

# Praktikum Künstliche Intelligenz

Team D Rot

Aufgabe 1.2

## a) Wahl der Varianten/Optionen

- ▶ **Selektor:** Random Rejection
- ▶ **Erklärung:** Wir wählen ein zufälliges Individuum unserer Population. Der Fitnesswert muss größer oder gleich einem zweiten zufälligen Fitnesswert, welcher in einem Bereich zwischen dem schlechtesten und besten Fitnesswert der Population liegt, sein.
- ▶ **Entscheidungsgrund:** Durch die zwei Zufallskomponenten entfällt bei einer Selektion nach gewichteten Wahrscheinlichkeiten die Erstellung eines großen Pools.

## a) Wahl der Varianten/Optionen

- ▶ **Mutator:** Swap-Mutator mit Random Resetting
- ▶ **Erklärung:** Wir wählen mit einer 2 %-igen Wahrscheinlichkeit zwei verschiedene Gene innerhalb einer Sequenz und tauschen diese. Danach wird mit einer 1 %-igen Wahrscheinlichkeit jedes Gen innerhalb dieser Sequenz mutiert.
- ▶ **Entscheidungsgrund:** Eine zweite Implementation war Pflicht.

## a) Wahl der Varianten/Optionen

- ▶ **Terminator:** IndividualHigherThan
- ▶ **Erklärung:** Beendet den Algorithmus, sobald die gewünschte Genauigkeit, ausgedrückt in Scorepunkten, erreicht wird.
- ▶ **Entscheidungsgrund:** Wir wollten ein möglichst gutes Ergebnis erzielen.

## a) Wahl der Varianten/Optionen

- ▶ **Initializer:** InitRecursiveCheat
- ▶ **Erklärung:** Setzt das letzte Gen auf Bremsen, alle übrigen werden aus dem Pool genommen.
- ▶ **Entscheidungsgrund:** Unser Auto fährt meistens wieder aus der Lücke.

## b) Fitnessfunktion

Sei  $S_{max}$  perfektes Individuum (Gensequenz) mit einem Highscore von 10000.

Wir wollen unsere Individuen an  $S_{max}$  annähern.

- ▶ **NumberOfCollision:** wenn  $! = 0$ , Fitnesswert = 0;
- ▶ **DistanceFromGoal:**  $Fehlerwert * 1000$
- ▶ **AngleToGoal:**  $Fehlerwert * 2000$
- ▶ **CurrentVelocity:**  $Fehlerwert * 400$
- ▶ **Entscheidung:** Wir möchten keine Sequenzen haben, die gegen andere Autos fahren. Der Winkel ist so hoch bewertet wurden, da das Auto sonst halb in der zweiten Reihe parkt und die Distanz zum Zielpunkt soll ebenfalls möglichst gering ausfallen.

## c) Gene a)

- ▶ **GeneS:** Lenkung: Vorwärts, Beschleunigung: Vorwärt
- ▶ **GeneT:** Lenkung: Vorwärts, Beschleunigung: Rückwärts
- ▶ **GeneU:** Bremsen anziehen
- ▶ **GeneV:** Lenkung: Links, Beschleunigung: Vorwärt
- ▶ **GeneW:** Lenkung: Links, Beschleunigung: Rückwärts
- ▶ **GeneX:** Lenkung: Rechts, Beschleunigung: Vorwärt
- ▶ **GeneY:** Lenkung: Rechts, Beschleunigung: Rückwärts

## c) Gene b)

- ▶ **Entscheidung:** Jedes Gen, ausgenommen U, ist eine Kombination aus einer Richtung (1,0,-1) und der Vorwärts- bzw. Rückwärtsbeschleunigung (1,-1). Durch die Kombination aus einer Bewegung und einer Richtung entfallen unnötige Sequenzen, die nur richtungswechselnde Gene enthalten.



## d) Einschätzung zur Optimierung

Zur Optimierung könnte man das letzte Gen einer Sequenz mit Bremsen hardcoden, damit das Auto nicht wieder aus der Parklücke herausfährt. Die Berechnung des Fitnesswertes und die Rekombination bzw. Mutation könnte parallel ausgeführt werden.

## e) In Realität anwendbar?

Unsere Lösung könnte übernommen werden, falls der Fahrer viel Zeit mitbringt um eine Simulation durchzuführen und die Abweichung unserer Fitnessfunktion zur perfekten Position fast 0 beträgt. (Äußere Einflussfaktoren schließen wir aus)

Wir vermuten, dass ein hoher Grad an Parallelisierung und bessere Rechenleistung dieses Zeitproblem lösen könnte. Auf wechselnde äußere Einflussfaktoren zu reagieren würde mit dieser Implementierung nicht schaffbar sein.