Game Playing
Part 1 Minimax Search

Yingyu Liang
yliang@cs.wisc.edu

Computer Sciences Department
University of Wisconsin, Madison

[based on slides from A. Moore http://www.cs.cmu.edu/~awm/tutorials, C. Dyer, J. Skrentny, Jerry Zhu]
Sadly, not these games (not in this course) ...
Overview

- two-player zero-sum discrete finite deterministic game of perfect information
- Minimax search
- Alpha-beta pruning
- Large games
- two-player zero-sum discrete finite NON-deterministic game of perfect information
Two-player zero-sum discrete finite deterministic games of perfect information

Definitions:

- **Zero-sum**: one player’s gain is the other player’s loss. Does not mean *fair*.
- **Discrete**: states and decisions have discrete values
- **Finite**: finite number of states and decisions
- **Deterministic**: no coin flips, die rolls – no chance
- **Perfect information**: each player can see the complete game state. No simultaneous decisions.
Which of these are: Two-player zero-sum discrete finite deterministic games of perfect information?

[Shamelessly copied from Andrew Moore]
Which of these are: Two-player zero-sum discrete finite deterministic games of perfect information?

[Shamelessly copied from Andrew Moore]
Which of these are: Two-player zero-sum discrete finite deterministic games of perfect information?

- **Zero-sum**: one player’s gain is the other player’s loss. Does not mean *fair*.
- **Discrete**: states and decisions have discrete values
Which of these are: Two-player zero-sum discrete finite deterministic games of perfect information?

Zero-sum: one player's gain is the other player's loss. Does not mean fair.
Discrete: states and decisions have discrete values
Finite: finite number of states and decisions

[Shamelessly copied from Andrew Moore]
Which of these are: Two-player zero-sum discrete finite deterministic games of perfect information?

- **Zero-sum**: one player's gain is the other player's loss. Does not mean *fair*.
- **Discrete**: states and decisions have discrete values
- **Finite**: finite number of states and decisions
- **Deterministic**: no coin flips, die rolls – no chance

[Shamelessly copied from Andrew Moore]
Which of these are: Two-player zero-sum discrete finite deterministic games of perfect information?

Zero-sum: one player's gain is the other player's loss. Does not mean fair.

Discrete: states and decisions have discrete values

Finite: finite number of states and decisions

Deterministic: no coin flips, die rolls – no chance

Perfect information: each player can see the complete game state. No simultaneous decisions.

[Shamelessly copied from Andrew Moore]
Which of these are: Two-player zero-sum discrete finite deterministic games of perfect information?

Zero-sum: one player’s gain is the other player’s loss. Does not mean fair.

Discrete: states and decisions have discrete values

Finite: finite number of states and decisions

Deterministic: no coin flips, die rolls – no chance

Perfect information: each player can see the complete game state. No simultaneous decisions.

[Shamelessly copied from Andrew Moore]
II-Nim: Max simple game

• There are 2 piles of sticks. Each pile has 2 sticks.
• Each player takes one or more sticks from one pile.
• The player who takes the last stick loses.

(ii, ii)
**II-Nim: Max simple game**

- There are 2 piles of sticks. Each pile has 2 sticks.
- Each player takes one or more sticks from one pile.
- The player who takes the last stick loses.

(ii, ii)

- Two players: Max and Min
- If Max wins, the score is +1; otherwise -1
- Min’s score is –Max’s
- Use Max’s as the score of the game
The game tree for II-Nim

Two players: Max and Min

Max wants the largest score
Min wants the smallest score

Convention: score is w.r.t. the first player Max. Min’s score = − Max
The game tree for II-Nim

Two players:
Max and Min

Symmetry
(i ii) = (ii i)

Max wants the largest score
Min wants the smallest score
The game tree for II-Nim

Two players: Max and Min

Max wants the largest score
Min wants the smallest score
The game tree for II-Nim

Two players: Max and Min

Max wants the largest score
Min wants the smallest score
The game tree for II-Nim

Two players:
Max and Min

Max wants the largest score
Min wants the smallest score
The game tree for II-Nim

Two players: Max and Min

Max wants the largest score
Min wants the smallest score
The game tree for II-Nim

Two players:
Max and Min

Max wants the largest score
Min wants the smallest score
The game tree for II-Nim

Two players: Max and Min

Max wants the largest score
Min wants the smallest score
The game tree for II-Nim

Two players: Max and Min

Max wants the largest score
Min wants the smallest score
The game tree for II-Nim

Two players: Max and Min

Max wants the largest score
Min wants the smallest score
Game theoretic value

- Game theoretic value (a.k.a. minimax value) of a node = the score of the terminal node that will be reached if both players play optimally.
Two players: Max and Min

Max wants the largest score
Min wants the smallest score
The game tree for II-Nim

Two players:
Max and Min

Max wants the largest score
Min wants the smallest score
The game tree for II-Nim

Two players:
Max and Min

Max wants the largest score
Min wants the smallest score
The game tree for II-Nim

Two players:
Max and Min

Max wants the largest score
Min wants the smallest score
The game tree for II-Nim

Two players: Max and Min

Max wants the largest score
Min wants the smallest score
The game tree for II-Nim

Two players: Max and Min

Symmetry \((i \ ii) = (ii \ i)\)

The first player always loses, if the second player plays optimally

Major difference from standard search: The opponent has control over which action to take, when it’s his turn.

Convention: score is w.r.t. the first player Max. Min’s score = − Max

Max wants the largest score
Min wants the smallest score
**Game theoretic value**

- Game theoretic value (a.k.a. minimax value) of a node = **the score of the terminal node that will be reached if both players play optimally.**
- = The numbers we filled in.
- Computed bottom up
  - In Max’s turn, take the max of the children (Max will pick that maximizing action)
  - In Min’s turn, take the min of the children (Min will pick that minimizing action)
- Implemented as a modified version of DFS: **minimax algorithm**
Minimax algorithm

function Max-Value(s)
inputs:
  s: current state in game, Max about to play
output: best-score (for Max) available from s
  if ( s is a terminal state )
      then return ( terminal value of s )
  else
      \( \alpha := -\infty \)
      for each \( s' \) in Succ(s)
          \( \alpha := \max( \alpha, \text{Min-value}(s') ) \)
  return \( \alpha \)

function Min-Value(s)
output: best-score (for Min) available from s
  if ( s is a terminal state )
      then return ( terminal value of s )
  else
      \( \beta := \infty \)
      for each \( s' \) in Succs(s)
          \( \beta := \min( \beta, \text{Max-value}(s') ) \)
  return \( \beta \)

• Time complexity?
• Space complexity?
## Minimax algorithm

<table>
<thead>
<tr>
<th>Function</th>
<th>Inputs</th>
<th>Output</th>
</tr>
</thead>
<tbody>
<tr>
<td><code>Max-Value(s)</code></td>
<td>s: current state in game, Max about to play</td>
<td>best-score (for Max) available from s</td>
</tr>
<tr>
<td><code>Min-Value(s)</code></td>
<td>s: current state in game, Min about to play</td>
<td>best-score (for Min) available from s</td>
</tr>
</tbody>
</table>

**function `Max-Value(s)`**

- **inputs:**
  - s: current state in game, Max about to play

- **output:** best-score (for Max) available from s

  - if (s is a terminal state)
    - then return (terminal value of s)
  - else
    - \( \alpha := -\infty \)
    - for each \( s' \) in Succ(s)
      - \( \alpha := \max(\alpha, \text{Min-value}(s')) \)
    - return \( \alpha \)

**function `Min-Value(s)`**

- **output:** best-score (for Min) available from s

  - if (s is a terminal state)
    - then return (terminal value of s)
  - else
    - \( \beta := \infty \)
    - for each \( s' \) in Succ(s)
      - \( \beta := \min(\beta, \text{Max-value}(s')) \)
    - return \( \beta \)

### Time complexity?
- \( O(b^m) \leftarrow \text{bad} \)

### Space complexity?
- \( O(bm) \)
What are the game theoretic values? In particular, A’s
Against a dumber opponent?

- Max surely loses!
- If Min not optimal,
- Which way?
- Why?