Adversarial Search

George Konidaris
gdk@cs.duke.edu

Spring 2016
“Chess is the Drosophila of Artificial Intelligence”
Kronrod, c. 1966

TuroChamp, 1948
Why Study Games?

Of interest:

• Many human activities (especially intellectual ones) can be modeled as games.
• Prestige.

Convenient:

• Perfect information.
• Concise, precise rules.
• Well defined “score”.
“Solved” Games

A game is solved if an optimal strategy is known.

Strong solved: *all positions*.
Weakly solved: *some (start) positions*.
Typical Game Setting

Games are usually:

- 2 player
- Alternating
- Zero-sum
  - Gain for one loss for another.
- Perfect information

Very much like search:

- Start state
- Successor function
- Terminal states (many)
- Objective function

*but alternating control.*
Game Trees

player 1 moves

player 2 moves

player 1 moves
Key Differences vs. Search

- You select to max score
- They select to min score
- Only get score here
Minimax Algorithm

Max player: select action to maximize return.
Min player: select action to minimize return.

This is optimal for both players (if zero sum).
Assumes perfect play, worst case.

Can run as depth first:
- Time $O(b^d)$
- Space $O(bd)$
Minimax

Decision tree with player 1 (pl) at the root and two possible moves for player 2 (p2) at each level. The values at the leaves represent the outcomes for player 1.
In Practice

Depth is too deep.
  • 10s to 100s of moves.

Breadth is too broad.
  • Chess: 35, Go: 361.

Full search never terminates for non-trivial games.

Solution: substitute evaluation function.
  • Like a heuristic - estimate value.
  • Perhaps run to fixed depth then estimate.
Search Control

- Horizon Effects
  - What if something interesting at horizon + 1?
  - How do you know?

- When to generate more nodes?
- How to selectively expand the frontier?
- How to allocate fixed move time?
Pruning

Single most useful search control method:

- Throw away whole branches.
- Use the min-max behavior.

- Cutoff search at *min* nodes where *max* can force a better outcome.
- Cutoff search at *max* nodes when *min* can force a worse outcome.

Resulting algorithm: *alpha-beta pruning.*
Alpha-Beta

Empirically, has the effect of reducing the branching factor by a square root for many problems.

Effectively doubles the search horizon.

Alpha-beta makes the difference between novice and expert computer game players. Most successful players use alpha-beta.
Deep Blue (1997)

480 Special Purpose Chips
200 million positions/sec
Search depth 6-8 moves (up to 20)
Games Today

World champion level:
• Backgammon
• Chess
• Checkers (solved)
• Othello
• Some poker types:

Perform well:
• Bridge
• Other poker types

Far off: Go
Go
Very Recently

Fan Hui
European Go Champion

0 - 5

AlphaGo (Google Deepmind)