By Daniel F. Dickinson
MakerPlace Volunteer

How AI beats humans at chess and other games

BOARD GAME WINNING AI

Introduction to Board AI

Which will hopefully keep you from being bored! 😿 😭

Topics

- Introduction
  - What is Artificial Intelligence (AI)
- Solving Single-Player Logic Games
  - The Zebra Problem
  - As Constraint Satisfaction Problems
- Playing Two-Player (Adversarial) Board Games
  - Tic-tac-toe
  - Heuristics
  - Chess

More Topics

- Beyond Logic—Randomness & Uncertainty
  - Contingency Problem Redux
  - Uncertainty vs. Rational Actor/Adversary
  - Crossover into gambling, probability and statistics (‘pure’ math)

What is Artificial Intelligence (AI)?

- “Finding a ‘good enough’ solution to hard problems quickly enough to be useful” — *Daniel F. Dickinson, 2021*
- The above is the actual goal of much of the Computer Science field known as Artificial Intelligence.
- A hard problem is one which cannot be solved in a period of time that is useful, even at a theoretical level.
- If you’re interested in a more concrete definition, the topic is ‘algorithmic complexity’ and/or ‘NP complete problems’.
- Non-deterministic polynomial time is often worse even than exponential, but not always.

Solving Single-Player Logic Games

The Zebra Puzzle

Not this Zebra

The Canonical Zebra Puzzle

1. There are five houses.
2. The Englishman lives in the red house.
3. The Spaniard owns the dog.
4. Coffee is drunk in the green house.
5. The Ukrainian drinks tea.
6. The green house is immediately to the right of the ivory house.
7. The Old Gold smoker owns snails.
8. Kools are smoked in the yellow house.
9. Milk is drunk in the middle house.
10. The Norwegian lives in the first house.
11. The man who smokes Chesterfields lives in the house next to the man with the fox.
12. Kools are smoked in the house next to the house where the horse is kept.
13. The Lucky Strike smoker drinks orange juice.
14. The Japanese smokes Parliaments.
15. The Norwegian lives next to the blue house.

- Now, who drinks water? Who owns the zebra?

- In the interest of clarity, it must be added that each of the five houses is painted a different color, and their inhabitants are of different national extractions, own different pets, drink different beverages and smoke different brands of American cigarets *[sic]*. One other thing: in statement 6, right means *your* right.
- Life International, December 17, 1962
- https://en.wikipedia.org/wiki/Zebra_Puzzle

Information About the Zebra Puzzle

- “Legend” attributes this to Albert Einstein or Lewis Carroll.
  - First verifiable appearance is in Life Magazine 1962.
- Turned into a generic puzzle for use in computer science.
  - Generic version (The Zebra Problem) in a term paper.
- Many ways to solve this puzzle. It is used as an AI problem because it can be formulated as a Binary Constraint Satisfaction Problem and analysis is ‘easy’.
- BCSP (simplified): A problem with a number of elements, which must meet a set of conditions for the relationships between pairs of elements. (e.g. The Zebra owner drinks Guiness and the Norwegian lives next to snail owner).
- Used as common benchmark for Constraint Satisfaction Problems

Solving the Zebra Puzzle as a CSP

- A CS Course Detailed Walkthrough of Solving via CSP
- A 4th year CS lecture on CSPs, including N-Queens, Map Colouring (CSPs also cover Timetables, and much more)
- A CIS Project on solving the Zebra (and other) problems
  - By 'yours truly', which is the only reason you'll ever hear of it…
- Artificial Intelligence CIS 4750 (uoguelph.ca) (Fall 2004)
  - General CSP presentation (lecture notes)
- CSP solvers can fully solve many human-complexity logic puzzles quickly, but don't eliminate the NP problem.
- A second year CS discussion on how to solve the Zebra Puzzle (but not solved as a Constraint Satisfaction Problem; demonstrates reducing complexity to avoid CSP/AI).

Two-Player (Adversarial) Games

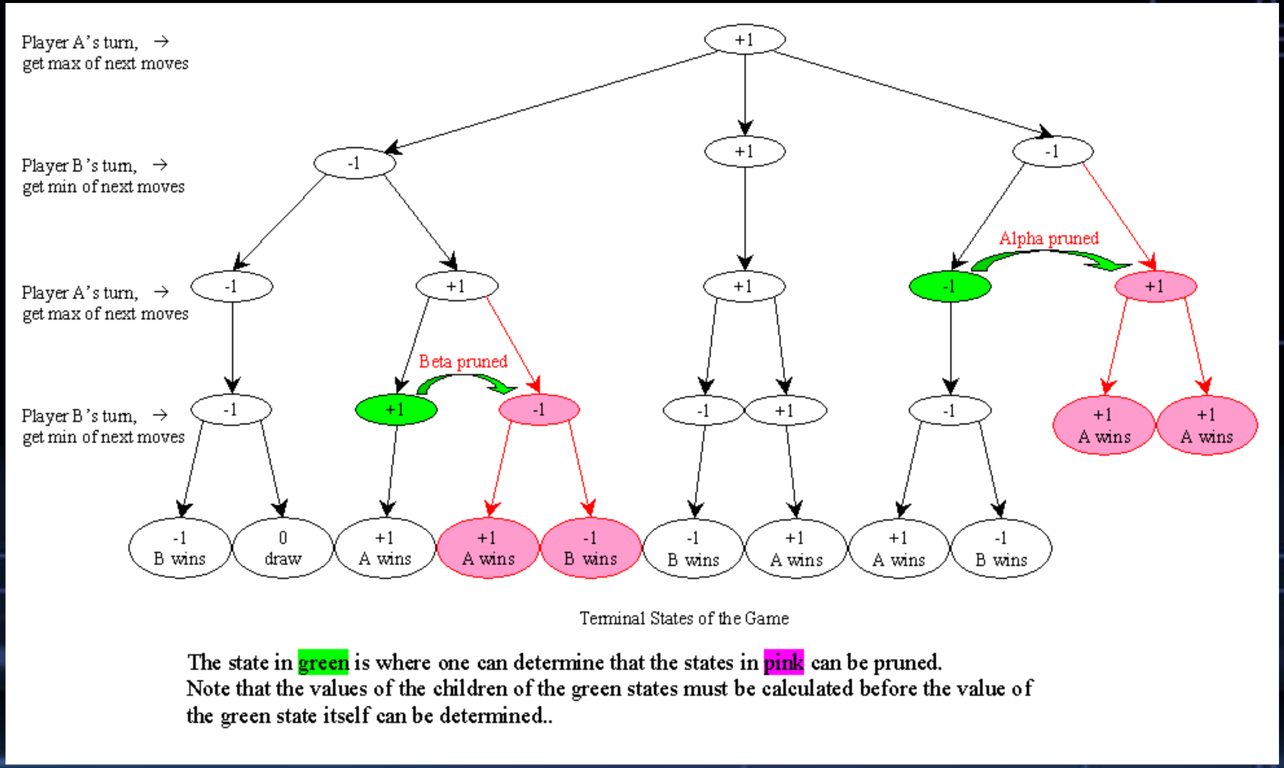
- Oware board with polished pebbles in starting position, in pencil sketch style (effects).

Analyzing Two-Player Games

Links

- [Min-Max Principle and Alpha-Beta Pruning (rice.edu)]()
- [Slide Serve Presentation on Min-Max Search and Alpha-Beta Pruning]()
- [Artificial Intelligence: A Modern Approach, 4th US ed.]()
  - Widely used (and massive) AI Textbook

α – β Pruning (Min-Max Game)



Tic-tac-toe

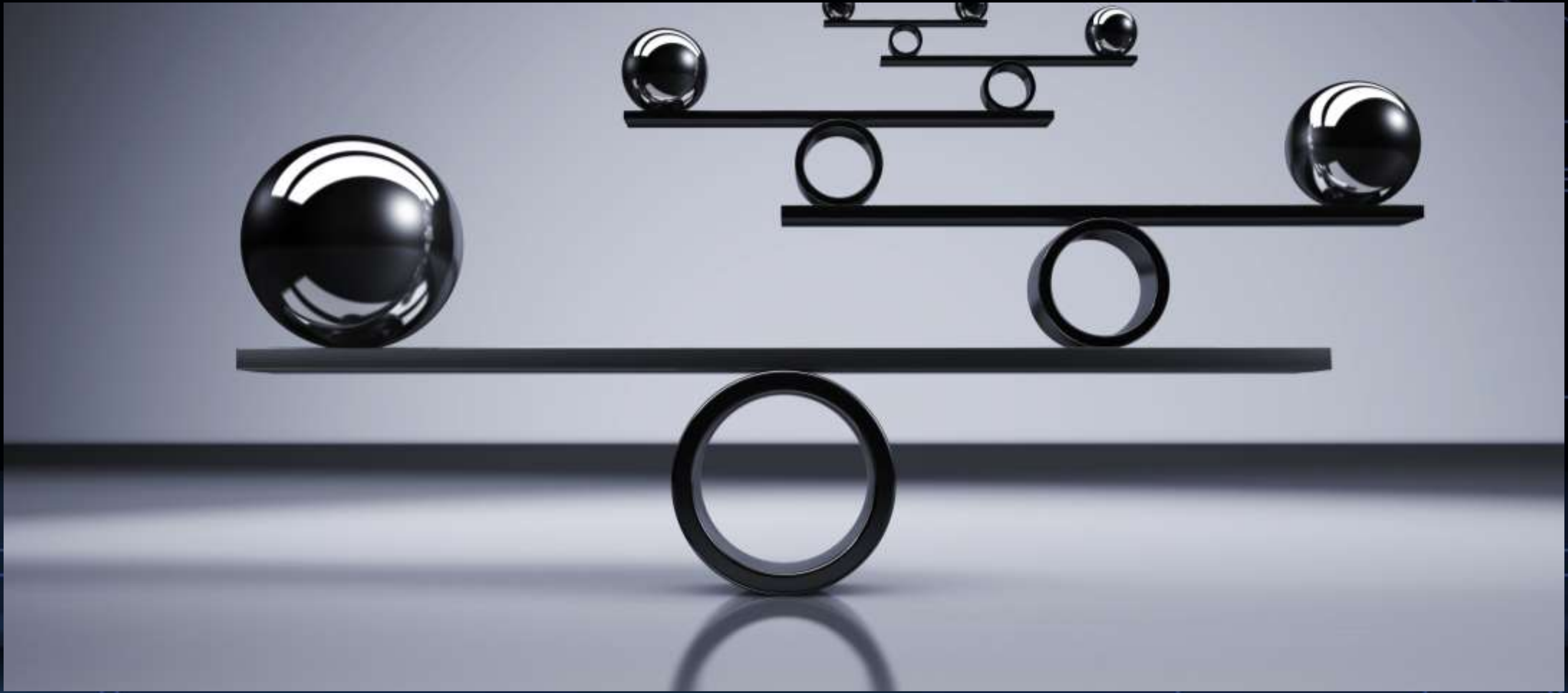
Tic-tac-toe Facts

- Tic-tac-toe on Wikipedia for the uninitiated.
- Against a competent rational actor the best result one can achieve is a draw.
- A strategy is to pick the move that leaves the ‘least advantageous move’ possible for the opponent.
- Need to recalculate best move on every turn because subsequent moves depend on the opponent’s move.
  - This is called the ‘contingency problem’.
- Small problem, so even uninformed search is enough.

Tic-tac-toe: Stratagems

- Q: What does ‘advantageous move’ mean?
  - A: It is a move that results in a win for the opponent, assuming the player makes the best possible response, as does the opponent, for all subsequent moves. A slightly less advantageous move is one that results in a draw. The least advantageous move (for the opponent) is one in which the player wins.
- How do you measure that on a per-move basis?
- This is the core of real AI: Heuristics

Heuristics



Heuristics for the Win

- An heuristic is essentially any technique for making an approximation.
- A ‘good’ heuristic is one that leads to desired outcomes, (in chess, checkmating the opponent) in the most cases. Alternatively, one which leads to the fewest number of undesired outcomes (losing in tic-tac-toe) over many games.
- Which type of measure (half-full/half-empty) to use depends on the type of game. (It’s part of the heuristic).

Creating Heuristics

- Logical Analysis (Human Identifies Evaluation Function)
  - Per-problem or class of problem
  - Tends to require more ‘high-level’ thought/analysis
- Machine Learning (System ‘Learns’ Evaluation Function)
  - Done right can identify patterns not readily obvious to humans
  - Done wrong it can mis-identify patterns
    - Garbage-In : Garbage-Out (GIGO)
  - Depends on the training set and problem ‘framework’
  - Too easy to trust the ‘black box’ (lack of insight into reasons)

Tic-tac-toe Heuristic

Using Tic-tac-toe to Practise Heuristics

- Cut off evaluation of the search tree before reaching the ‘bottom’ (as is required of more complex games like chess, due to time constraints).
- Instead of ‘knowing’ the ‘utility’ of move with certainty (that is we no longer prove a move will lead to win, lose, or draw), so we have to use an ‘evaluation function’ which *estimates* the probability of the move being a win, lose, or draw.
- The quality of this evaluation function (heuristic) is vital.

Chess: Machine Mastery

Playing Chess: The Space Rations Version

- Chess on Wikipedia & Rules of Chess on Wikipedia
- 8 x 8 square board
- Player and opponent start on opposite sites of the board
- 2 rows of ‘pieces’ per player
- There a variety of types of pieces, each of which follow different rules of movement (e.g. a Rook moves in a horizontal or vertical line, any number squares, a Bishop moves diagonally, a Queen can move in any straight line, any number of squares).
- Goal is ‘checkmate’ (opponent’s King is threatened with capture and has no legal move to avoid capture).

Chess: An AI Perspective

- Similar to tic-tac-toe, there is win, lose, or draw (checkmate for player, opponent, or a stalemate).
- For chess, the branch factor (~35) makes the number of moves to examine impossible to solve with uninformed search.
- General strategy is still to pick the move that leaves the least advantageous move for the opponent.
- The computer can’t examine every possibility in the allotted time (even with informed search).

Chess: Informed Search

- Of course, improving our search so we can examine more scenarios (which helps our heuristics perform better) is important as well.
- Also, preprocessing (i.e. if one is allowed to store partial searches examined ahead of time, one doesn’t have to recalculate the entire search space) can reduce the ‘cost’ of search.
- In many respects this similar to what humans do, only the computer has a more accurate, and larger, memory.

Chess: Heuristics

- Humans also use heuristics for chess because we have less brute-force power than (probably) a smartphone, so we can’t accurately examine every relevant branch either.
- Over the years grandmasters have developed a history of positions and (possibly) the best response. The memory of the positions and responses is the basis of many human heuristics.
- For computers, the evaluation function can be informed by this memory, so they once again have an advantage.

Beyond Logic—Randomness & Uncertainty

Board Games with Cards and/or Dice

- This is more challenging for a machine because it can no longer preprocess positions.
- Further the evaluation function can no longer be designed on the assumption that the next move will be the ideal move for the opponent.
- More importantly, the machine's response depends on a chance element at a point in the future. Therefore the machine's 'look-ahead' for it's response to the opponent has to be probabilistic, which means the machine may not make the move that would have been ideal, could it have known the future. Over many games though, the machine likely wins.

Discussion of Some Combo Games

- Computers don’t do better or worse than humans at games like Snakes and Ladders in which there are no choices, only chance and rules.
- Can a computer do better than a human at Monopoly® (absent side deals)? (For two-player, and multi-player).
  - Discuss.
- Pitiless Sorrow (why you should not play Sorry® against a machine).
  - Discussion of human elements of a game…