
POKERBOT

Bachelor Thesis

Pflichtenheft

eingereicht an der

Berner Fachhochschule

Fachbereich Technik und Informatik

Autoren

Steven Heller, Bruno Wegmüller, Christian Zweili

Referenten

Dr. Bernhard Anrig

Dr. Jürgen Eckerle

eingereicht am

03. Oktober 2010

Inhaltsverzeichnis

1	Einführung.....	4
1.1	Aufgabenstellung.....	4
1.2	Situationsanalyse.....	5
1.2.1	Definition Pokerbot.....	5
1.2.2	Einsatzgebiete und Abstraktion.....	5
1.2.3	Stand der Forschung.....	6
1.3	Weltklasse-Pokerbots.....	7
1.3.1	Poki.....	7
1.3.2	PsOpti.....	7
1.3.3	BluffBot.....	8
1.3.4	Hindernisse zum Weltklasse Bot.....	8
1.4	Komponenten eines Pokerbots.....	9
1.4.1	Analyse der Handstärke.....	9
1.4.2	Analyse des Handpotentials.....	9
1.4.3	Biet-Strategie.....	9
1.4.4	Bluffen.....	10
1.4.5	Opponent Modelling.....	10
1.5	Problemstellung.....	11
1.5.1	Weitere Problemstellungen.....	12
1.6	Motivation.....	12
1.7	Vorausgegangene Arbeiten.....	12
1.8	Limit oder No-Limit Realisation.....	13
2	Zielbestimmung.....	14
2.1	Übersicht.....	14
2.2	Muss Kriterien.....	15
2.2.1	Implementation einer lauffähigen Umsetzung eines Pokerbots.....	15
2.2.2	Effiziente Implementierung.....	16
2.2.3	Qualitätsmessung.....	17
2.2.4	Abgabe-Kriterien Modul Bachelor Thesis.....	18
2.2.5	Erstellen eines Schlussberichts.....	18
2.2.6	Erstellen einer Präsentation am Ende des Semesters.....	18
2.2.7	Erstellen einer Projektbeschreibung 1 A4 Seite.....	18
2.2.8	Führen eines Laborjournals.....	18
2.3	Kann Kriterien.....	19
2.3.1	Teilnahme bei der 2011 Annual Computer Poker Competition.....	19
2.3.2	Erstellung einer interaktiven Spielumgebung.....	19
2.3.3	Verschiedene Datenstrukturen für den gelernten Spielbaum.....	19
2.4	Mögliche Stärke unseres Bots.....	19
3	Planung und Organisation.....	20
3.1	Organisationsstruktur.....	20
3.1.1	Verteilung der Rollen.....	20

3.2	Projektplanung.....	20
3.3	Meilensteine	21
3.3.1	Laborjournale	21
4	Produkteinsatz	22
4.1	Anwendungsbereiche	22
4.2	Zielgruppe.....	22
4.3	Betriebsbedingungen.....	22
4.4	Lizenz	22
5	Produktumgebung	23
5.1	Hardware	23
5.2	Software	23
6	Entwicklungsumgebung.....	24
6.1	Testing und Development.....	24
6.2	Dokumentation	24
7	Literatur und Quellen.....	25
7.1	Literaturverzeichnis.....	25
8	Glossar.....	26

1 Einführung

1.1 Aufgabenstellung

Diese Bachelor-Arbeit hat zum Ziel, eine lauffähige und performante Implementation eines Poker-Bots zu realisieren.

Charakteristisch für Poker ist, dass der Spielverlauf von der Kartenzuteilung und damit vom Zufall abhängig ist und jeder Spieler nur unvollständige Information besitzt (da er nur seine eigene Karten und das Board kennt). Poker entzieht sich damit einem klassischen Brute-Force-Ansatz, wie man ihn bei Schach und vielen anderen Strategiespielen erfolgreich anwenden konnte. Das derzeit stärkste Pokerprogramm basiert auf einem Lernalgorithmus, dem Counterfactual Regret Minimization Algorithmus.

In der vorhergegangenen Projektarbeit wurden die für eine Realisierung notwendigen theoretischen Grundlagen erarbeitet und anhand eines einfacheren Spieles erfolgreich erprobt.

In der Bachelor-Arbeit soll dies nun auf einen allgemeinen Poker-Bot erweitert werden (Texas Hold'em). Die folgenden Ziele werden gesetzt:

- Implementation des Poker-Bots
- Effiziente Speicherverwaltung
- Qualitätsmessungen der Implementation

1.2 Situationsanalyse

In dieser kurzen Situationsanalyse werfen wir einen Blick auf die aktuelle Entwicklung von Pokerbots, beleuchten den aktuellen Stand der Forschung und gehen kurz darauf ein, über welche Komponenten ein Pokerbot verfügen sollte.

Die Situationsanalyse fällt für ein Pflichtenheft sehr umfangreich aus. Wir möchten diesen Teil im Schlussbericht wiederverwenden.

1.2.1 Definition Pokerbot

Auf dem Internetportal <http://pokera1.org/wiki/> wird die Definition eines Pokerbots folgendermassen umschrieben: *"Ein Pokerbot ist ein Computerprogramm, welches Poker spielt. Pokerbot ist dabei die Abkürzung für poker robot."*

1.2.2 Einsatzgebiete und Abstraktion

Es gibt keine verlässlichen Zahlen wie viel Geld beim Online Poker weltweit umgesetzt wird, da die meisten Unternehmen nicht börsenkotiert sind. Jedoch schätzt die renommierte Zeitung Spiegel den Gesamtumsatz auf 24 Milliarden Dollar im Jahr 2010¹. Angesichts dieser grossen Zahlen verwundert es nicht, ziehen Online Casinos auch Leute mit krimineller Energie an. Die Verlockung ist gross, mit einem Pokerbot richtiges Geld zu verdienen, während man beispielsweise am Schlafen ist. Obwohl bei den meisten Online Casinos verboten, tummeln sich diverse Verkäufer von Pokerbots im Internet herum.

Andere Anreize hat die Entwicklung von Pokerbot im akademischen Umfeld. Seit dem denkwürdigen Sieg von Deep Blue gegen den damals amtierenden Schachweltmeister Gary Kasparow im Jahre 1997, suchte man sich neue Anwendungsgebiete um andere Ansätze der künstlichen Intelligenz zu testen. Poker stellt dabei mit seiner grossen Anzahl vom Zuständen und dem hohen Mass an unvollständiger Information hohe Anforderungen an einen Pokerbot. Aus diesem Grund beschäftigen sich auch viele akademische Institute mit der Entwicklung von Pokerbots.

¹ siehe <http://www.spiegel.de/netzwelt/web/0,1518,552406,00.html> (09.05.2008)

Wir werden uns im Rahmen dieser Bachelor Thesis ausschliesslich auf die Entwicklung eines Pokerbots im Rahmen einer akademischen Herausforderung beschränken.

1.2.3 Stand der Forschung

Aufgrund diverser veröffentlichter Papers zum Thema Computer Poker lassen sich zwei verschiedene Verfahren herauskristallisieren:

- der Einsatz eines regelbasierten Systems (Expertensystem)
- die Abstraktion des Spiels zum Annähern an ein pseudo-optimales Nash-Gleichgewicht (ϵ -Nash Gleichgewicht)

2 Universitäten buhlen dabei um die Vorherrschaft bei der Entwicklung des stärksten Pokerbots:

- University of Alberta
- Carnegie Mellon University

Die Universität von Alberta hat 2 Pokerbots entwickelt, welche in verschiedenen Anwendungsbereichen Erfolg haben:

- Poki (full-ring limit Texas Hold'em) und seine Nachfolger
- PsOpti (heads-up limit Texas Hold'em) und seine Nachfolger

Beide Pokerbots versuchen ein Nash-Gleichgewicht anzunähern.

Die Carnegie Mellon University aus Pittsburgh hat einen Pokerbot entwickelt, welcher in folgenden Anwendungsgebieten Erfolg hat:

- Bluffbot (heads-up limit Texas Hold'em und heads-up no-limit Texas Hold'em)

Der BluffBot ist als Expertensystem realisiert.

1.3 Weltklasse-Pokerbots

Werfen wir einmal einen Blick auf die erfolgreichsten Pokerbots der Computer Poker Competition. Weiter versuchen wir die Frage noch zu erläutern, was einen Pokerbot von Weltklasse ausmacht.

1.3.1 Poki

Poki spielt sogenannte Full-ring Games (z. B. mit 10 Spielern). Dort erreicht er gemäss den Angaben in Terence Conrad Schauenbergs Master Thesis eine durchschnittliche Spielstärke². Die Spielstärke nimmt dabei mit steigender Anzahl Mitspieler stetig ab. Das Problem ist die zu langsame Adaption der eigenen Strategie, um sich dem Spiel der Gegner anpassen zu können. Im Heads-up, also wenn nur noch 2 Spieler übrig sind, spielt Poki nur durchschnittlich.

1.3.2 PsOpti

Dieser Bot spielt nur 2 Spieler Heads-up Texas Hold'em limit und erreicht dort ein sehr gutes Niveau³. In allen Experimenten konnte er mit Weltklassenspielern mithalten. Es nähert ein Nash-Gleichgewicht an. Durch verschiedene Abstraktionen wie dem gleichbehandeln ähnlicher Karten (Bucketing) kann ein solches pseudo-optimales Nash-Gleichgewicht angenähert werden.

PsOpti ist ein dabei schon ein sehr starkes Programm, welches aber trotzdem Schwächen aufweist⁴:

- Es wird eine pseudo-optimale Spieltheoretische Lösung verwendet, dessen Schwächen permanent sind und ausgenutzt werden können.

² vgl. Terence Conrad Schauenberg, *Opponent Modeling and Search in Poker*, Seite 6

³ vgl. Terence Conrad Schauenberg, *Opponent Modeling and Search in Poker*, Seite 7

⁴ vgl. Terence Conrad Schauenberg, *Opponent Modeling and Search in Poker*, Seite 7

1.3.3 BluffBot

Der BluffBot ist ein nicht-adaptives Expertensystem, welches mit den Prinzipien der Spieltheorie entwickelt wurde⁵. Der Bot wurde zusätzlich online zur Verfügung gestellt, um die Strategie durch zahlreiche Spiele immer etwas zu verbessern.

1.3.4 Hindernisse zum Weltklasse Bot

Gemäss der Computer Poker Research Group (CPRG) der Universität Alberta sollte ein Pokerbot über folgende Komponenten verfügen⁶:

- Analyse der Handstärke
- Analyse des Handpotentials
- Biet-Strategie
- Bluffen
- Unvorhersehbarkeiten
- Opponent Modelling

Gerade beim Opponent Modelling steckt die Entwicklung noch in den Kinderschuhen. Das Opponent Modelling stellt einen eminent wichtigen Teil des Pokerspiels dar. Nur wer sich auf seine Gegner einstellen kann, kann auch wirklich viel Geld gewinnen.

Die besten Pokerbots spielen beim Heads-up limit Texas Hold'em auf Augenhöhe mit den besten Pokerspielern der Welt. Dehnt man das Spiel aber auf die Variante No Limit aus, oder erhöht die Spieleranzahl auf mehr als 2, sind nur noch durchschnittliche Resultate zu erwarten.

⁵ vgl. <http://www.bluffbot.com/>

⁶ vgl. Darse Billings et al., *The challenge of poker*, Seite 8f

1.4 Komponenten eines Pokerbots

Hier gehen wir kurz auf die oben beschriebenen Komponenten ein, und versuchen eine Prognose abzugeben, wie die Implementation in unserem Bot schlussendlich aussehen könnte.

1.4.1 Analyse der Handstärke

Die Handstärke misst die Stärke der eigenen Hand in Relation zu anderen. Wir werden hier zu Beginn der Arbeit auf bekannte Bibliotheken zurückgreifen. Da unser Pokerbot modular aufgebaut werden soll, sollte es möglich sein, verschiedene Analysetools einfach austauschen zu können.

1.4.2 Analyse des Handpotentials

Das Handpotential ist eine Wahrscheinlichkeit, dass sich die Hand im Laufe des Spiels noch verbessert (zum Beispiel durch Community Cards). Je mehr Karten noch im Spiel sind, welche die eigene Hand verbessern können, desto höher ist ihr Potential. Auch hier werden wir zu Beginn auf bekannte Bibliotheken zurückgreifen.

1.4.3 Biet-Strategie

Eine Biet-Strategie kann, wie schon angedeutet, auf ganz unterschiedliche Art und Weise implementiert werden. Denkbar wären beispielsweise regelbasierte Systeme oder auch erlernte Systeme (reinforcement learning), welche versuchen ein Nash-Gleichgewicht anzunähern.

Das im Moment führende Verfahren, um ein solches Nash-Gleichgewicht anzunähern, ist dabei das Counterfactual Minimization Regret (CFR) Verfahren der University of Alberta.

Im Rahmen der Projektarbeit im Modul Projekt 2, haben wir das Lernverfahren anhand eines einfachen Spiels (Kuhn Poker) implementiert und konnten ein Nash-Gleichgewicht annähern⁷. Beim ungleich grösseren Poker, müssen wir aber Abstraktionen vornehmen, um die verschiedenen Zustände berechnen zu können. Dies wird vor allem durch das Zusammenlegen von ähnlichen Spielzuständen in so genannte

Buckets erreicht.

1.4.4 Bluffen

Ein Bluff ist eine Bet mit einer schwachen Hand um dem Gegner eine starke Hand zu vermitteln. Als Resultat kann man dabei die Hand unmittelbar gewinnen, oder aber den Gegner "verwirren", indem man das nächste Mal mit einer starken Hand bettet (da der Gegner denkt man blufft erneut).

Bluffen ist spieltheoretisch die Abweichung von der optimalen Strategie um den Gegner zu verwirren.

Im Paper von Finnegan Southey et. al. wird das Bluffen als Teil des Opponent Modellings angesehen. Mittels eines bayes'schen Wahrscheinlichkeitsmodells wird versucht, die Unsicherheit der Spieldynamik von der Unsicherheit der gegnerischen Strategie zu trennen⁸.

Unser Pokerbot wird höchstwahrscheinlich kein Modell für Bluffs besitzen. Hier fehlt uns die Erfahrung und die Zeit für eine Umsetzung.

1.4.5 Opponent Modelling

Opponent Modelling heisst so viel wie "*das Verhalten des Gegenspielers zu modellieren*". Dem Opponent Modelling kommt im Poker eine grosse Bedeutung zu. Nehmen wir als Beispiel das einfache Spiel "Schere, Stein, Papier" welches spieltheoretisch gelöst ist. Die beste Strategie ist, in einem Drittel der Fälle Schere, Stein und Papier zu spielen. Mit dieser Taktik verliert man über eine lange Zeit nicht. Spielt man aber gegen einen Gegner, welcher nur immer Stein spielt, gewinnt man immer nur in einem Drittel der Fälle. Mit einer dem Gegner angepassten Strategie, könnte man viel mehr als nur jedes dritte Spiel gewinnen. Beim Poker verhält es sich dabei nicht anders.

⁷ vgl. Steven Heller et al., *Schlussbericht Projekt2*

⁸ vgl. Finnegan Southey et. al., *Bayes' Bluff: Opponent Modelling in Poker*, Abstract

Folgendes Zitat bringt es auf den Punkt:

Dabei geht es darum, die Spielstile der Kontrahenten zu durchschauen, indem Eigenschaften wie Spielstärke oder Aggressivität beobachtet und gespeichert werden (vgl. Billings u.a. 1998). Die so gewonnenen Erkenntnisse über die Gegner können dann in die eigene Strategie integriert werden. (Frank Bandau, 2006)⁹

1.5 Problemstellung

Im Zentrum dieser Bachelor Thesis soll erneut das Lernverfahren Counterfactual Minimization Regret stehen. Die im Modul Projekt2 gewonnenen Erkenntnisse müssen nun auf Texas Hold'em ausgedehnt werden. Dies führt zu einigen komplexen Problemstellungen:

- anstelle von nur 3 Karten beim Kuhn Poker haben wir jetzt 52
- anstelle von 1-2 haben wir nun 1-4 Bietrunden
- Zufallsknoten treten nun auch tiefer im Spielbaum auf
- anstelle von 3 Buckets haben wir mind. 8 Buckets
- der Spielbaum wird viel grösser
- die Berechnung des Nash-Gleichgewichts wird wesentlich mehr Zeit in Anspruch nehmen

⁹ vgl. Frank Bandau, *Eine kritische Betrachtung des spieltheoretischen Strategiebegriffs*, Seite 10

1.5.1 Weitere Problemstellungen

Das Spiel Texas Hold'em Poker ist um einiges komplexer als Kuhn Poker. Dies könnte noch zu weiteren Problemstellungen führen:

- durch die vielen Zustände zu Beginn des Spiels müssen unter Umständen ein eigenes Pre-Flop und ein Post-Flop Modell entwickelt werden
- es muss eine Testumgebung bereitgestellt werden, um gegen andere Bots spielen zu können, sowie eine Güte für die Qualität des Bots zu erhalten. Da die Zeit für einen Aufbau eines eigenen Testing Frameworks fehlt, muss an dieser Stelle noch eine Evaluation vorgenommen werden
- der Pokerbot muss nach durchgeführter Berechnung persistiert werden können. Die Persistierung konnte beim Kuhn Poker nicht mehr implementiert werden

Weitere Problemstellungen können auch später noch auftreten, welche zu Beginn des Projekts noch nicht abgegrenzt werden können.

1.6 Motivation

Die Motivation ist natürlich gross, zu schauen ob das Lernverfahren auch beim Texas Hold'em zu einem Nash-Gleichgewicht konvergiert. Durch die Implementation von Kuhn Poker haben wir schon einiges an Vorarbeiten geleistet, auf denen wir nun aufbauen können. Der Wettbewerb für Pokerbots¹⁰ stellt für uns ebenfalls eine grosse Motivation dar.

1.7 Vorausgegangene Arbeiten

Im Rahmen der Projektarbeit¹¹ im Modul Projekt 2 haben wir das Lernverfahren Counterfactual Minimization Regret beleuchtet und implementiert. Auf den Ergebnissen dieser Arbeit können wir aufbauen und das Lernverfahren für Poker ausdehnen.

Die im letzten Jahr erstellte Arbeit von Raphael Joggi¹² beleuchtete das Erstellen eines Pokerbots bei der Variante Texas Holdem No-Limit. Diese Variante ist aufgrund der vielen Zustände akademisch weniger gut beleuchtet. Die Arbeit von Herr Joggi

¹⁰ vgl. Kapitel 2.1

¹¹ vgl. Steven Heller et al., *Schlussbericht Projekt2*

¹² vgl. Raphael Joggi (2009), *artificial intelligent*, (Berner Fachhochschule)

beleuchtete einige theoretische Konzepte, blieb jedoch eine Implementation schuldig. An seinen gewonnenen Erkenntnissen können wir uns aber dennoch orientieren.

1.8 Limit oder No-Limit Realisation

Ein ganz wichtiger Punkt dieser Arbeit ist die Pokervariante mit der unser Pokerbot realisiert werden soll. Wir schwankten lange Zeit zwischen der Variante No-Limit und Limit hin- und her. Beide Varianten haben ihre individuellen Vor- und Nachteile:

No-Limit:

- + faszinierender da die gängige Variante (Limit Hold'em wird praktisch nicht mehr gespielt)
- + Vorarbeit von Raphael Joggi¹³
- wir haben diese Variante nicht mit Kuhn Poker beleuchtet
- der Spielbaum wird enorm viel komplexer, da unendlich viele Bietrunden möglich sind

Limit:

- + akademisch gut untersucht (einige Papers erhältlich)
- + die Kuhn Poker Lösung von uns entspricht in etwa dem Verfahren (Spielbaumgenerierung müsste nicht umgeschrieben werden)
- + bessere Qualitätsmessung möglich, da mehr Bots publiziert wurden
- die für uns persönlich nicht so faszinierende Variante

Schlussendlich kamen wir zur Meinung, dass wir das Verfahren No-Limit in Kombination mit dem CFR Verfahren zu wenig gut beleuchtet haben. Wir befürchten noch unbekannte Side-Effects, welche ein Konvergieren des Verfahrens verunmöglichen. Wir waren dabei der Meinung, dass wir die Variante No-Limit zuerst wieder bei Kuhn Poker testen müssten, um dann später eine No-Limit Lösung realisieren zu können. Deshalb haben wir uns für die Variante Limit Hold'em entschieden. Wir werden die beiden Varianten und unsere Entscheidung für Limit Hold'em im Rahmen des Schlussberichtes noch genauer gegenüberstellen.

¹³ vgl. Raphael Joggi, artificial intelligent

2 Zielbestimmung

Aus der Aufgabenstellung sowie der Situationsanalyse leiten wir folgende Ziele ab, welche in Muss- und Kann-Ziele gegliedert werden.

2.1 Übersicht

Der folgenden Tabelle sind in übersichtlicher Form unsere zu erreichenden Muss- und Kannziele zu entnehmen:

Ziel	Muss / Kann
Implementation einer lauffähigen Umsetzung eines Pokerbots	Muss
Effiziente Implementierung	Muss
Qualitätsmessung mit anderen Bots	Muss
Modularisierte Implementation	Muss
Diverse Abgabekriterien Bachelor Thesis	Muss
Vergleich verschiedener Datenstrukturen	Kann
Teilnahme bei der Annual Poker Competition	Kann
Erstellung einer Interaktiven Spielumgebung	Kann

Alle Ziele sind in den folgenden Abschnitten noch genauer definiert.

2.2 Muss Kriterien

2.2.1 Implementation einer lauffähigen Umsetzung eines Pokerbots

Um eine Vergleichsmöglichkeit mit anderen Bots zu haben und nicht gleichzeitig noch ein eigenes Testing Framework implementieren zu müssen, wollen wir unseren Bot an die technischen Anforderungen der jährlich stattfindenden Computer Poker Competition anlehnen¹⁴. Unser Bot sollte also die API der Computer Poker Competition implementieren. Diese API entspricht der API des Testing Frameworks der Poker Academy Pro 2.5¹⁵, einer Software, welche von der University of Alberta erstellt wurde. Wird diese API implementiert, kann der Bot gegen andere Bots antreten.

Die folgenden Themen sind dabei für unsere Arbeit relevant:

- Bucketing
- DFA Texas Hold'em Limit
- automatische Spielbaumgenerierung aus DFA
- Lernverfahren konvergiert gegen Nash-Gleichgewicht
- Persistierung des Pokerbots

Die einzelnen Komponenten des Pokerbots (siehe Kapitel 1.3.4 Hindernisse zum Weltklasse Bot) sollten dabei modular implementiert werden, so dass einzelne Komponenten ausgetauscht und deren Einfluss verglichen werden kann. Da einzelne Komponenten aus Zeitgründen wahrscheinlich als selbstständige Arbeiten realisiert werden müssten, lassen wir mit diesem Ziel auch Platz für spätere Arbeiten zu diesem Thema.

¹⁴ siehe www.computerpokercompetition.com

¹⁵ siehe www.poker-academy.com

2.2.2 Effiziente Implementierung

Unter einer effizienten Implementierung verstehen wir das Einhalten der Effizienzrichtlinien der Computer Poker Competition. Auf Grund der Tatsache, dass der nächste Wettbewerb noch nicht offiziell publiziert ist, verwenden wir in einem ersten Schritt die letztjährigen Angaben¹⁶. Dabei muss ein Spiel von 1000 Händen innerhalb von 10 Minuten absolviert werden:

- *The total amount of time a player can take in one match is seven seconds times the number of hands. If the maximum time is exceeded, then the program automatically chooses to fold (even if it has the option to check). As an example, in a 1000 hand match, one could take 1 second/hand on the first 999 hands, and take 6007 seconds on the last hand.*
- *NEW: The total time a player can take for a single action is 10 minutes.*

Das Ziel von uns ist somit alle 1000 Hände spielen zu können in diesen 10 Minuten.

Um diese Anforderung zu erfüllen muss das implementierte Programm auf einer Hardware mit den folgenden Spezifikationen lauffähig sein:

The competition will be run on 64-bit Kubuntu 9.10. Each process will be allowed 30 GB of uncompressed disk space. The machines are Intel Core 2 Duo E8400 CPUs running at 3.0 GHz with 4 GB of RAM. Because we are giving each competitor 4GB of RAM, this means that no other agents will be run on the same machine, and therefore the second processor is available for use. The competition will use 44 of these machines which have now all been configured. If you have specific software requests, contact the chair.

(http://www.computerpokercompetition.org/index.php?option=com_content&view=article&id=103:technical-2010&catid=40:technical&Itemid=64)

¹⁶ siehe www.computerpokercompetition.com

Unser Pokerbot darf also nicht mehr als:

- 30 GB Speicherplatz
- 4 GB RAM

benötigen.

2.2.3 Qualitätsmessung

Als Qualitätsmessung kann die Software Poker Academy Pro 2.5 hinzugezogen werden, welche über ausgeklügelte Statistiken verfügt. Interessant ist sicher der Vergleich unserer Implementation mit folgenden Pokerbots:

- PsOpti (Weltweit führender Pokerbot) oder deren Nachfolger
- Always-Call-Bot (dieser Bot callt jede Bet)
- Always-Raise-Bot (dieser Bot erhöht jede Bet)

Zusätzlich ist sicherlich auch ein Randombot (Aktionen werden zufällig ausgewählt) beizuziehen. Zum Vergleichen werden wir Spiele zwischen unserem Bot und dem entsprechenden Bot ausführen und dabei die Gewinnchancen untersuchen.

Abgabe-Kriterien Modul Bachelor Thesis

Das Modul Bachelor Thesis setzt selber einige Musskriterien voraus.

2.2.4 Erstellen eines Schlussberichts

Am Ende des Semesters muss ein Schlussbericht abgegeben werden. Dieser Schlussbericht stellt die Projektdokumentation dar. Eine elektronische Version muss z. Hd. des Fachbereichs abgegeben werden. Informationen dazu sollten noch folgen.

2.2.5 Erstellen einer Präsentation am Ende des Semesters

Erstellen einer Kurzpräsentation im Rahmen des Finaltages.

2.2.6 Erstellen einer Projektbeschreibung 1 A4 Seite

Gemäss Dr. Bernhard Anrig muss zusätzlich eine A4 Seite, welche die Bachelor Thesis kurz zusammenfasst, erstellt werden. Details werden gemäss Herr Dr. Anrig noch per Mail während des Semesters folgen.

2.2.7 Führen eines Laborjournals

Zur besseren Dokumentation der einzelnen Projektphasen, führen wir ein Laborjournal mit allen ausgeführten Arbeiten aufgeteilt auf alle 3 Studenten. Das Laborjournal ist wöchentlich zu führen und soll einfach aufgebaut sein. Pro Student reicht eine Zeile pro Woche aus.

Die Laborjournale, sowie alle Beschlüsse innerhalb der Gruppe werden auf der Kollaborationsplattform origo¹⁷ veröffentlicht.

¹⁷ siehe <http://origo.ethz.ch>

2.3 Kann Kriterien

2.3.1 Teilnahme bei der 2011 Annual Computer Poker Competition

Die Teilnahmebedingungen für die jährlich stattfindende Computer Poker Competition sind noch nicht auf der Webseite aufgeschaltet. Geht man von ähnlichen Teilnahmebedingungen wie im letzten Jahr aus, findet der Wettbewerb im Rahmen der AAAI Conference on Artificial Intelligence¹⁸ statt. Die nächste Konferenz findet in San Francisco am 7–11 August 2011 statt. Der Abgabetermin ist in der Regel ein paar Monate vor der Konferenz, wird aber nach dem Abgabetermin dieser Arbeit sein.

2.3.2 Erstellung einer interaktiven Spielumgebung

Es wäre natürlich schön, wenn man gegen den Pokerbot spielen könnte. Beim Kuhn Poker haben wir gesehen, dass die Zeit dafür sehr knapp war, um noch eine interaktive Spielumgebung (damit gegen den Computer gespielt werden kann) zu implementieren.

2.3.3 Verschiedene Datenstrukturen für den gelernten Spielbaum

Ein Faktor für einen effizienten Bot wird die persistente Speicherung des gelernten Spielbaumes sein. Falls noch Zeit vorhanden ist, wäre es sicherlich interessant eine Implementation mit unterschiedlichen Verfahren zu realisieren. Die unterschiedlichen Implementationen wären mittels eines qualitativen Vergleichs gegenüberzustellen und zu bewerten.

2.4 Mögliche Stärke unseres Bots

Wie schon bei der Projektarbeit müssen wir uns in kurzer Zeit sehr viel Stoff aneignen:

- spieltheoretische Komponenten von Poker
- Poker Strategien
- Nash-Gleichgewichte
- Spielbäume, extensive Darstellung
- Portierung des Lernverfahrens von Kuhn Poker zu Texas Hold'em Poker

Schlussendlich fehlt es uns auch an der Erfahrung auf diesem Gebiet. Es ist also nicht realistisch, einen Pokerbot auf Weltklasse-Niveau zu entwickeln.

¹⁸ siehe <http://www.aaai.org/home.html>

3 Planung und Organisation

3.1 Organisationsstruktur

Name	Kontaktdaten	Rolle
Steven Heller	hells1@bfh.ch	Projektmitarbeiter
Bruno Wegmüller	wegmb1@bfh.ch	Projektleiter
Christian Zweili	zweic1@bfh.ch	Projektmitarbeiter
Dr. Jürgen Eckerle	juergen.eckerle@bfh.ch	Projektbetreuer
Dr. Bernhard Anrig	bernhard.anrig@bfh.ch	Projektbetreuer

3.1.1 Verteilung der Rollen

Alle Studenten nehmen die gleiche Stellung innerhalb des Projektes ein. Der Projektleiter dient als Schnittstelle zwischen dem Projektteam und den Betreuern um die Kommunikation zu vereinfachen.

3.2 Projektplanung

An den 16 Wochen stehen uns ca. 360 Stunden pro Person zur Verfügung (1 ECTS Punkt stellt ca. 30 Stunden dar). Als 3er Gruppe käme dies somit auf ca. 1080 Stunden. Diese Grobplanung geht von etwas über 1000 Stunden aus, was pro Student und Woche eine Auslastung von ca. 20 Stunden bedeutet.

Task_ID Beschreibung	Monat	September			Oktober				November				Dezember			Januar			
		38	39	40	41	42	43	44	45	46	47	48	49	50	51	52	1	2	3
0 Kick-Off	Aufwand in h																		
1 Pflichtenheft erstellen	18																		
1.1 IST-Analyse	4																		
1.2 Sammeln der benötigten Informationen	2																		
1.3 Inhalte definieren	2																		
1.4 Würdigung und Zieldefinition	1																		
1.5 Festlegen / Formulieren der Ziele (Muss / Kann)	1																		
1.6 Pflichtenheft fertigstellen	8																		
2 Bereitstellen Testumgebung	34																		
2.1 Software Poker Academy Pro organisieren	2																		
2.2 API definieren	4																		
2.3 Server einrichten	4																		
2.4 Entwicklungsumgebung bereitstellen	8																		
2.5 Always-Call-Bot und Always-Raise-Bot implementieren	4																		
2.6 Maven Integration erstellen	4																		
2.7 Dokumentation Testumgebung schreiben	8																		
3 Implementation Pokerbot	272																		
3.1 Automaten Definition Hold'em Poker	24																		
3.2 Evaluation und Integration Handanalyse Tool	8																		
3.3 Evaluation und Integration Handpotential Tool	8																		
3.4 Bucketing	64																		
3.5 Karten vom Test-Framework erhalten	4																		
3.6 erste Tests mit dem Framework	4																		
3.7 Persistierung des eigenen Bots	64																		
3.8 Lernverfahren für Poker berechnen	32																		
3.9 Dokumentation Implementation	64																		
4 Projektbericht verfassen	124																		
4.1 Dokumentation CFR Algorithmus noch verbessern	30																		
4.2 Erstellung Projektbericht	64																		
4.3 Korrekturen	30																		
Total Stunden Projekt	448																		

3.3 Meilensteine

1. Pflichtenheft erstellen

2. Testumgebung bereitstellen

- Implementation einfacher Bots wie Always-Call und Always-Raise
- Karten vom Framework beziehen
- Auswertungen erstellen

3. Implementation Pokerbot

- Bucketing
- DFA Texas Hold'em erstellen
- automatische Spielbaumgenerierung aus DFA
- Lernverfahren konvergiert gegen Nash-Gleichgewicht
- Persistierung des Pokerbots

3.3.1 Laborjournale

Die Laborjournale werden auf der Origo (<http://minimizationregret.origo.ethz.ch>) Plattform jede Woche aktualisiert. Sie geben Auskunft über gemeinsame Beschlüsse, sowie aufgewendete Zeit.



Die Laborjournale sind unter Doc / Mainpage (Einstiegsseite) / Bachelorthesis / Laborjournale zu finden.

4 Produkteinsatz

4.1 Anwendungsbereiche

Der Anwendungsbereich des Produktes Pokerbot beschränkt sich aufs automatische Pokerspiel mit der von der University of Alberta definierten API. Der Einsatz als Pokerbot an der Computer Poker Competition wäre ebenfalls möglich.

4.2 Zielgruppe

Das Produkt dient ausschliesslich für die in die Bachelor Thesis involvierten Anspruchsgruppen. Namentlich sind dies die Betreuer, der Fachbereich, der Experte und die Studenten.

4.3 Betriebsbedingungen

Der Pokerbot sollte unter den unter Punkt 2.2.2 (Effiziente Implementierung) definierten Hardwarebedingungen laufen.

4.4 Lizenz

Wir sind damit einverstanden, dass die BFH/TI alle Güter inklusive der Immaterialgüter der durch die Studenten geleisteten Arbeiten grundsätzlich nutzen darf.

5 Produktumgebung

5.1 Hardware

Der Pokerbot sollte unter den unter Punkt 2.2.2 (Effiziente Implementierung) definierten Hardwarebedingungen laufen.

5.2 Software

Der Pokerbot wird mit Java entwickelt. Da wir die Vorarbeit ebenfalls mit Java realisiert haben, ist es die logische Konsequenz, den Pokerbot ebenfalls mit Java zu entwickeln. So können wir auf dem bereits bestehenden Programmgerüst aufbauen.

6 Entwicklungsumgebung

6.1 Testing und Development

Das Produkt wird mit Eclipse entwickelt und verwendet die Toolchain von Maven als Konfigurationsmanagement. Mit Maven lässt sich das Projekt kompilieren, die Projektdokumentation erstellen sowie automatische Unit Tests durchführen.

Mit Origo nutzen wir zusätzlich eine Kollaborationsplattform, welche es uns die für die Entwicklung der Software wichtigen Tools bereitstellt (Subversion, Backup, etc).

Jeder Student und ebenfalls die Betreuer verfügen über einen eigenes Login. Dieser Login wurde im Rahmen des Moduls Projekt2 schon erstellt.

6.2 Dokumentation

Als zentrale Dokumentationsplattform nutzen wir ebenfalls die Kollaborationsplattform Origo. Hier werden alle im Schlussbericht referenzierten Papers bereitgestellt, sowie die Laborjournale abgelegt. Der aktuelle Stand der Projektdokumentation (Schlussbericht) lässt sich auch laufend überblicken.

Aus den Erfahrungen der Projektarbeit werden wir die Dokumentation dabei nicht mehr direkt als Wiki bereitstellen, sondern schreiben diese schon direkt in eine Worddatei.

7 Literatur und Quellen

7.1 Literaturverzeichnis

Frank Bandau (2006), Eine kritische Betrachtung des spieltheoretischen Strategiebegriffs, Seminararbeit, (HBK-Braunschweig, Institut für Medienforschung, Abteilung Medienwissenschaften)

http://www.strategiespielen.de/wordpress/wp-content/uploads/bandau_eine_kritische_betrachtung_des_spieltheoretischen_strategiebegriffs.pdf

Darse Billings et. al. (2001): *The challenge of poker*, (University of Alberta)

<http://webdocs.cs.ualberta.ca/~darse/Papers/AIJ02.pdf>

Steven Heller, Bruno Wegmüller, Christian Zweili, (2010): *Schlussbericht Projekt2*, (Berner Fachhochschule)

Raphael Joggi (2009), *artificial intelligent*, (Berner Fachhochschule)

.Nick Abou Risk, Duane Szafron (2010), Using Counterfactual Regret Minimization to Create Competitive Multiplayer Poker Agents, (University of Alberta)

<http://webdocs.cs.ualberta.ca/~duane/publications/pdf/2010aamas.pdf>

Terence Conrad Schauenberg (2006), *Opponent Modeling and Search in Poker*, (University of Alberta)

<http://webdocs.cs.ualberta.ca/~darse/Papers/schauenberg.msc.pdf>

Finnegan Southey et. al. (2005), *Bayes' Bluff: Opponent Modelling in Poker*, (University of Alberta)

<http://webdocs.cs.ualberta.ca/~bowling/papers/05uai.pdf>

8 Glossar

Bet	der Einsatz
CFR	Abkürzung für Counterfactual minimization regret. Ein Lernverfahren für Spiele mit unvollständiger Information.
Community Cards	Gemeinschaftskarten; d.h. die Karten, welche in der Mitte zu liegen kommen und für alle Teilnehmer gelten. Auch bekannt als Flop, Turn und River.
Full Ring	Spiel mit vielen Spielern (ca. 10)
Limit Hold'em	Poker-Form mit einem fixen Limit-Betrag, bis zu dem die Spieler erhöhen können und einer maximalen Anzahl von Erhöhungen. Auch Split Limit genannt.
No-Limit Hold'em	Poker-Form ohne fixe Beträge und ohne maximale Anzahl Erhöhungen.
Side-Effect	Wird in der Informatik als unbeabsichtigte Wirkung beschrieben.
Toolchain	Als Toolchain (englisch für Werkzeugkette) wird in der Softwareentwicklung eine systematische Sammlung von Werkzeug-Programmen bezeichnet, welche zur Erzeugung eines Produktes Verwendung findet.