AN ANALYSIS OF MINIMAX SEARCH AND ENDGAME DATABASES IN EVOLVING AWALE GAME PLAYER

Ngwira Seleman[1] ,Randle O.A[2] and Zuva T[3]

[1,3] Department of Computer Systems Engineering, Tshwane University of Technology, Soshanguve, South Africa
tzuva@hotmail.com

[2]Department of Computer Science, Tshwane University of Technology, Soshanguve, South Africa
rotran86@yahoo.com

ABSTRACT

This paper studies the respective performance of minimax search and endgame databases in competing against the Awale shareware. It also investigates the performance of combining both techniques to evolve a hybrid player against the Awale shareware.

KEYWORDS

minimax search, endgame, databases, Awale

1. INTRODUCTION

Evaluation is very crucial and tedious task in Game playing techniques. There are many techniques used evolving game players, algorithms and systems in literature so in order to proclaim the best among many, choose one to use and improve there is need to evaluate them. One way to evaluate them is by comparing the results of the techniques used which is the aim of this paper. Other ways of evaluating include measuring their effectiveness; another is by evaluating the technique used.

Game playing is an interesting aspect of computer systems and has become very important after the computer was able to solve the game of chess [1,2 ,3]. The computer also defeated the world best chess player Kasparov, the human world champion in a tournament competition in 1997[3].The success of the computer in playing chess resulted in increased interest in games by researchers in both mathematics and artificial intelligence. Also games are now the most interesting artificial intelligence application to the public [4] and one of those games is Awale also known as Ayo or Awele.

Awale is a count-and-capture, two –person-zero-sum board game , which comprises 12 pits on two rows called as usual, North and South, with 4 seeds in each pit at the beginning of a game. The following rules are commonly applied. An agent selects all seeds from a non-empty pit on his row and sows them counter-clockwise into each pit excluding the starting pit. If the last seed is sown into a pit on the opponent's row, leaving that pit with 2 or 3 seeds, the agent captures the

seeds in the pit and seeds in preceding pits on the opponent's row that contain 2 or 3 seeds(this is called the 2-3 capture rule).An agent cannot capture all the seeds on the opponent's row, so he is obliged to make a move that will give his opponent a move and this is called the golden rule. A controversial rule of Awale, yet to be resolved, is when an agent cannot move in such a way that he gives his opponent a legal move, then either the game is cancelled or the agent that caused this stalemate loses the game no matter his score. The game ends(1)when an agent has captured more than 24 seeds, or (2) when both agents have captured 24 seeds leading to a draw or (3) when fewer seeds circulate endlessly on the board. Case (3) has the following specialisation: if there are fewer seeds on the board that neither agent can ever capture, but both agents will always have a legal move, the game ends and each agent is awarded the seeds on his row.

The objective of this paper is to show a survey of various machine learning techniques which have been implemented using Minimax search, endgame databases or a combination of both in evolving Awale game players.

The rest of the paper is succinctly described as follows:
- Section 2 overviews game tree and minimax search
- Section 3 describes endgame databases
- Section 4 presents the hybrid combination
- Section 5 is the conclusion

2. GAME TREE AND MINIMAX SEARCH

A game tree algorithm computes the root successor with the highest pay-off for the current player or the minimax value of a game tree from which the best move can easily be inferred. A large number of game playing algorithms use game tree to represent game positions and moves. Nodes of the tree are game positions and the root node corresponds to the current game position. Branches of a node represent legal moves from the position represented by the node and a leaf node has no successor.

Generally, the value of a leaf is estimated by the evaluator and represents the number in proportion to the chance of winning the game. Specifically, constructing such an evaluator for some games like Ayo is a difficult task. The leaves in a game tree are game positions for which an integer value evaluator f exists giving the pay-off in that position Generally, the value of a leaf is estimated by the evaluator and represents the number in proportion to the chance of winning the game. The evaluator can be extended to the minimax function, which determines the value for each player in a node and is formally given in (1) as follows [5,6]:

$$f(n) = \begin{cases} eval(n), & \text{if } n \text{ is a leaf node} \\ \max\{f(c) \mid c \text{ is a child node of } n\}, & \text{if } n \text{ is a max node} \\ \min\{f(c) \mid c \text{ is a child node of } n\}, & \text{if } n \text{ is a min node} \end{cases}$$

(1)

The function eval(n) scores the resulting board position at each leaf node n. The standard method of scoring is in terms of a linear polynomial [7]. It has been shown that every game tree algorithm constructs a superposition of a max (T^+) and a min(T^-) solution tree. The equivalent evaluator is the following Stockman equality [8]:

$$f(n) = \begin{cases} \max\{g(T^-) \mid T^- \text{ is a min } tree \, rooted \, in \, n\} \\ \min\{g(T^+) \mid T^+ \text{ is a max } tree \, rooted \, in \, n\} \end{cases}$$

(2)

Where the function g is defined by [9]:

$$g(T^+) = \max\{f(c) \mid c \text{ is a terminal in } T^+\}$$
$$g(T^-) = \min\{f(c) \mid c \text{ is a terminal in } T^-\}$$

(3)

Conventionally, the basic idea of minimax algorithm is synonymously related to the following optimization procedure. Max player tries as much as possible to increase the minimum value of the game, while Min tends to decrease its maximum value at node n as both players play towards optimality. The entire process can be formally described by the following extended Stockman formula (4) below:

$$f(n) = \begin{cases} \max\{f(c) \mid c \text{ is a child node of } n\} - f(n), if \, n \text{ is a min } node \\ \min\{f(c) \mid c \text{ is a child node of } n\} + f(n), if \, n \text{ is a max } node \end{cases}$$

(4)

Minimax search has been highly explored with success for playing games such as through massively parallelized, hardware orientated alpha-beta search. Minimax search Algorithm constructs a game tree in ascending order from the root and then employs backward induction to predict the game value as it descends the tree from the leaves.The principal Variation (PV) is the optimum path that suggests best moves for players of Two Person Zero Sum (TPZS) games such as Awale.

The problem with minimax search is that if the evaluator is applied at the leaves and backward induction is performed to compute the value of a game tree, there is no guarantee that a correct move.Also the issue of how to design a suitable evaluator and how to select a correct move without the rationality assumption [4]

Studies have shown that improving the evaluation function does not still improve the results such as the six features considered for the design of an evaluation function [9] , also another agent was created using minimax search which was evolved using a genetic algorithm with the objective of showing that a better representation can lead to a deeper search [10].Six additional features were added to those used in [9] to improve performance of an Ayo agent. The result obtained at the strongest level (grandmaster level) of play is shown below in Table 1. The results in table 1 also show the average moves employed in the games, and the standard derivations(STD) for seed captured and moves(A move can be counted when both players have moved seeds alternately).

Table 1 Results of playing Awale at the Grandmaster level using Minimax Search

Results from Davis and Kendall(2002) at depth 7		
Moves(Standard Derivation)		Seeds captured(Standard Deviation)
	Evolved Agent	Awale Shareware
80.00(5.48)	4.40(0.55)	26.80(1.64)
Results from Daoud et al.,(2004) at depth 5		
51.10(0.19)	6.40(0.27)	26.50(0.10)

3. ENDGAME DATABASE

Game playing have also used endgame databases to evolve Awale game players.[11] has been offered for evolving Awale game player. Others include Lithindion[12] which is quite different from others in its class because it is a combination of alpha-beta search and endgame databases. Another endgame database player (Marvin) was evolved using a method called drop-out expansion [13] which is a combination of depth-first and breadth-first search. Another program called Softwari[14]constructs large endgame databases. There is only one commonality, they all focus on searching and database utilization and gave little attention to evaluation functions [4]. Generally, endgame databases can only be computed when few pieces remain on board and they can require solution lengths that defy the capabilities of minimax based searches for optimal play. Endgame databases have also been used to evolved case based reasoning(CBR)[15] and aggregate malanobis distance function(AMDF)[16] which respectively used endgame database containing good Tcchucallion strategies while the other used 2 clusters of strategies which respectively contained good and bad tchucallion to evolve their players. The result of using end game can sometimes not give the best result. ADMF played against Awale but could not successfully defeat Awale at the amateur stage.

Table 2 Results of Playing Awale using Endgame Databases

	No of moves(Average)	Evolved Player	Awale
Initiation	23	26	5
Beginner	33	26	7.6
Amateur	40	18.6	27.5
Grand master	61.6	13.4	25

4. HYBRID COMBINATION

A very important approach of evolving an Awale player is the hybrid technique or the combination method, which involves combining 2 techniques to evolve an Awale player. Minimax has been combined with other techniques to evolve players such as with Refinement assisted Minimax (RAM) [4], Case based reasoning (CBR) [15],Aggregate Malanobis Distance Function(AMDF)[16]. These combinations have shown that move selection can be assisted by employing them to guide the selection process. Also the combinations have shown that in some cases such as CBR it is inherently adequate for handling irrationality that frequently occurs in the game of Awale/Ayo[15].These combinations have produced players that produced very attractive results by defeating Awale at the highest level(grand Master level). The combination of Minimax and AMDF [16] played against Awale shareware developed by [17] and defeated it at all stages except at the grand master stage where the evolved player competed vigorously against awale but lost the game. The player evolved by [15] which combined Minimax and CBR defeated Awale at all the stages including the grandmaster stage. The result of [15] is shownbelow in table 3.The results show the results of playing Awale at the grandmaster level.

Table 3 Results of playing Awale using a combination of Minimax Search and Endgame Databases

Minimax(STD)	Awale(STD)	Moves(STD)	Overrides(STD)
16.00(5.27)	26.50(0.53)	68.00(45.33)	Not applicable
Minimax	Minimax-CBR	Moves	Overrides
7.00(3.16)	28.00(3.16)	38.50(11.92)	10.10(2.23)
Minimax-CBR	Awale	Moves	Overrides
25.50(0.53)	15.00(1.05)	42.70(2.31)	24.00(2.11)

5. CONCLUSIONS

This study has looked at all three stages and strongly suggests that the combination of endgame database with minimax search technique which always produces interesting result. Rather than separating the techniques. The results obtained by combining them are appealing and exciting.

REFERENCES

[1]   Thompson, K., Computer Strength, in Advances in Computer Chess 3,M.Clarke (ed.), Pergamon Press, Oxford, pp. 55-56,(1982)
[2]   Thomson,K.:6-piece Endgames, ICCA Journal, vol 19, no. 4, pp 215-226, (1996)
[3]   Hamilton, S. And Garber,L., Deep Blue's hard ware-software synergy, IEEE Computer, 30, 10, pp. 29-35, (1997)
[4]   Olugbara., An Investigation of Minimax Search technique for Evolving Ayo/Awari Player.Proceedings of IEEE-ICICT 4th International Conference on Information and Communication Technology, Cairo, Egypt, (2006).
[5]   Bruin, A.D., Pijls, W. and Plaat, A. Solution Trees as a Basic for Game Tree Search, ICCA Journal, 17, 4, pp. 207-219, (1994)
[6]   Bruin, A.D. and Pijls, W., Trends in Game Tree Search. SOFSEM, pp. 255-274, (1996)
[7]   Samuel, A.L., Some Studies in Machine Learning Using the Game of Checkers, IBM J. ofRes. And Dev. 3, 210-229, (1959)
[8]   Pijls, W. and Bruin, A.D., Game Tree Algorithms and Solution Trees, theor. Compt. Sci., 252 (1-20: pp. 197-215, (2001)
[9]   Davis, J.E. and Kendall, G., An Investigation, using co-evolution, to evolve an Awari Player. In proceedings of Congress on Evolutionary Computation (CEC 2002), pp. 1408-1413, (2002)
[10] Daoud, M., Kharma, N., Haidar, A. and Popoola, J., Ayo, the Awari Player, or How Better Representation Trumps Deeper Search, Proceedings of the 2004 IEEE Congress on Evolutionary Computation, pp. 1001-1006, (2004)
[11] Lincke, T.R and Marzetta, A., Large endgame databases with Limited Memory Space. ICGA Journal, 23, 3, pp. 131-138, (2000)
[12] Allis, V ., Muellen, M. V .D. and Herik, J. V. D, Proof-number Search, Artificial Intelligence, vol. 66, pp. 91-124, (1994)
[13] Lincke, T.R., Strategies for the Automatic Construction of opening Books: Computers and Games, pp. 74-86, (2000)
[14] The University of Alberta Computer Awari Group, http://www.cs.ualberta.ca/awari
[15] Olugbara, O.O., Adigun, M.O., Ojo, S .O. and Adewoye, T.O., An efficient heuristic for evolving an agent in the strategy game of Ayo, ICGA Journal ,pp.92-96 (2006)
[16] Randle, O.A., Olugbara, O.O and Lall, M. Investigating the Performance of Minimax Search and Aggregate Mahalanobis Distance Function in evolving an Ayo/Awale Player. World Academy of Science and Technology (WASET). (2012)
[17] Myraid software, http://www.myraid-online.com/awale.htm