



Fachbereich Technik und Informatik
Herbstsemester 2012

Bachelor Thesis - AI Bot für Computerspiele

Pflichtenheft

Studierende: Lukas Kuster
Stefan Käser

Betreuung: Dr. Jürgen Eckerle

Experte: Dr. Federico Flückiger

Datum: 10. Oktober 2012
Version: V01.01



Inhaltsverzeichnis

1	Allgemeines	1
1.1	Zweck des Dokuments	1
1.2	Ausgangslage	1
1.3	Ziel des Moduls 7321 Bachelor Thesis	1
1.4	Aufgabenstellung	1
2	Ziele	3
2.1	Funktionale Anforderungen	3
2.2	Nicht funktionale Anforderungen	4
2.3	Abgrenzungskriterien	4
3	Rahmenbedingungen	5
3.1	Spielbeschrieb	5
3.2	Verwendete Software	6
3.3	Verwendete Hardware	6
4	Projektorganisation	7
4.1	Beteiligte Personen	7
4.2	Projektmeetings	7
4.3	Dokumentation	7
4.4	Abgabe	7
4.5	Zeitplan	7





1 Allgemeines

1.1 Zweck des Dokuments

Das Pflichtenheft beschreibt die Ziele, welche mit der angestrebten Lösung der Aufgabe der Bachelor Thesis zu erreichen sind, sowie Zeitplan und Inhalt der Dokumentation.

1.2 Ausgangslage

Im vorgängigen Modul Projekt 2 haben wir Vorarbeiten zur Bachelor Thesis durchgeführt. Wir haben uns in das Thema künstliche Intelligenz eingearbeitet und uns die AI Challenge Ants genauer angeschaut. Die Pfadsuche wurde vertieft angeschaut. Die Algorithmen A* und HPA* wurden implementiert. Wir haben auf einen modularen und strukturierten Aufbau unseres Bots geachtet, damit die Bachelorarbeit auf dem Code aufbauen kann.

1.3 Ziel des Moduls 7321 Bachelor Thesis

Dieses Modul hat zum Ziel, dass die Studierenden

- selbständig und verantwortungsbewusst in kleinen Gruppen an einem Projekt arbeiten können;
- die erworbenen Fähigkeiten aus dem Schwerpunktfach bzw. aus dem Studium im Projekt anwenden können;
- die grundsätzlichen Prinzipien des Ablauf eines Software-Projekts in einem grösseren konkreten Projekt selbstständig einsetzen können;
- in der Lage sind, einen umfassenden Bericht zu ihrer Arbeit zu verfassen, die Resultate mündlich zu präsentieren und ihre Arbeit zu verteidigen.

Quelle: <http://www.ti.bfh.ch/fileadmin/modules/BTI7321-de.xml>

1.4 Aufgabenstellung

Jedes Jahr findet der von der Universität Waterloo veranstaltete Wettbewerb AI Challenge statt. Ziel des Wettbewerbs, der im Turniermodus durchgeführt wird, ist es, ein KI-Programm zu entwickeln, das ein speziell für diesen Wettbewerb entworfenes Computerspiel bestmöglich meistert. Im letztjährigen Turnier galt es im Computerspiel Ants eine Ameisenkolonie zu steuern.

In dieser Bachelorarbeit soll auf der bestehenden Projektarbeit aufgesetzt werden, in der ein Programm für das letztjährige Computerspiel Ants entwickelt wurde. Ziel dabei ist es nicht nur ein ganz konkretes Computerspiel zu meistern, sondern auch allgemein verwendbare KI-Methoden, wie beispielsweise die Pfadsuche, im Rahmen eines Frameworks zur Verfügung zu stellen. Dabei bleibt noch offen, ob das bestehende Computerspiel Ants weiterentwickelt wird oder ob das aktuelle Computerspiel des AI Challenge 2012 realisiert wird. Ein Schwerpunkt dieser Bachelorarbeit könnte auf der Verfeinerung von Pfadsuchverfahren und auf der Erprobung und Realisierung von Influence-Maps liegen.

Quelle: <https://www2.ti.bfh.ch/fbi/2013/Studienbetrieb/BaThesisHS12/aufgabestellungen/IERJ1-3-12-de.html>





2 Ziele

Der im Rahmen von Projekt 2 entwickelte Bot soll um Logik für taktische und strategische Entscheidungen und koordinierte Bewegung erweitert werden.

2.1 Funktionale Anforderungen

2.1.1 Musskriterien

Der Bot unterscheidet zwischen diversen Aufgaben:

- Nahrungsbeschaffung
- Angriff
- Verteidigung
- Erkundung

Der Bot identifiziert zur Erfüllung dieser Aufgaben konkrete Ziele, wie z.B.:

- Gegnerische Hügel angreifen, was bei Erfolg den Score erhöht und das eigentliche Ziel des Spiels ist.
- Isolierte gegnerische Ameisen angreifen.
- Schwachstellen in der gegnerischen Verteidigung ausnutzen.
- Engpässe im Terrain sichern bzw. versperren.
- Konfliktzonen, d.h. viele Ameisen auf einem engen Raum, erkennen und entsprechend reagieren.

Die Auswahl von Taktik und Strategie basiert auf der Bewertung der Situation auf dem Spielfeld, z.B. anhand folgender Kriterien:

- Dominante/unterlegene Position
- Sicherheit verschiedener Gebiete des Spielfelds (eigener/gegnerischer Einfluss)
- Konfliktpotenzial in verschiedenen Gebieten des Spielfelds

Anhand der Situationsbeurteilung werden die unterschiedlichen Aufgaben entsprechend gewichtet. Stark gewichtete Aufgaben erhalten mehr Ressourcen (Ameisen) zur Durchführung.

Die Situationsbeurteilung fließt auch in die taktische Logik ein, wie folgende Beispiele illustrieren:

- Bei der Pfadsuche wird die Sicherheit der zu durchquerenden Gebiete berücksichtigt
- In Kampfsituationen kann der Bot die Ameisen in Formationen gliedern, die geeignet sind, eine lokale Überzahl eigener gegenüber gegnerischen Ameisen zu erzeugen
- Beim Aufeinandertreffen mit gegnerischen Ameisen wird entschieden, ob angegriffen, die Stellung gehalten oder geflüchtet wird.



2.1.2 Kannkriterien

Das Verhalten des Bots ist konfigurierbar, so dass zum Beispiel ein "agressiver" Bot gegen einen defensiven Bot antreten kann.

2.2 Nicht funktionale Anforderungen

2.2.1 Musskriterien

Modularer Aufbau für eine gute Testbarkeit der Komponenten.

Wichtige Funktionen wie die Pfadsuche und die Berechnung von Influence Maps sollen in separaten Modulen implementiert werden, damit sie auch von anderen Projekten verwendet werden könnten.

Die Codedokumentation ist vollständig und dient der Verständlichkeit.

2.2.2 Kannkriterien

Für die wiederverwendbaren Module wird jeweils ein kleines Tutorial geschrieben, wie die Module verwendbar sind.

2.3 Abgrenzungskriterien

Da der Wettbewerb Ants AI Challenge bereits beendet ist, ist es uns nicht möglich dem Wettbewerb teilzunehmen.



3 Rahmenbedingungen

3.1 Spielbeschrieb

3.1.1 Der Wettbewerb

Die AI Challenge¹ ist ein internationaler Wettbewerb des University of Waterloo Computer Science Club der im Zeitraum Herbst 2011 bis Januar 2012 zum 3. Mal stattgefunden hat. Das Spiel ist ein zugbasiertes Multiplayerspiel in welchem sich Ameisenvölker gegenseitig bekämpfen. Ziel einer AI-Challenge ist es, einen Bot zu schreiben, der die gegebenen Aufgaben mit möglichst intelligenten Algorithmen löst. Die zu lösenden Aufgaben der Ants AI Challenge sind die Futtersuche, das Explorieren der Karten, das Angreifen von gegnerischen Völkern und deren Ameisenhaufen sowie dem Schützen des eigenen Ameisenhaufens.

3.1.2 Wettbewerbsteilnahme

Der Wettbewerb ist bereits beendet, wir können nicht mehr aktiv daran teilnehmen. Sowohl die Spielengine inkl. Map-Generator als auch etliche teilnehmende Bots stehen aber zum Download zur Verfügung. Damit ist gewährleistet, dass wir unseren Bot unter Wettbewerbsbedingungen testen können.

3.1.3 Spielregeln

Nachfolgend sind die wichtigsten Regeln, die während dem Spiel berücksichtigt werden müssen, aufgelistet.

- Pro Zug können alle Ameisen um ein Feld (vertikal oder horizontal) verschoben werden.
- Pro Zug steht insgesamt eine Rechenzeit von einer Sekunde zur Verfügung. Es dürfen keine Threads erstellt werden.
- Bewegt sich eine Ameise in die 4er Nachbarschaft eines Futterpixel, wird dieses eingesammelt. Beim nächsten Zug entsteht bei dem Ameisenhägel eine neue Ameise.
- Die Landkarte besteht aus passierbaren Landpixel sowie unpassierbaren Wasserstellen.
- Ein Gegner wird geschlagen, wenn im Kampfradius der eigenen Ameise mehr eigene Ameisen stehen als gegnerische Ameisen im Kampfradius der Ameise die angegriffen wird.
- Ein Gegner ist ausgeschieden wenn alle seine eigenen Ameisenhägel vom Gegner vernichtet wurden. Pro verlorenem Hügel gibt es einen Punkteabzug. Pro feindlichen Hügel, der zerstört wird gibt es zwei Bonuspunkte.
- Steht nach einer definierbaren Zeit (Anzahl Züge) kein Sieger fest, wird der Sieger anhand der Punkte ermittelt.

Die ausführlichen Regeln können auf der Webseite nachgelesen werden: <http://aichallenge.org/specification.php>

¹<http://www.aichallenge.org>



3.1.4 Schnittstelle

Die Spielschnittstelle ist simpel gehalten. Nach jeder Spielrunde erhält der Bot das neue Spielfeld mittels String-InputStream, die Spielzüge gibt der Bot dem Spielcontroller mittels String-OutputStream bekannt. Unser MyBot leitet von der Basis-Klasse Bot² ab. Ein Spielzug wird im folgendem Format in den Output-Stream gelegt:

o <Zeile> <Spalte> <Richtung>

Beispiel:

o 4 7 W

Die Ameise wird von der Position Zeile 4 und Spalte 7 nach Westen bewegt.

Der Spielcontroller ist in Python realisiert, der Bot kann aber in allen gängigen Programmiersprachen wie Java, Python, C#, C++ etc. geschrieben werden.

3.2 Verwendete Software

Als Entwicklungsumgebung wird Eclipse verwendet. Die Programmierung erfolgt in Java. Die Bachelorarbeit baut auf dem Codestand des Projekt 2 auf. Dieser Stand basiert auf dem Java-Starter-Paket, welches wir von der AI-Challenge Homepage herunter luden und weiterentwickelten. Das Starter-Paket beinhaltet alle nötigen Komponenten um ein Spiel durchzuführen und anzuschauen. Als Versionskontrolle verwenden wir git mit einem zentralen Repository auf github.com. Die Spielengine ist in Python geschrieben. Zum Ausführen braucht es dadurch eine aktuelle Version von Python. Zum Konfigurieren und Starten des Programms verwenden wir zudem ANT-Skripts.

3.3 Verwendete Hardware

Es ist keine spezielle Hardware von Nöten. Das Spiel kann auf einem handelsüblichen Computer / Laptop ausgeführt werden.

²Die Klasse ist im Code unter ants.bot.Bot.Java auffindbar



4 Projektorganisation

4.1 Beteiligte Personen

Studierende:

Lukas Kuster *kustl1@bfh.ch*

Stefan Käser *kases1@bfh.ch*

Betreuung:

Dr. Jürgen Eckerle *juergen.eckerle@bfh.ch*

Experte:

Dr. Federico Flückiger *federico.flueckiger@bluewin.ch*

4.2 Projektmeetings

- Es findet ein Treffen mit dem Betreuer alle 1-2 Wochen statt.
- Ein Treffen mit dem Experten findet am Anfang und am Ende der Arbeit statt, oder auf Wunsch des Experten.

4.3 Dokumentation

Die Dokumentation soll sich am Aufbau und Inhalt des Berichts aus dem Projekt 2 anlehnen.

- Das Dokument beschränkt sich auf das Wesentliche.
- Verwendete AI-Techniken werden erläutert
- Entscheidungen und deren Grundlagen sind dokumentiert.
- Testberichte dokumentieren die durchgeführten Modultests.
- Klassendiagramme sollen einen oberflächlichen Detaillierungsgrad haben, so dass das Wichtigste auf den ersten Blick sichtbar ist.
- Anleitung zum Ausführen eines Spiels

4.4 Abgabe

Folgende Lieferobjekte werden am Ende der Arbeit abgegeben.

- Dokumentation
- Sourcecode

4.5 Zeitplan

Die Projektarbeit richtet sich nach folgendem Zeitplan:

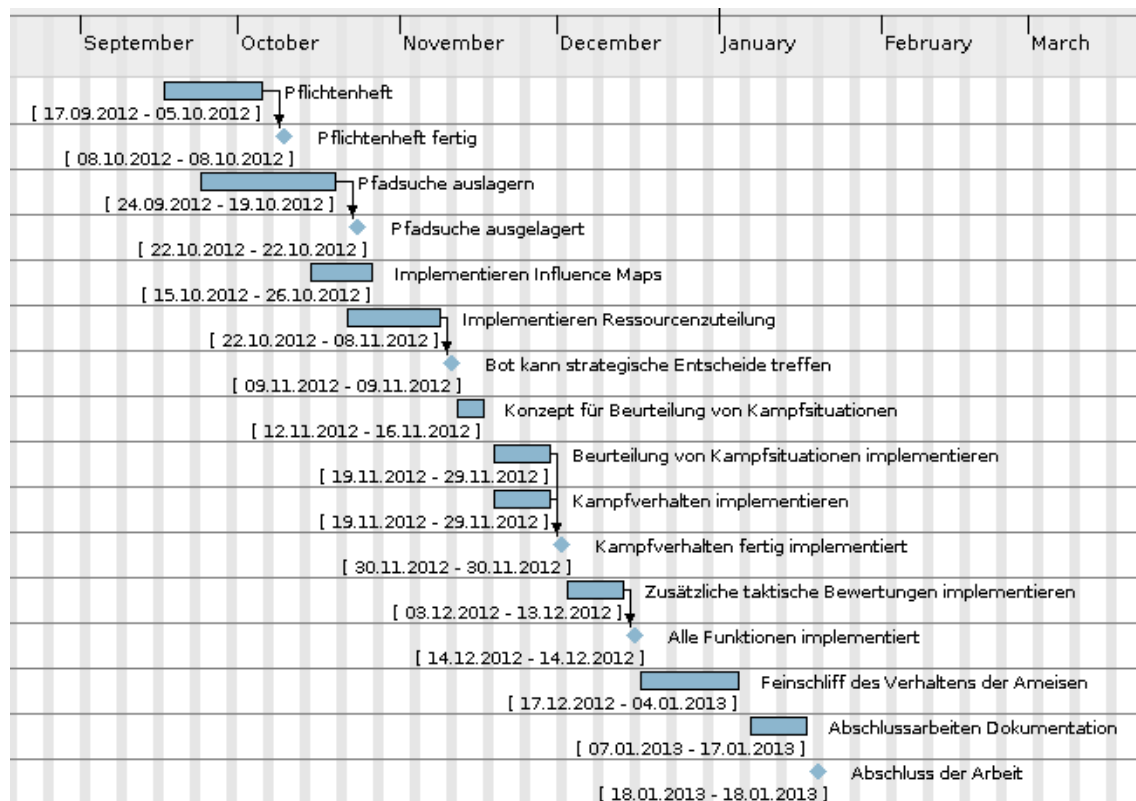


Abbildung 4.1: Projektablauf