier sind die „Maximale Distanzen“ Die Durchschnitte der Batch-Maxima. Die Batch-Maxima sind jeweils die Höchstwerte der Maximalen Distanzen pro Welle



Am besten Abgeschnitten hat „Also Punish X“ mit einer Durchschnittlichen Maximalen Distanz von 7,187. Darunter folgt die Konfiguration „Linear Inputs“ mit einer durchschnittlichen maximalen Distanz von 6,107 und „Rotational Factor“ mit einer durchschnittlichen maximalen Distanz von ….  
  
Der Unterschied zwischen der besten und schlechtesten Kreatur war bei Konfiguration …XY… am höchsten

* Fehlt in CSV:
  + Der Unterschied maximaler Distanzen bei verschiedenen Batches war bei Konfiguration …XY… am höchsten

# [Eigene Kapitel] Auswertung und Diskussion

### -Warum war Konfiguration xy am besten?

# Ausblick

## Als Playground - Didaktisch, Forschend und Unterhaltend

7.2 Was hätte man noch machen können

# Fazit

# Literaturverzeichnis

Fügen Sie hier ihre verwendete Literatur ein. Beachten Sie dabei die Vorgaben zu den Zitierstilen

# Stichwortverzeichnis

Das Stichwortverzeichnis ist optional. Wenn Sie kein Stichwortverzeichnis in Ihrer Arbeit verwenden wollen können Sie dieses Kapitel entfernen!

# Eidesstattliche Erklärung

Ich versichere, dass ich die vorstehende Arbeit selbständig verfasst und hierzu keine anderen als die angegebenen Hilfsmittel verwendet habe. Alle Stellen der Arbeit die wörtlich oder sinngemäß aus fremden Quellen entnommen wurden, sind als solche kenntlich gemacht.

Die Arbeit wurde bisher in gleicher oder ähnlicher Form in keinem anderen Studiengang als Prüfungsleistung vorgelegt oder an anderer Stelle veröffentlicht.

Ich bin mir bewusst, dass eine falsche Erklärung rechtliche Folgen haben kann.

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

[Ort, Datum Name]

Auch hier müssen die Platzhalter mit den korrekten Daten ersetzt werden.

# [Anhang]

Der Anhang ist optional. Wenn Sie keinen Anhang in Ihrer Arbeit verwenden wollen können Sie dieses Kapitel entfernen!