

# Fachpraktikum Künstliche Intelligenz: Multiagentenprogrammierung

Alexander Lorenz, Miriam Wolf, Sebastian Loder, and Steffen Jan Jendrny

Fernuniversität Hagen, Universitätsstraße 47, 58097 Hagen  
<https://www.fernuni-hagen.de/>

## 1 Einleitung

### 1.1 Auswahl technischer Rahmenbedingungen

Für die Auswahl der technischen Rahmenbedingungen hat sich die Gruppe an der Serverprogrammierung orientiert. Diese ist in Java entwickelt und deswegen sind die Agenten ebenfalls in Java entwickelt worden. Ein weiterer Grund dafür war, dass die Schnittstelle der Programmierkenntnisse aller Gruppenmitglieder auf Java fiel. Zum Einsatz kamen verschiedene Editoren wie Eclipse[1], IntelliJ[2] oder NetBeans[3].

### 1.2 Aufteilung Gruppenmitglieder - Arbeit

**Alexander**

**Miriam**

Um einen Schritt nach vorn zu gehen oder einen Block zu zerstören, müssen die Agenten eine Aktion an den Server schicken. Um einen optimalen Weg durch die Blöcke zu finden und sich zur Endzone durchzukämpfen, muss die nächstmögliche Aktion herausgefunden werden. Miriam hat sich um die Möglichkeit gekümmert, welche Aktion die nächstmögliche ist. Sie hatte die Aufgabe, das Verhalten der Agenten in der lokalen Sicht zu bearbeiten. Die Erklärung, was die lokale Sicht des Agenten ist wird später genauer erläutert.

Weiter hatte sie die Aufgabe der Praktikumsorganisatorin inne. Hierfür musste sie die Treffen der Gruppenleiter koordinieren, die Turnierplanung übernehmen und auch die Turniere begleiten. Dazu gehörten unter anderem das Erstellen der Turnierkonfiguration oder das Starten des Servers selbst beim Turnier. Abschließend kam noch die Planung des Präsentationsnachmittags dazu und das Ausarbeiten des allgemeinen Dokumentationsteils.

**Sebastian**

**Steffen**

## **2 Softwarearchitektur**

### **2.1 Strategie**

### **2.2 Erkundung der Karte**

#### **Kreisförmige Karte**

Hier geht es darum, dass die Karte keinen Rand hat.  
Kurz beschreiben, dass wir zu Beginn eine begrenzte Karte hatten und dann eine offene Karte

### **2.3 Entscheidungsverhalten der Agenten**

#### **Rollen**

Die Rollen werden in der Serverkonfiguration vorgegeben. In den Turnieren standen fünf Rollen bereit: default, worker, constructor, explorer und digger. Die Rollen haben verschiedene Aktionen, welche sie ausführen dürfen. Zu Beginn haben alle Agenten die *default*-Rolle inne. Diese enthält alle Standardaktionen wie *move*, *rotate*, *skip*, *adopt*, *detach* und *clear*. Um nach verschiedenen Dingen suchen zu können, benötigen die Agenten die Rolle *explorer*. Mit diesen kann ein *survey* nach einem Dispenser gemacht werden. Sobald die Agenten die Aufgabe bekommen einen Block abzuholen, müssen sie in die Rolle *worker* wechseln. Damit können sie Blöcke aufnehmen, sich mit anderen Agenten verbinden und auch die Aufgaben abgeben. Die Besonderheit ist, dass alle Rollen die Aktionen der default-Rolle erben. Die Standardaktionen besitzen somit alle anderen Rollen ebenfalls.

Die Rolle *constructor* und *digger* hat die Gruppe nicht weiter verfolgt, da diese für unsere Strategie nicht notwendig war.

#### **Aufgaben**

NextTaskPlanner umgesetzt

Liste der Pläne: exploreMap,  
goToDispenser,  
goToGoalzone,  
goToRolezone,  
solveTask,  
surveyDispenser,  
surveyGoalZone,  
surveyRoleZone,  
surveyRandom,  
connectToAgent,

discoverMapSize,  
cleanMap,

### Wegfindung

Random  
Spirale  
Manhattan  
A\*

### Gruppenbildung

Gruppenbildung

## 2.4 Globale und lokale Sicht

Die Sicht der Agenten können in eine globale und eine lokalen Sicht unterschieden werden. Die globale Sicht besteht aus einer gespeicherten Karte pro Agent, die sich durch die Bewegungen des Agenten auf der Karte erweitert. Hier werden die verschiedenen Dinge wie Dispenser, Blöcke oder Zonen gespeichert. Sobald sich die Agenten in Gruppen finden, werden die Karten synchronisiert. Eine genauere Erklärung zu den Karten befindet sich im Kapitel 2.2.

Die lokale Sicht des Agenten ist auf eine festgelegte Größe beschränkt. Die Sichtweite des Agenten wird in der Serverkonfiguration festgelegt. Bei einer Sichtweite von 5 sieht der Agent 5 Kästchen nach links, rechts, oben und unten, wie in Abbildung 1 dargestellt.



**Fig. 1.** Sicht des Agenten

Im ersten Schritt ermittelt der Agent einen möglichen Weg, wie er zu seinem Ziel gelangt. Hierfür wird der entsprechende Wegalgorithmus verwendet, welcher in Kapitel 2.3/ genauer erläutert wird. Sobald der Weg ermittelt wurde und bevor ein Schritt des Weges beschritten wird, prüft der Agent auf Aktionen, welche vorher ausgeführt werden müssen. Diese Aktionen können einen Rollenwechseln beinhalten, ungenutzte Blöcke fallen lassen, einen Block vom Dispenser anfordern oder einen Block aufnehmen, eine Aufgabe in der Endzone abgeben oder sich mit einem anderen Agenten verbinden. Diese Aktionen sind Momentaufnahmen, die der Agent ausführen muss, bevor er zu einem neuen Ziel geht. Wenn keine dieser Aktionen passt, wird der Agent den Weg zu seinem Ziel gehen. Hier kommt nochmals eine Entscheidungsmöglichkeit für den Agenten in Frage. Ist der nächste Schritt frei und ich kann den Weg gehen, dann geht er diesen. Sollte aber beispielsweise ein Block in der Richtung sein, in die der Agent gehen möchte, so muss er diesen zunächst zerstören. Wenn ein Agent in der Richtung steht, so wird ein Weg um diesen Agenten herum erstellt. In Abbildung 1 wäre der Weg in Richtung Westen durch einen Block gehindert, sodass er diesen zunächst zerstören muss, bevor er in diese Richtung gehen kann.

Zusammengefasst muss der Agent in jedem Schritt entscheiden, ob es eine Aktion gibt, die gerade notwendig ist, wie einen Block aufzunehmen oder ob der Schritt, den er gehen möchte, möglich ist.

## 2.5 Synchronisation und Kommunikation

Die Kommunikation der Agenten funktioniert, sobald sie sich in einer Gruppe befinden. Die Synchronisation der Gruppen ist in Kapitel 2.3 genauer beschrieben. Für die Kommunikation wurde eine Schnittstelle entwickelt, welche die Nachricht, den Senderagenten und den Empfängeragenten in einer Nachrichtenbox bereithält. Stehen zwei Agenten um einen Dispenser und beide möchten einen Block anfordern, so wird der Agent zunächst prüfen, ob eine Nachricht für ihn vorliegt. Ist dies nicht der Fall, untersucht der Agent, ob andere Agenten in der Nähe des Dispensers stehen. Wenn die Prüfung erfolgreich ist, so wird er eine Nachricht an den Agenten senden und dieser wird dann warten, bis der Dispenser frei ist. Er selbst stellt eine Anfrage an den Dispenser und nimmt den Block dann auf.

## 3 Turniere

### 3.1 Gedächnisstütze

**Turnier 1** Das erste Turnier wurde nach einer ziemlich kurzen Entwicklungszeit abgehalten. Um die Erfahrung eines Turniers aber nicht zu verlieren, nahm das Team dennoch teil. Zu diesem Zeitpunkt waren die Agenten gerade in der Lage, die Karte zu erkunden. Der Fokus der ersten Entwicklungszeit lag darauf, die Karte zu erkunden, sich mit den Tasks und den Gegebenheiten des Turniers vertraut zu machen. Aufgaben konnten zu dem Zeitpunkt ebenfalls keine abgegeben werden.

Das Ziel für das nächste Turnier war es, die Blöcke zu zerstören und zu den verschiedenen Zonen und Dispensern zu gehen. Eine bessere Wegfindung war ebenfalls notwendig.

**Turnier 2** Beim zweiten Turnier war die Wegfindung anders, aber immernoch simpel. Der Agent ist die Karte mit einem Spiralförmigen Weg abgelaufen und hat somit verschiedene Dinge erkannt. Sobald er einen Dispenser in seinem Sichtfeld hatte, ging er hin und die Blöcke wurden aufgenommen. Die ersten Entwicklungsschritte des A\* waren ebenfalls weniger erfolgreich. Die Bewegung des Agenten war das größte Problem.

Für das nächste Turnier sollten die Agenten eine performante Wegfindung erhalten, die unnötigen Blöcke fallen lassen und den Task klüger auswählen.

**Turnier 3** Im Turnier 3 konnte der Agent schon die Tasks nach Rentabilität auswählen. Das herausfinden von Dispensern und Zonen wurde mittels survey umgesetzt. Der Agent konnte sich den Weg durch die Blöcke schlagen und den Block in die korrekte Position drehen. In das Abgeben von Tasks wurde vorher viel Zeit investiert. Durch technische Probleme am Turniertag wurden leider wieder keine Punkte gemacht. Es war ziemlich enttäuschend für die Gruppe. Fürs vierte Turnier soll es endlich Punkte regnen.

**Turnier 4** Im vierten Turnier

**Turnier 5** Beim Vorletzten Turnier hat die Gruppe schon an der Entwicklung der 2er Tasks gearbeitet. Da die Turniere vorher jedoch so schlecht liefen, hat sich die Gruppe entschieden, den Agenten mit 1er Tasks zu starten. Durch die veränderte Konfiguration sind wenige 1er Tasks entstanden und die Agenten haben leider wenige Punkte gemacht.

Fürs letzte Turnier sollten die 2er Tasks unbedingt funktionierten und die Verbindung von zwei Agenten.

**Turnier 6** Um beim Abschlussturnier nicht letzter zu werden, wurden vorher noch einige Stunden investiert und es hat sich gelohnt. Die Entwicklung der 2er Tasks war erfolgreich, die Agenten konnten miteinander kommunizieren und sich miteinander Verbinden. Die Umschaltung zwischen 1er und 2er Tasks hat sehr gut funktioniert und die Agenten haben sich auch noch selten gegenseitig blockiert.

Alles in allem war es das beste Turnier des Teams und zur großen Erleichterung der Teammitglieder haben die Agenten einiges an Punkten geholt

### 3.2 Problemerkennung

Problemerkennung und Verständnisaufbau

Alexander, Miriam, Sebastian, Steffen

### **3.3 Verbesserungsmöglichkeiten**

Kommunikation im Team, Mögliche Strategische Änderungen

### **3.4 Lösungsstrategien**

Optimierung der Lösungsstrategien

### **3.5 Interaktion mit/gegen andere Agenten**

Optimierung der Strategie zur Handhabung gegnerischer Agenten

## **4 Fazit**

### **4.1 Schwierigkeiten**

### **4.2 Ausblick und weiter Möglichkeiten**

## **5 FAQ**

Hier stehen wir Frage und Antwort :)

## **References**

1. <https://www.eclipse.org/>
2. <https://www.jetbrains.com/idea/>
3. <https://netbeans.apache.org/>