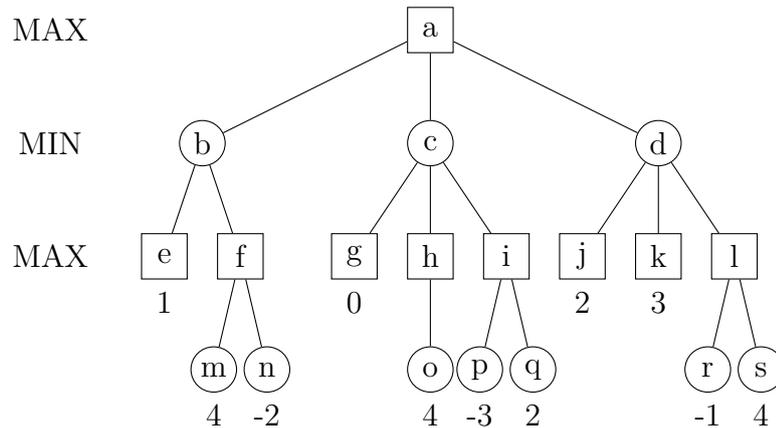


## 2. Übungsblatt

(zum 20.04.2011)

### Aufgabe 5      Minimax-Prinzip und $\alpha$ - $\beta$ -Stutzen

Gegeben sei der folgende Spielbaum:



- Bestimmen Sie anhand des Minimax-Prinzips die optimale Spielsequenz, sowie die Bewertung des erreichten Endzustandes.
- Welche Knoten müssen durch  $\alpha$ - $\beta$ -Stutzen nicht besucht werden?
- Nehmen Sie an, dass zusätzliches Wissen über das Spiel bekannt ist. Der maximale Nutzenwert sei 4, d.h. es ist dem MAX-Spieler unmöglich eine größere Ausgabe zu erzielen. Wie kann dieses Wissen genutzt werden, um  $\alpha$ - $\beta$ -Stutzen weiter zu verbessern?

### Aufgabe 6      Verbesserungen beim Erstellen des Spielbaums

Um die Größe des Spielbaums zu reduzieren gibt es viele Verbesserungen des  $\alpha$ - $\beta$ -Stutzens [13]. Informieren Sie sich über mindestens 3 der folgenden Punkte. Erläutern Sie Ihre gewählten Verbesserungen an einem Beispiel. Inwieweit ist eine Kombination dieser Verbesserungen möglich?

- Iterative Tiefensuche [9], Ruhesuche, Negamax, Negascout [11]
- Transpositionstabellen [3], Zobrist-Hashing [14]
- Widerlegungstabellen [1]
- Minimale Fenstersuche [10, 9, 11]
- Aspirationssuche [2]
- Killer-Heuristik [1], Geschichtsheuristik [12], Schmetterlingsheuristik [6]
- inkrementelle Zugberechnung [8]

### **Aufgabe 7      Brettbewertung bei Mühle-Spiel**

Will man den Minimax- bzw.  $\alpha$ - $\beta$ -Algorithmus für das Mühle-Spiel verwenden, muss jede Brettstellung  $s \in \mathcal{G}$  anhand ausgewählter Kriterien  $f_i : \mathcal{G} \rightarrow \mathbb{R}$ ,  $1 \leq i \leq n$  bewertet werden. Ein zu maximierendes/minimierendes Kriterium könnte beispielsweise die Anzahl der eigenen/gegnerischen offenen Mühlen sein. Für gewöhnlich wird dann die Fitness/Güte von  $s$  anhand einer gewichteten linearen Funktion  $f(s) = \sum_{i=1}^n w_i f_i(s)$  bestimmt. Um somit nicht für jedes  $s$  alle möglichen Brettstellungen berechnen bzw. evaluieren zu müssen, gilt es

1. eine hinreichend gute Menge an Kriterien  $f_i$  zu finden, und
2. deren Gewichte  $w_i$  so zu wählen, dass  $f$  (fast) jedes Brett ausreichend gut bewertet.

Die Aufgabe eines evolutionären Algorithmus könnte somit die Bestimmung der  $w_i$  sein.

- a) Informieren Sie sich über das Brettspiel Mühle. Welche Kriterien sind Ihrer Meinung nach entscheidend, um ein Brett  $s$  hinreichend gut zu bewerten?
- b) Geben Sie eine Kodierungsvorschrift für ein Mühlebrett an.
- c) Geben Sie für die Berechnung jedes Ihrer Kriterien jeweils einen Pseudocode an.
- d) Was können Sie a priori über die Wahl der jeweiligen Gewichte der Kriterien sagen?

### **Aufgabe 8      Alternative Lösungen des Mühle-Spiels**

Sicherlich ist die Verwendung des Minimax-Algorithmus nur eine Möglichkeit, evolutionäre Algorithmen für das Mühle-Spiel zu benutzen [4, 5, 7].

- a) Welche anderen Möglichkeiten zur Lösung des Mühle-Spiels mithilfe von evolutionären Algorithmen sind möglich?  
Hinweis: Informieren Sie sich über evolutionäre Methoden des „Fallbasierten Schließen“, der „Mehrkriterienoptimierung“, oder des „Maschinelles Lernens“. Vielleicht hilft Ihnen auch das Schlüsselwort „Genetische Programmierung“.
- b) Machen Sie konkrete Vorschläge zur Umsetzung Ihrer alternativen Lösungsmethode(n).
- c) Diskutieren Sie Vor- und Nachteile Ihrer Lösungsmethode(n).

## Literaturverzeichnis

- [1] S. G. Akl and M. M. Newborn. The principal continuation and the killer heuristic. In *ACM Annual Conference Proceedings*, pages 466–473, New York, NY, USA, 1977. ACM Press.
- [2] G. M. Baudet. *The Design and Analysis of Algorithms for Asynchronous Multiprocessors*. PhD thesis, Department of Computer Science, Carnegie Mellon University, Pittsburgh, PA, USA, 1978.
- [3] D. M. Breuker, H. J. van den Herik, J. W. H. M. Uiterwijk, and L. V. Allis. Replacement schemes and Two-Level tables. *ICCA Journal*, 19(3):175–180, 1996.
- [4] O. David-Tabibi, H. J. van den Herik, M. Koppel, and N. S. Netanyahu. Simulating human grandmasters: evolution and coevolution of evaluation functions. *Proceedings of the 11th Annual conference on Genetic and evolutionary computation*, page 1483–1490, 2009.
- [5] N. Franken and A. P. Engelbrecht. Evolving intelligent game-playing agents. *Proceedings of the 2003 annual research conference of the South African institute of computer scientists and information technologists on Enablement through technology*, page 102–110, 2003.
- [6] D. Hartmann. Butterfly boards. *ICCA Journal*, 11(2-3):64–71, 1988.
- [7] G. Kendall and G. Whitwell. An evolutionary approach for the tuning of a chess evaluation function using population dynamics. In *Evolutionary Computation, 2001. Proceedings of the 2001 Congress on*, volume 2, pages 995–1002, 2001.
- [8] L. Kocsis. *Learning Search Decisions*. PhD thesis, Universiteit Maastricht, Maastricht, The Netherlands, 2003.
- [9] T. A. Marsland. Relative efficiency of alpha-beta implementations. *Proceedings of the Eighth international joint conference on Artificial intelligence - Volume 2*, page 763–766, 1983.
- [10] J. Pearl. *Heuristics: Intelligent Search Strategies for Computer Problem Solving*. Addison Wesley Longman Publishing Co., Inc., Boston, MA, USA, 1984.
- [11] A. Reinefeld. An improvement of the scout tree search algorithm. *ICCA Journal*, 4(6):4–14, Dec. 1983.
- [12] J. Schaeffer. The history heuristic and Alpha-Beta search enhancements in practice. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 11:1203–1212, Nov. 1989.
- [13] J. Schaeffer and A. Plaat. New advances in Alpha-Beta searching. *Proceedings of the 1996 ACM 24th annual conference on Computer science*, page 124–130, 1996.
- [14] A. L. Zobrist. A new hashing method with application for game playing. Technical Report 88, Computer Science Department, The University of Wisconsin, Madison, WI, USA, Apr. 1970. Reprinted in *ICCA Journal*, 13(2):69–73, 1990.