

# ARES-ABALONE

**Philipp Moritz**

pcmoritz@googlemail.com

**Ulrich Müller**

ulimllr@googlemail.com

**Jean-Nicolas Lang**

jlang@physik.uni-wuerzburg.de

**Bijan Chokoufe Néjad**

bijan.chokoufenejad@physik.uni-wuerzburg.de

## Abstract

Abstract

**Keywords:** Keywords

## 1 Introduction

Introduction

## 2 Abalone

Abalone ist ein Brettspiel für zwei Spieler. Jeder Spieler besitzt dazu in der Grundstellung 14 Spielsteine auf einem hexagonalen Brett. Abwechselnd sind Züge mit einem, zwei oder drei linear zusammenhängenden Steinen in die selbe Richtung erlaubt. In der Regel kann ein Zug von gegnerischen Steinen blockiert werden. Eine Möglichkeit, gegnerische Steine zu verschieben, stellen die sogenannten Sumitos dar. Dabei werden zwei oder drei Kugeln in einer Reihe bewegt und können eine Minderheit gegnerischer Kugeln wegschieben (also 3 gegen 1, 3 gegen 2, oder 2 gegen 1). Ziel des Spiel ist es 6 Spielsteine des Gegners vom Brett zu schieben.

Abalone wird von der Denksportorganisation gefördert und ist Bestandteil der Denk-Sport-Plympiade.

## 3 Heuristik

Heuristiken werden in der Informatik eingesetzt, um den Rechenaufwand zu für die Lösung eines Problems gering zu halten. Mit dem Einsatz einer Heuristik wird auf die Garantie einer optimalen Lösung verzichtet, die z.B. beim systematischen überprüfen alles Spielabläufe entstehen würde. Dafür wird die Laufzeit des Programmes durch verschiedene Faustregeln und Schätzungen bewertet und der Zug ausgeführt, der nach wenigen Zügen zur besten Bewertung führt. Für die Bewertung einer Stellung, kamen folgende Bewertungskriterien zu Einsatz:

- Wie weit sind die Steine in der Mitte des Brettes?
- Wie sehr befinden sich die eigenen Steine auf einem Haufen?
- Können gegnerische Steine geschlagen werden?
- Was für günstige/ungünstige Muster kommen an der Grenze der zwei Spieler vor?

Mit diesen Kriterien wurde jeder Stellung eine "Energie" zugeordnet, die wie folgt berechnet wurde:

$$\begin{aligned}
 H_{ges} &= H_{eigen} + H_{surface} + H_{interface} + V(r) \\
 H_{eigen} &= \lambda_1 \sum_{(i,j)} \text{Spielerfarbe}(i,j) \\
 H_{surface} &= \lambda_2 \sum_{(i,j)} \sum_{(p,q)} \langle (i,j) | pq \rangle \delta((p,q) \text{ ist leer}) \\
 H_{interface} &= \lambda_3 \sum_{\text{Muster}} \sum_{\text{Richtungen}} \delta(\text{wenn Muster gefunden}) \\
 V(r) &= \lambda_4 \sum_{(i,j)} V(r(i,j))
 \end{aligned}$$

Jede hypothetische Stellung, die nach einem möglichen Zug entstehen wird bewertet und die Züge bevorzugt, die Hamiltonfunktion maximieren.

## 4 Alpha-Beta-Suche

Alpha-Beta-Suche

## 5 GIT

Git (engl. *Blödmann*) dient zur Versionsverwaltung von Dateien. Es zeichnet sich aus durch höchste Geschwindigkeit, simples Design (Konsole!?) und besonders intuitive und einfache Steuerung, die es uns ermöglicht hat, in kürzester Zeit (circa 12 Stunden) ein nahezu funktionsfähiges Netzwerk aufzusetzen. Zu allererst muss ein "Repository" auf der Seite <https://github.com/> erstellt werden. Freundlicherweise hat sich Herr Moritz um diese Aufgabe gekümmert. Anschließend können Daten geladen ("pull"-Funktion) und hochgeladen ("push"-Funktion). Üblicherweise sieht der Uploadvorgang wie folgt aus

```

$user git add .
$user git commit -m "Kommentar"
$user git push

```

Aufgrund der Tatsache, dass alle Beteiligten die Files nach belieben ändern konnten, kam es häufig zu folgender Warnung

```

$CONFLICT (content): Merge conflict in abalone.cc.orig
Automatic merge failed; fix conflicts and then commit the result.

```

In diesem Fall muss "gemerged" werden.

```
$user git mergetool -t meld
```

Oftmals ist das nicht ausreichend und wir empfehlen die Befehle

```
git add . ; git commit -m "Kommentar"; git push;  
git mergetool -t meld; git push
```

in möglichst zufälliger Reihenfolge auszuführen.

## 6 Grillen

Grillen