

Vier-Gewinnt

2. Mai 2019

Inhaltsverzeichnis

| | | |
|----------|--------------------------------|----------|
| 1 | Einleitung | 1 |
| 2 | Vier Gewinnt | 1 |
| 2.1 | Spielidee | 1 |
| 2.2 | Implementation | 1 |
| 3 | Implementation der KI | 1 |
| 3.1 | Taktik der KI | 1 |
| 3.2 | Schwächen | 3 |
| 3.3 | Effizienzbetrachtung | 3 |
| 4 | Künstliche Intelligenz | 3 |
| 4.1 | Aktueller Stand | 3 |
| 4.2 | Ausblick | 3 |

1 Einleitung

2 Vier Gewinnt

2.1 Spielidee

Die Grundidee des „Vier Gewinnt“ Spieles ist es vier der eigenen Spielsteine bzw. Symbole in eine Reihe zu bringen. Dabei ist es egal ob dies horizontal, vertikal oder diagonal erreicht wird. Es spielen immer zwei Spieler gegeneinander, die abwechselnd einen Spielstein von oben in das Spielfeld einfügen, der dann so lange herunterfällt, bis er auf das Ende des Spielfeldes oder einen anderen Spielstein stößt. Sobald zum ersten Mal vier Spielsteine eines Spielers eine Reihe bilden, endet das Spiel und der betreffende Spieler gewinnt. Das Spielfeld in der Grundversion ist ein 7x6 Feld. Grundsätzlich sind aber auch größere und kleinere Felder möglich. ¹

2.2 Implementation

Zunächst habe ich, wie in der Softwareentwicklung üblich die Klassendiagramme angelegt,

Klassendiagramme

um diese dann nach Umwandlung in ein Implementationsdiagramm dann umzusetzen.

Implementationsdiagramme

Nach der Umsetzung versuchte ich die Laufzeit etwas zu senken, indem ich eine eigene Klasse erstellte an deren Objekten die Simulationen durchgeführt wird, um es mehr abzutrennen und um die redundante Speicherung der bisherigen Züge zu vermeiden.

Klassendiagramme Vier Game und Logik

3 Implementation der KI

3.1 Taktik der KI

Die Taktik der KI (Künstliche Intelligenz) ist natürlich zu gewinnen oder eine (unmittelbare) Niederlage zu verhindern. Dazu wird ein Minimax-Algorithmus genutzt, um die beste Entscheidung zu treffen. Bei diesem Verfahren durchläuft der Algorithmus einen Suchbaum, der bei der maximalen Suchtiefe je-

¹Lehmann, Jörg „Vier gewinnt“ <http://www.brettspiele-report.de/vier-gewinnt/> Stand: 01.05.2019

| height | | | | | | | | | | | |
|--------|---|---|---|---|---|---|----|----|-------|--|--|
| 11 | = | | | | | | | | | | |
| 10 | = | = | | | | | | | | | |
| 9 | = | = | + | | | | | | | | |
| 8 | = | = | - | + | - | | | | | | |
| 7 | = | = | + | = | + | | | | | | |
| 6 | = | = | - | + | - | - | | | | | |
| 5 | = | = | = | = | + | + | + | | | | |
| 4 | = | = | - | = | - | - | - | - | | | |
| | 4 | 5 | 6 | 7 | 8 | 9 | 10 | 11 | width | | |

Abbildung 1:
2

dem Ergebnis einen Wert zuweist oder, wenn schon vorher ein Sieg oder eine Niederlage auftritt, diesen jeweils einen positiven bzw. negativen Wert zuordnet. Bei der Auswertung wird dann immer das Minimum der gegnerischen Züge gewertet und das Maximum der eigenen, so dass jeder Spieler ein „perfektes“ Spiel spielt. Des Weiteren bevorzugt der Algorithmus einen schnellen Sieg bzw. eine spätere Niederlage.

Neben dem Minimax-Algorithmus könnte man auch vorberechnete Eröffnungszüge nutzen, aber diese unterscheiden sich für jede Spielfeldgröße. Einige Spiele gelten dadurch auch schon als gelöst:

Zu Abbildung 1:
+ der 1. Spieler gewinnt
- der 2. Spieler gewinnt
= Unentschieden

3.2 Schwächen

Theoretisch sind der KI nur Grenzen gesetzt durch die Rechenleistung bzw. die Tiefe der Simulation

3.3 Effizienzbetrachtung

Um mehr Züge zu simulieren ist erheblich mehr Rechenleistung erforderlich. Die Komplexität steigt nahezu exponentiell, aber je weiter das Spiel fortgeschritten ist und je weniger Möglichkeiten es gibt, desto schneller wird der Algorithmus. Ich habe Simulationen durchgeführt mit verschiedenen Feldgrößen und habe den ersten Zug von der KI berechnen lassen, dabei habe ich die Zeit gemessen und damit passende Tiefen gewählt, um die Zeit bei weniger als 10 Sekunden zu halten. Hätte ich die vordefinierten Pfade genommen, wäre deutlich an Rechenzeit gespart worden, jedoch bräuchte man zum Erstellen dieser sehr lange bzw. würde es den Rahmen dieser Facharbeit sprengen. Auch wäre es möglich anstatt der VierLogik Klasse auch einfach nur kopierte Arrays zu verwenden, aber dabei ist es schwer darauf zu achten, dass keines dieser Arrays verändert wird.

Aber nun zur Effizienz bei verschiedenen Feldgrößen. Da die Höhe nicht so ausschlaggebend ist wird hier nur betrachtet wie sich die Ausführungszeit verändert, wenn die Suchtiefe und die Feldbreite variiert werden. Die KI versucht in jedem Fall den ersten Zug zu berechnen. *Diagramm 5-7-9w 1-10d*

4 Künstliche Intelligenz

Definition

4.1 Aktueller Stand

tensorflow, machine learning

4.2 Ausblick

Filme usw. (Wargames, Terminator, 2001, I Robot)