



Report for AlphaDog (cs440 #3 project)

CS 440 P 3
Hongyu Zhou
Huiyi Chen
April 27 2017

Problem Definition

In this assignment we are going to design a AI player for the Atropos Game, which will including a static evaluator and an adversarial search algorithm from the ones you have studied in class(alpha-beta search with Minimax).
The code will be clearly commented and excutable. You can play with it or let it play against other players.

Method and Implementation

1. function availmove(board, lastplay): Find all the available moves and put it in a 2d array

This function will return the 2-d int array with number 1 for circles that are available for the moves with the same index as (height,leftdistance), and with number 0 for circles that are not available for move. However, if the last Play is null(the script goes first), then it will return an empty array.
2. function alphaBeta(board, lastplay, depth, alpha, beta, ismax): Find the best move using minmax & alpha-beta pruning algorithm.

This is the core function which will evaluate the move that available and score it by using alpha-beta search with Minimax and then find the best move.

Experiments

We will test our programe by running a tournament with 10 game.
There will be a tournament for the script vs random player and a tournament for ourselves vs our script.
The game will be place at a board size of 7.

We will evaluate the accuracy of the script player according the win and lose games.
Also, we will evaluate the speed of the script player by recording the time.
Finally, we will play with the script player and test its intelligence.

Results

The result for the tournaments is:

script vs random player(default)

script (8 win , 2 lose)

random player(2 win, 8 lose)

So, the accuracy is 80% for the script

The totally moves for 10 game is 146, which mean, each game take approximately 15 moves

The average time for a game is 6 second which mean the script can make a move within 1/3 second

Human(ourselves) vs script(alphaDog)

Script(7 win, 3 lose) Human(3 win, 7 lose) So, the accuracy is 70% for the script vs human.

We try to make a quick decision when we are playing with the script to make the game fair, but we still take much more time than computer and make some stupid mistakes.

Base on the tests our script is very intelligent.

```
dhcp-wifi-8021x-168-122-201-50:p3 hongyuzhou$ java AtroposGame 7 "python alp
This will be a fantastic battle between Script and Default!
```

```
      [1  3]
    [3  0  2]
  [1  0  0  3]
[3  0  0  0  2]
[1  0  0  0  0  3]
[3  0  0  0  0  0  2]
[1  0  0  0  0  0  0  3]
[3  0  0  0  0  0  0  0  2]
[1  2  1  2  1  2  1  2]
Last Play: null
```

Default will go first.

The game had the following moves:

```
[Circle colored 2 at: (1, 5, 3).
, Circle colored 2 at: (1, 4, 4).
, Circle colored 3 at: (2, 4, 3).
, Circle colored 3 at: (2, 3, 4).
, Circle colored 1 at: (3, 2, 4).
, Circle colored 3 at: (3, 1, 5).
, Circle colored 1 at: (2, 1, 6).
, Circle colored 1 at: (1, 1, 7).
, Circle colored 2 at: (1, 2, 6).
, Circle colored 1 at: (2, 2, 5).
, Circle colored 1 at: (1, 3, 5).
]
```

Script has won in 11 moves!

Final board:

```
      [1  3]
    [3  0  2]
  [1  0  0  3]
[3  0  0  0  2]
[1  0  0  0  0  3]
[3  3  1  0  0  0  2]
[1  1  1  3  3  0  0  3]
[3  1  2  1  2  2  0  0  2]
[1  2  1  2  1  2  1  2]
Last Play: (1, 1, 3, 5)
```

```
dhcp-wifi-8021x-168-122-201-50:p3 hongyuzhou$
```

] Script has won in 15 moves!
Final board:

```
      [1  3]
    [3  3  2]
  [1  3  2  3]
 [3  0  2  0  2]
[1  0  0  3  0  3]
 [3  3  2  3  1  0  2]
[1  3  2  3  3  0  0  3]
[3  1  1  0  0  0  0  0  2]
 [1  2  1  2  1  2  1  2]
Last Play: (1, 1, 2, 6)
```

] Script has won in 28 moves!
Final board:

```
      [1  3]
    [3  3  2]
  [1  3  3  3]
 [3  3  2  2  2]
[1  3  3  2  2  3]
 [3  3  2  3  3  3  2]
[1  1  3  3  1  1  3  3]
[3  1  1  1  1  2  1  1  2]
 [1  2  1  2  1  2  1  2]
Last Play: (1, 1, 7, 1)
```

] Script has won in 5 moves!
Final board:

```
      [1  3]
    [3  0  2]
  [1  0  0  3]
 [3  0  0  0  2]
[1  0  0  0  0  3]
 [3  3  0  0  0  0  2]
[1  3  2  0  0  0  0  3]
[3  1  1  0  0  0  0  0  2]
 [1  2  1  2  1  2  1  2]
Last Play: (1, 1, 2, 6)
```

] Default has won in 8 moves!
Final board:

```
      [1  3]
    [3  0  2]
  [1  0  0  3]
 [3  0  0  0  2]
[1  0  0  0  0  3]
 [3  1  3  0  0  0  2]
[1  0  3  3  2  0  0  3]
[3  0  3  2  2  0  0  0  2]
 [1  2  1  2  1  2  1  2]
Last Play: (3, 1, 2, 6)
```


Script has won in 17 moves!
Final board:

```
      [1  3]
     [3  3  2]
    [1  3  3  3]
   [3  3  3  2  2]
  [1  3  1  1  3  3]
 [3  3  1  1  2  2  2]
[1  0  3  3  0  0  0  3]
[3  0  0  0  0  0  0  0  2]
 [1  2  1  2  1  2  1  2]
Last Play: (1, 4, 3, 2)
```

Script has won in 28 moves!
Final board:

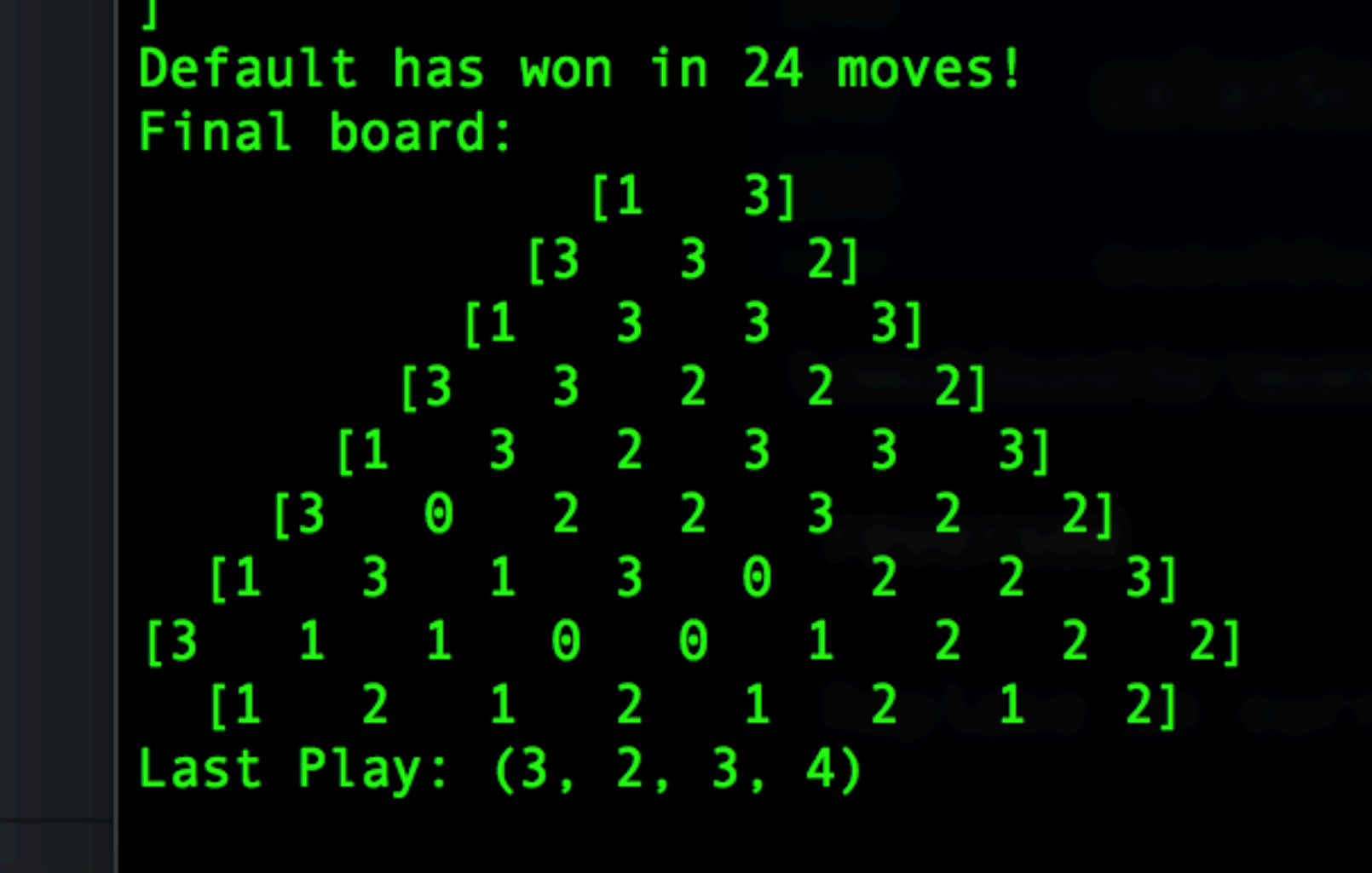
```
      [1  3]
     [3  3  2]
    [1  3  3  3]
   [3  3  3  2  2]
  [1  3  3  2  3  3]
 [3  3  3  3  3  2  2]
[1  3  1  1  3  1  3  3]
[3  1  1  2  1  1  2  2  2]
 [1  2  1  2  1  2  1  2]
Last Play: (1, 2, 5, 2)
```

Script has won in 26 moves!
Final board:

```
      [1  3]
     [3  3  2]
    [1  1  3  3]
   [3  3  3  3  2]
  [1  3  3  1  3  3]
 [3  3  3  3  3  3  2]
[1  1  2  2  2  2  3  3]
[3  1  1  0  0  1  2  2  2]
 [1  2  1  2  1  2  1  2]
Last Play: (1, 2, 1, 6)
```

Script has won in 7 moves!
Final board:

```
      [1  3]
     [3  0  2]
    [1  0  0  3]
   [3  0  0  0  2]
  [1  0  0  0  0  3]
 [3  3  0  0  0  0  2]
[1  1  3  3  0  0  0  3]
[3  1  1  2  0  0  0  0  2]
 [1  2  1  2  1  2  1  2]
Last Play: (1, 1, 2, 6)
```



Discussion

- The strength of our script is that it runs really fast, however the weakness is that the evaluator is not perfect
- We try to improve the winrate by setting a higher seach depth for the algorithm, but we find that the winrate did not goes high when the depth goes too high. We find that the win rate may decrease when depth is to high.
- The Experiments are general success, the script player have a very high accuracy. However It can not guarantee a 100% winrate is the limit.
- In the potential future we want to set more factors for the evaluation function and use machine learning to tuning the score system to get a almost 100% win rate

Conclusions

Atropos Game is a really interesting game. We learned a lot and have fun in the process of design an AI player for the game. And we find that the most import thing for an artificial intelligence is their value system, which is the value functionin our script. To let the compter know what is important and interpret things like human is a great work.

Credits and Bibliography

https://en.wikipedia.org/wiki/Alpha%E2%80%9Cbeta_pruning
<https://en.wikipedia.org/wiki/Minimax>
http://inst.eecs.berkeley.edu/~cs61b/fa14/ta-materials/apps/ab_tree_practice/