Adversarial Search (Games)

CPSC 383: Explorations in Artificial Intelligence and Machine Learning
Fall 2025

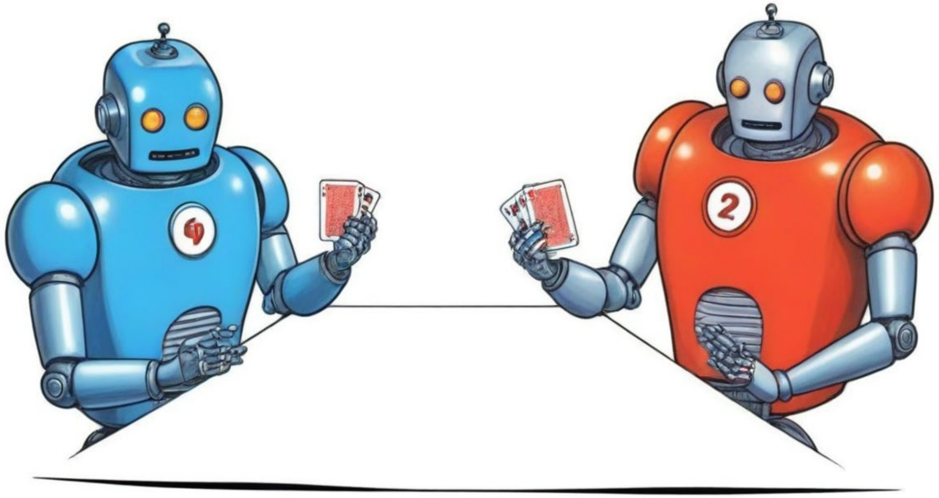
Jonathan Hudson, Ph.D
Associate Professor (Teaching)
Department of Computer Science
University of Calgary

August 27, 2025

UNIVERSITY OF CALGARY

Outline

- Multiplayer Games
- Games Theory
- Optimal Decisions in Games
    1. minimax decisions
    2. α–β pruning
    3. Monte Carlo Tree Search (MCTS)
- Resource limits and approximate evaluation
- Games of chance
- Games of imperfect information
- Limitations of Game Search Algorithms

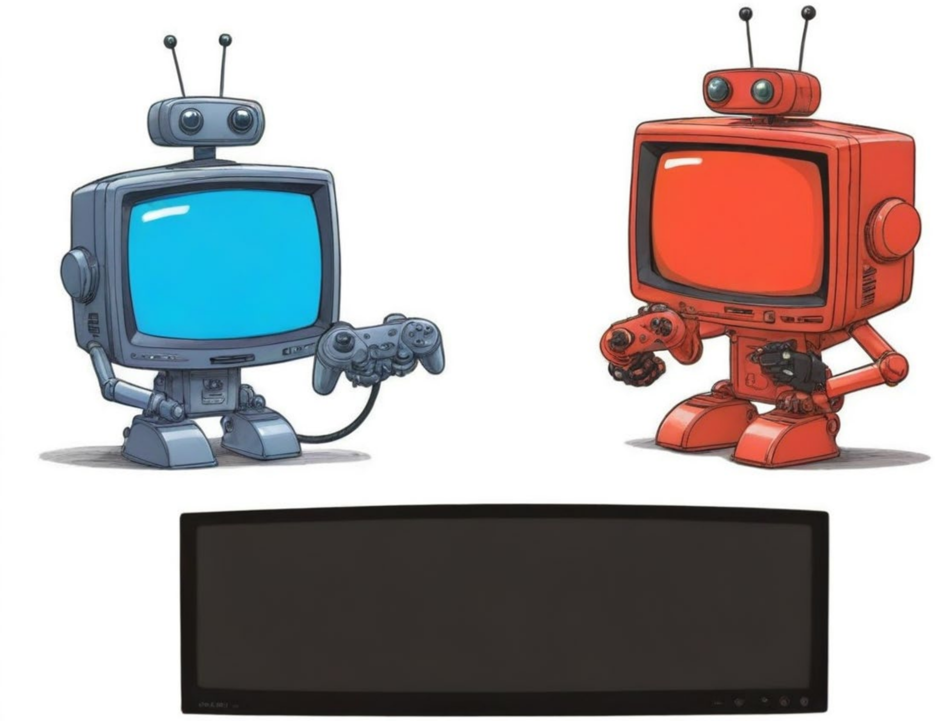


Multiplayer Games

Playing multi-player games

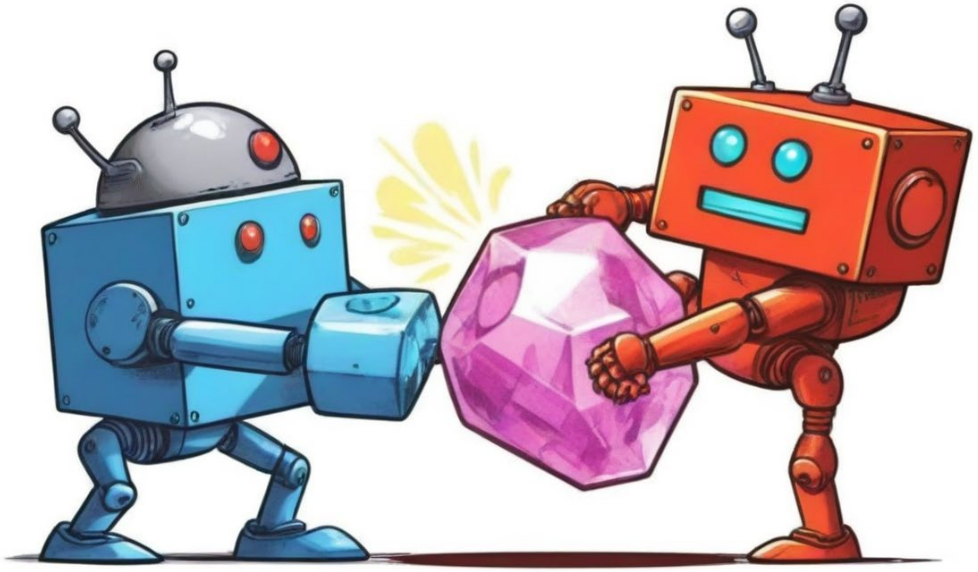
Problem: How to determine the best next move in a game like chess or checkers given a limited amount of computing time.

Idea: Search among the possible alternative moves and their consequences



Types of Games

- General Games
  - Sometime stochastic or imperfect information
  - Agents have independent utilities
  - Cooperation, indifference, competition, and more are all possible



- Zero-Sum Games
  - Agents have opposite utilities
  - Adversarial, pure competition

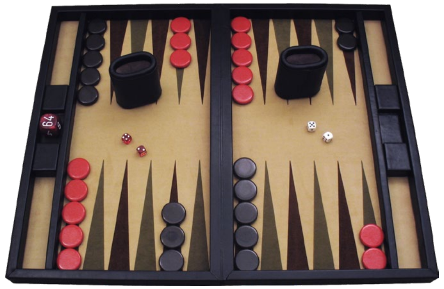
UNIVERSITY OF CALGARY

Types of Games

	deterministic	chance
perfect		
imperfect		

[This Photo] by Unknown Author is licensed under [CC BY] - BATTLESHIPS

[This Photo] by Unknown Author is licensed under [CC BY] - POKER



[This Photo] by Unknown Author is licensed under [CC BY-SA] - BACKGAMMON

[This Photo] by Unknown Author is licensed under [CC BY-SA] - GO

Types of Games

	deterministic	chance
perfect	 This Photo by Unknown Author is licensed under CC BY-SA	 This Photo by Unknown Author is licensed under CC BY-SA
imperfect	 This Photo by Unknown Author is licensed under CC BY	 This Photo by Unknown Author is licensed under CC BY

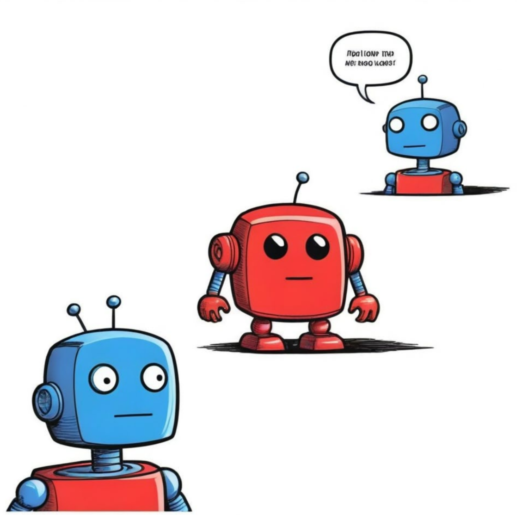
Types of Games

	deterministic	chance
perfect	chess, checkers, othello, go, tic tac toe	backgammon, monopoly
imperfect	battleships, blind tic tac toe	bridge, poker, scrabble, nuclear war

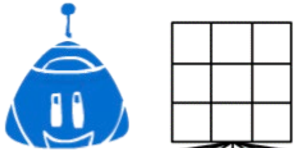
Games Theory

Our Basic ZeroSum Two Player Game Model

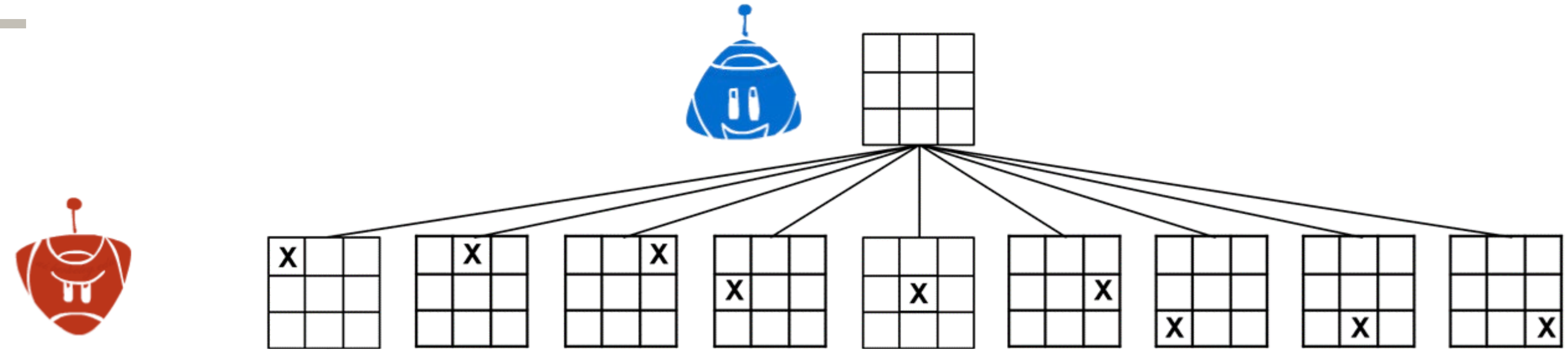
- *Fundamentally required to establish the definition of world for any AI problem*
- **Two players**
  - Max-min (always play for optimal gain)
  - Taking turns
  - Game state fully observable by opponent
- **Moves**: action
- **Position**: state
- **Zero sum**:
  - Good for one player == bad for the other.
    - No win-win outcomes for a move/action, game can still end in draw.
- *Any one of these is altered and everything changes*

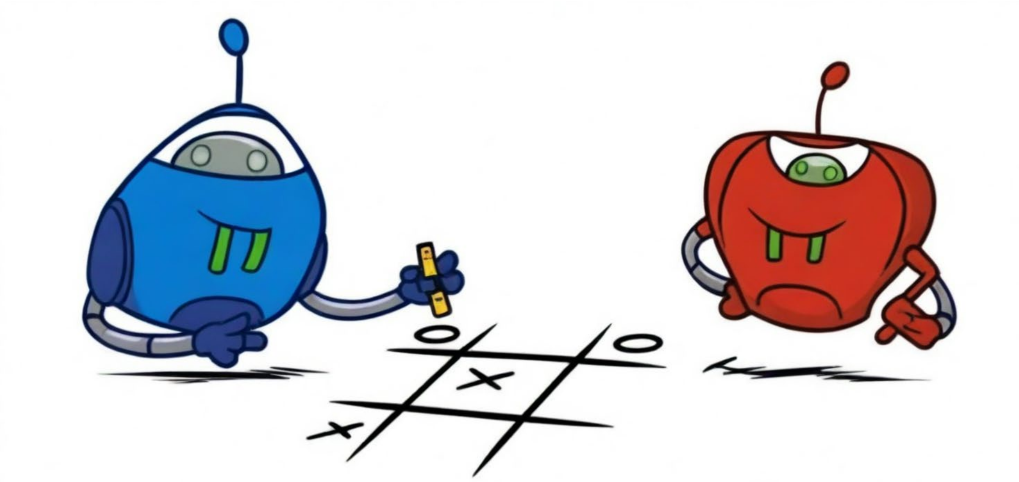


Tic-TacToe Game tree

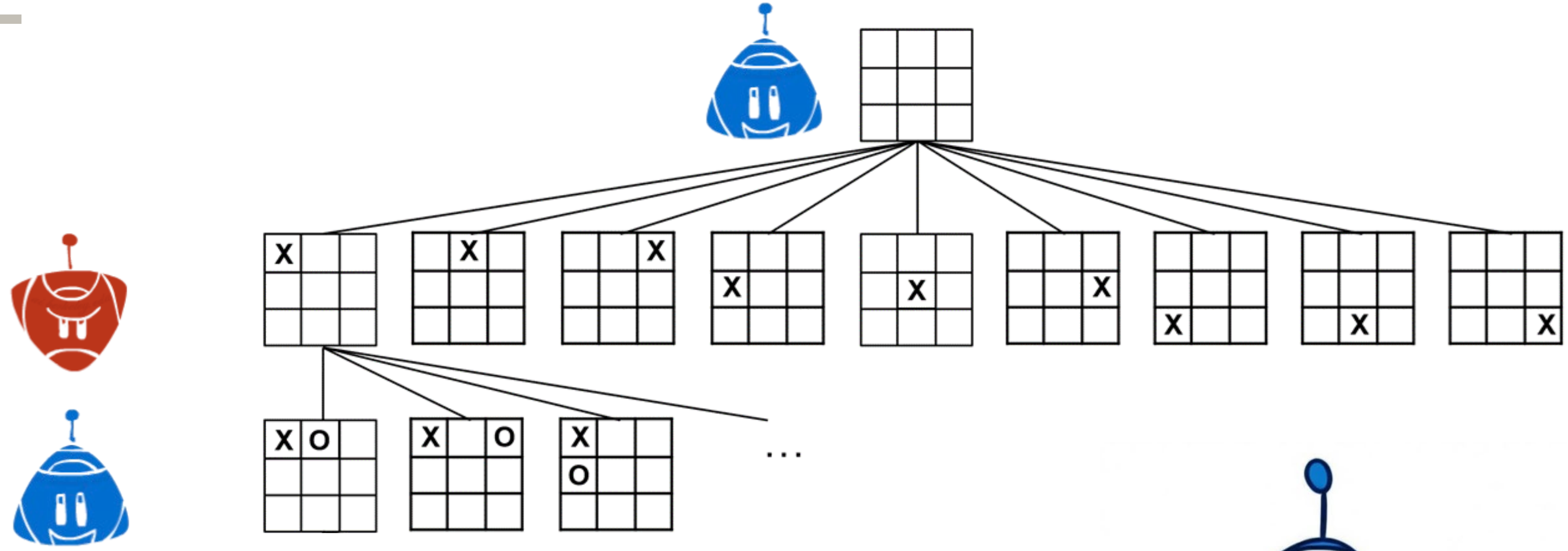


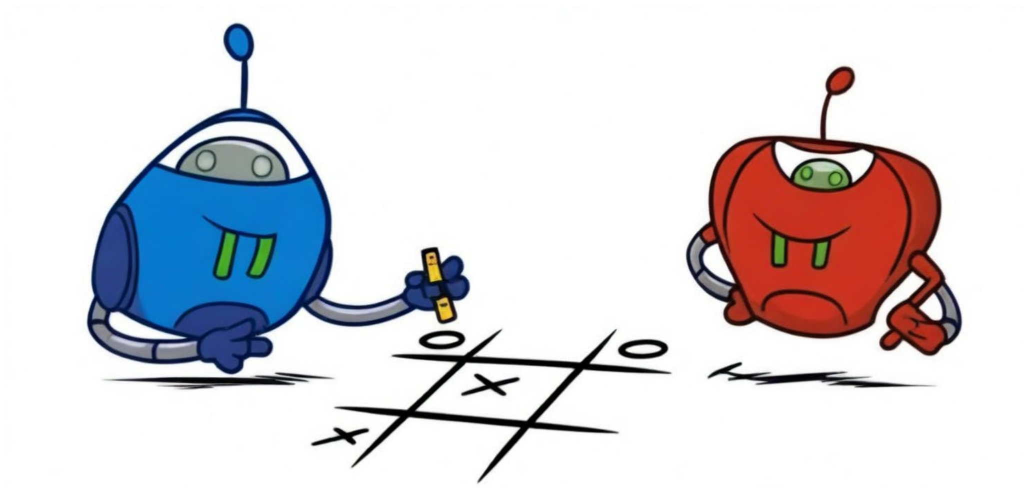
Tic-TacToe Game tree

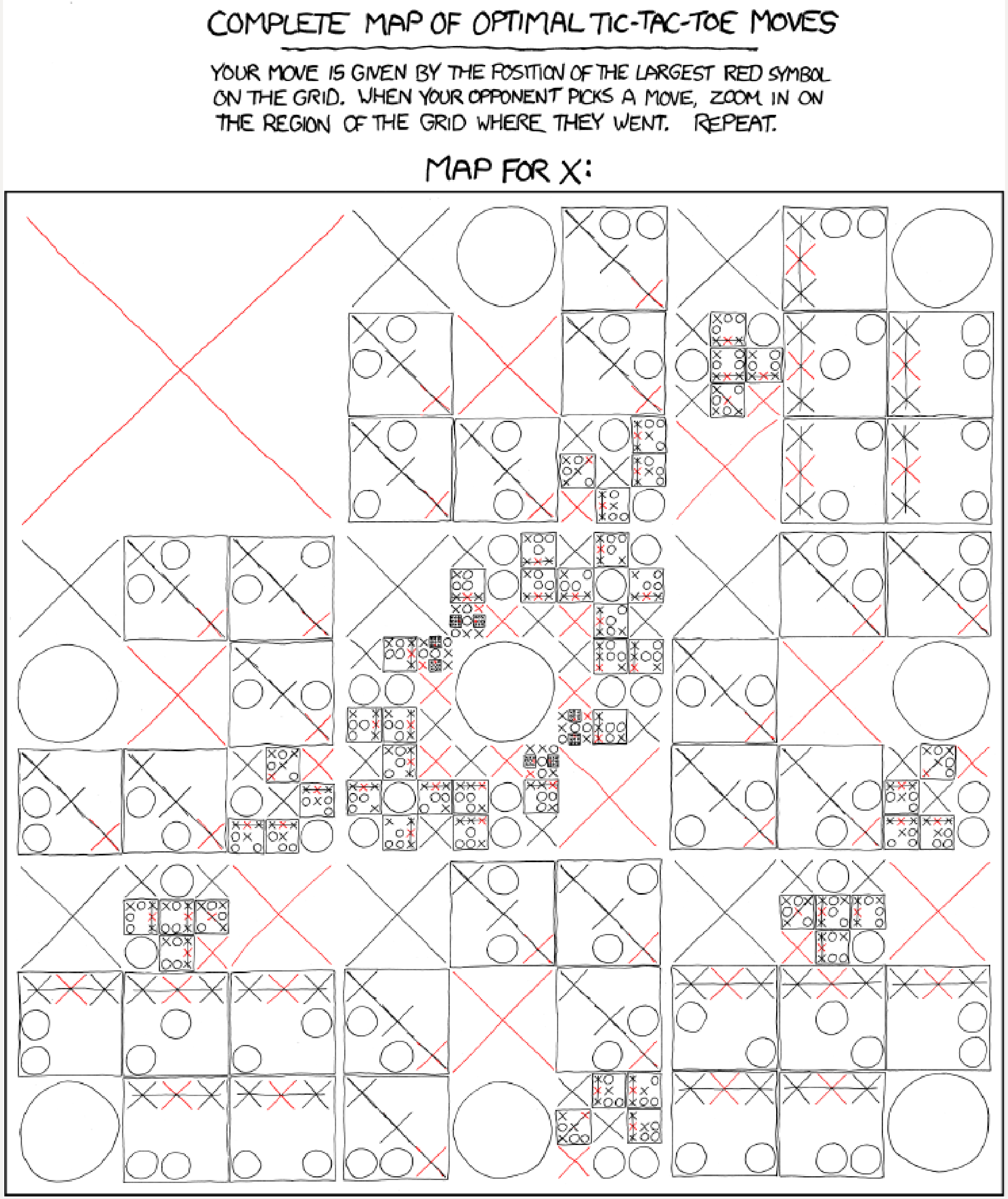




Tic-TacToe Game tree






COMPLETE MAP OF OPTIMAL TIC-TAC-TOE MOVES
YOUR MOVE IS GIVEN BY THE POSITION OF THE LARGEST RED SYMBOL ON THE GRID. WHEN YOUR OPPONENT PICKS A MOVE, ZOOM IN ON THE REGION OF THE GRID WHERE THEY WENT. REPEAT.
MAP FOR X:

Minimax

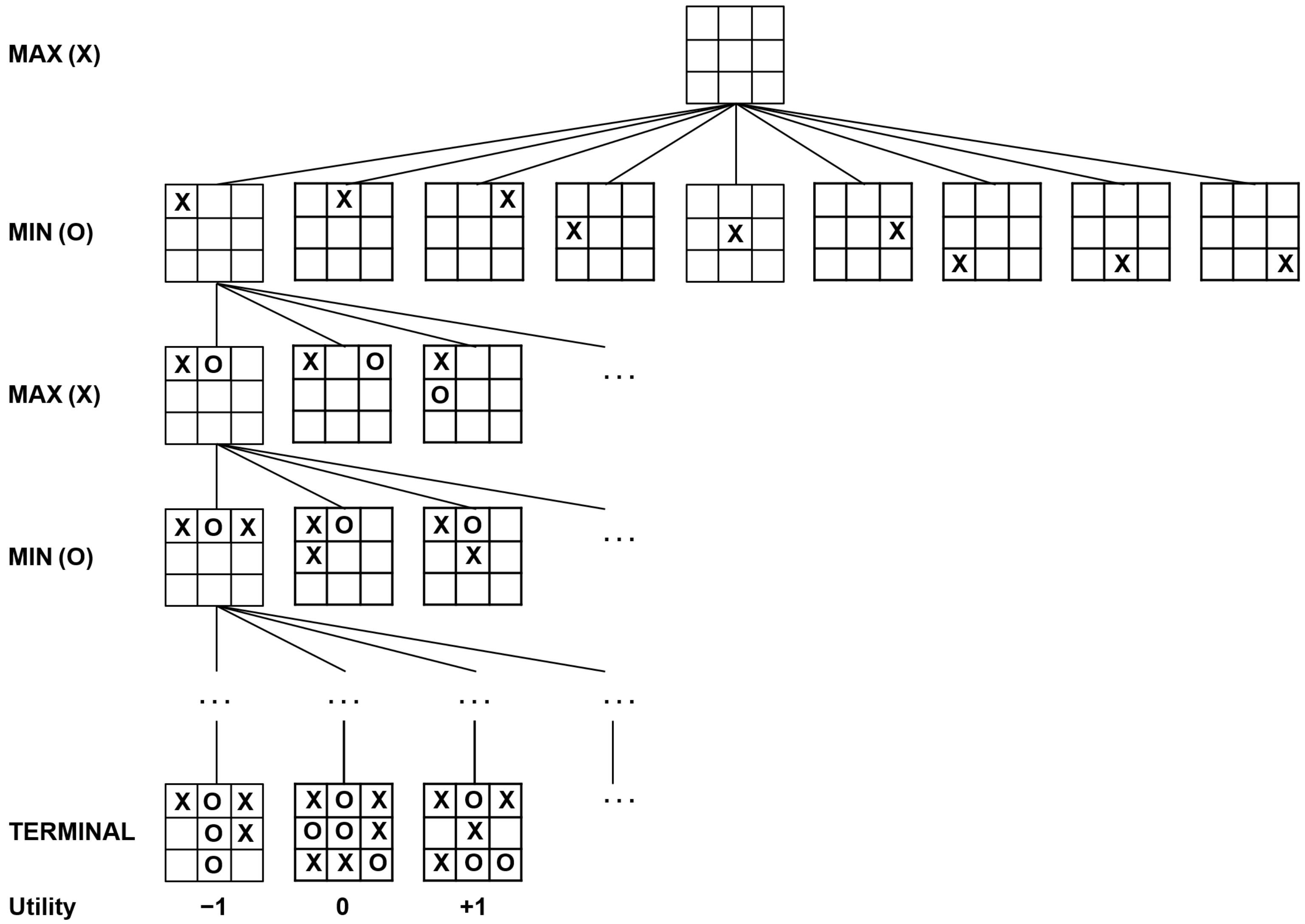
Subtitle

Minimax

- Perfect play for deterministic, perfect-information games
- Idea: choose move to position with highest minimax value = best achievable payoff against best play

UNIVERSITY OF CALGARY

Tic-TacToe Game tree (minimax)

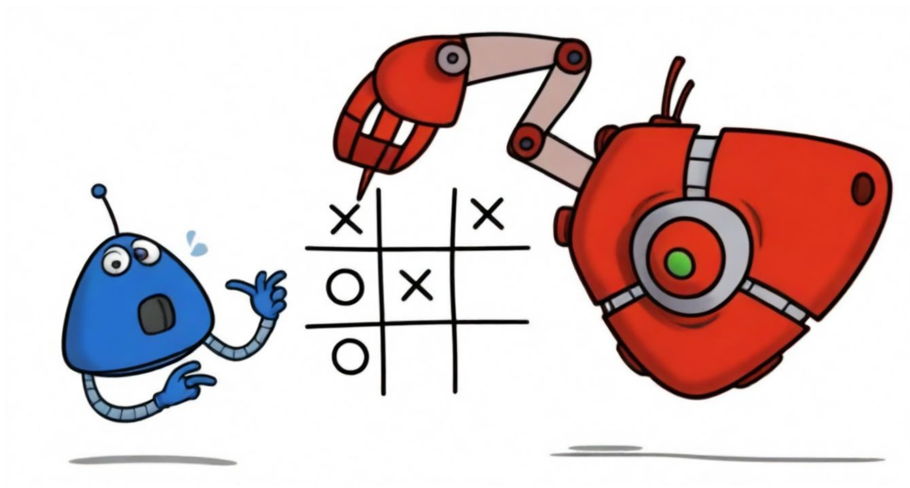


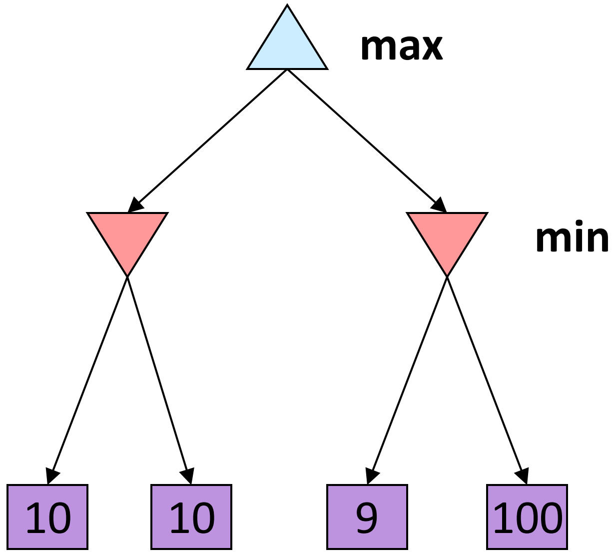
Properties of minimax

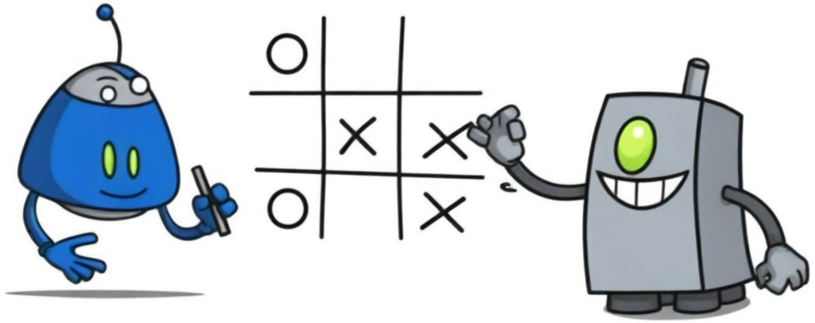
- Complete: Yes, if tree is finite (chess has specific rules for this)

- Optimal: Yes, against an optimal opponent.

Minimax Properties







Non-optimal player?

UNIVERSITY OF CALGARY

Properties of minimax

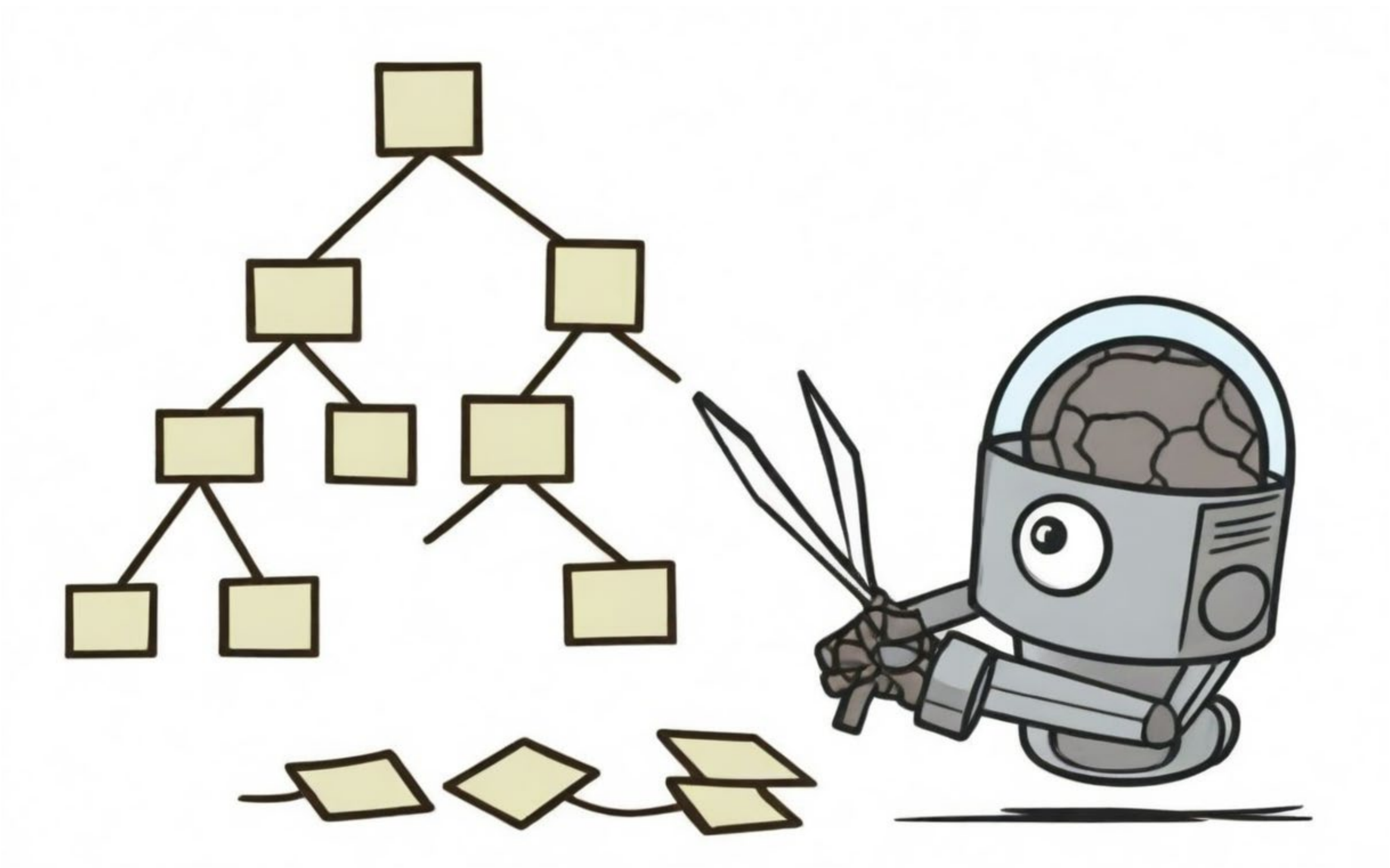
- For chess, branches ≈ 35, depth ≈ 100 for "reasonable" games
  - exponential b^m
    - ⇒ exact solution completely infeasible

- But do we need to explore every path?

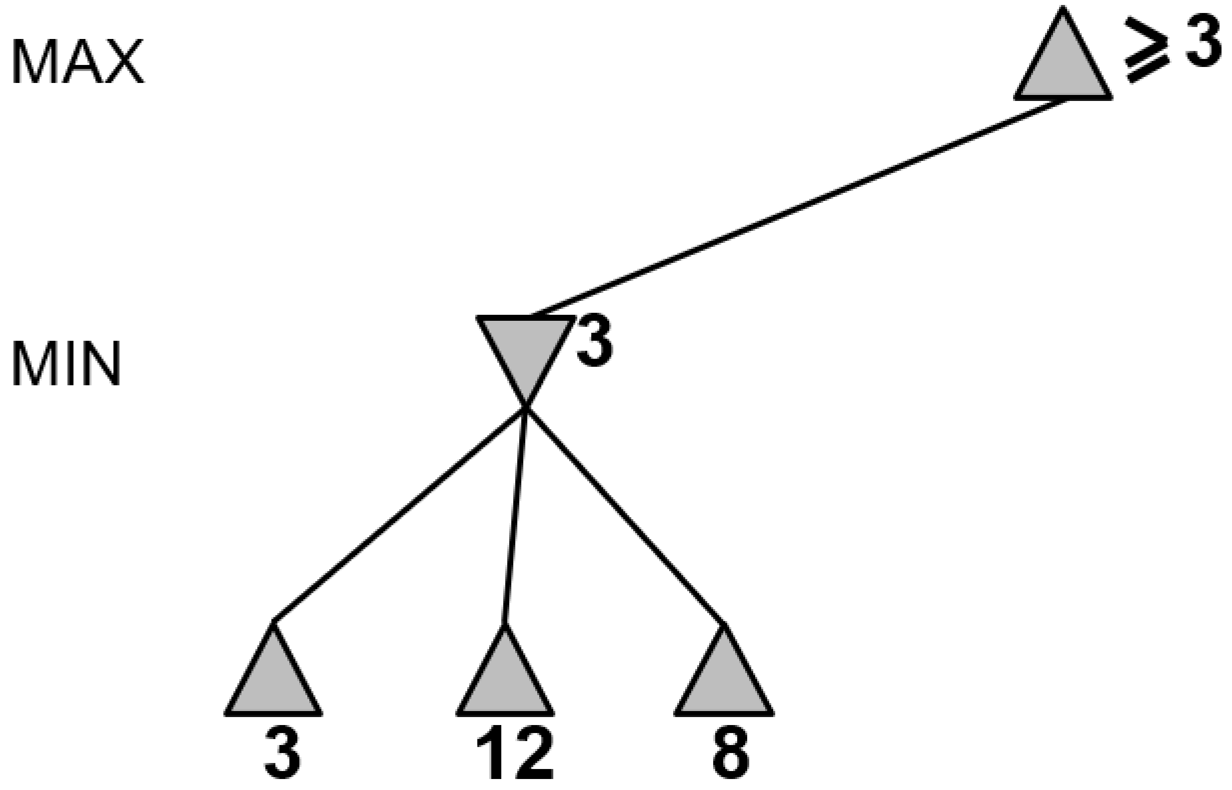
αlpha-βeta pruning

Subtitle

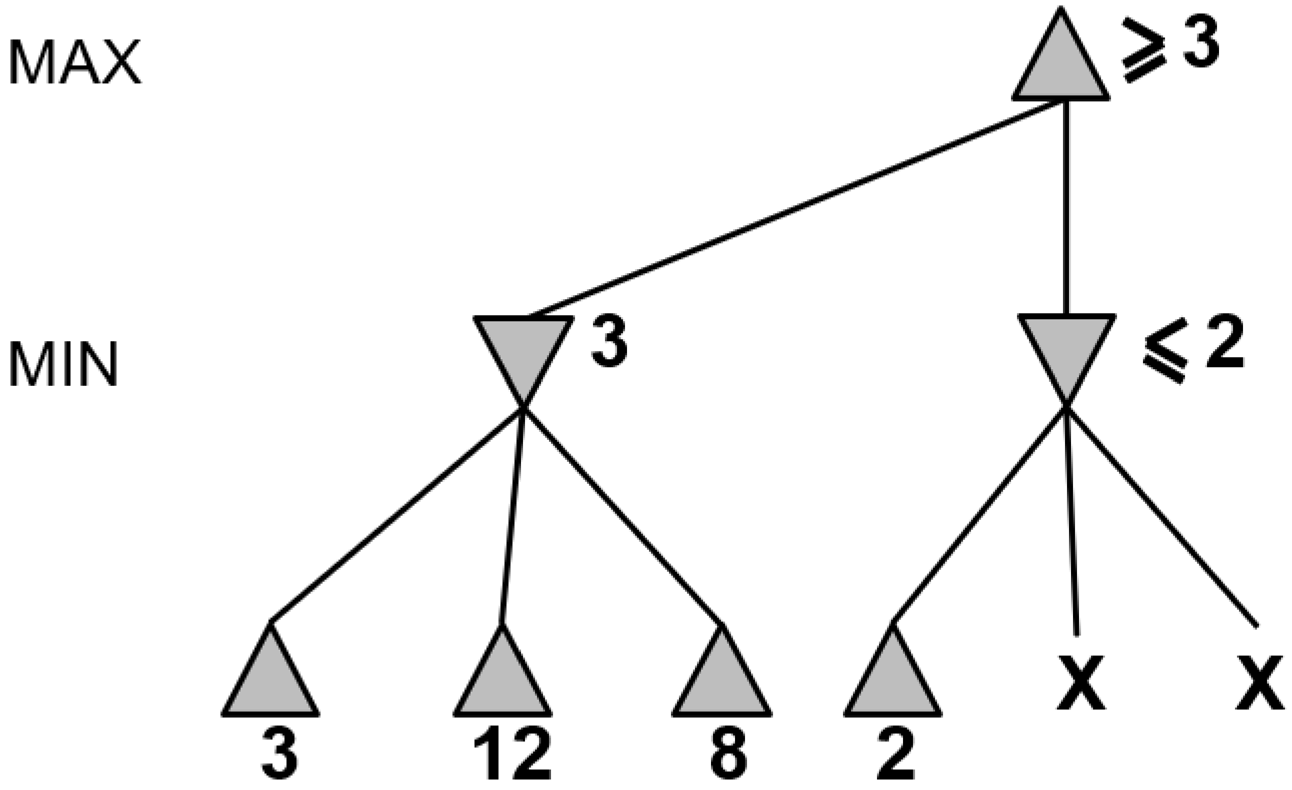
Game Tree Pruning



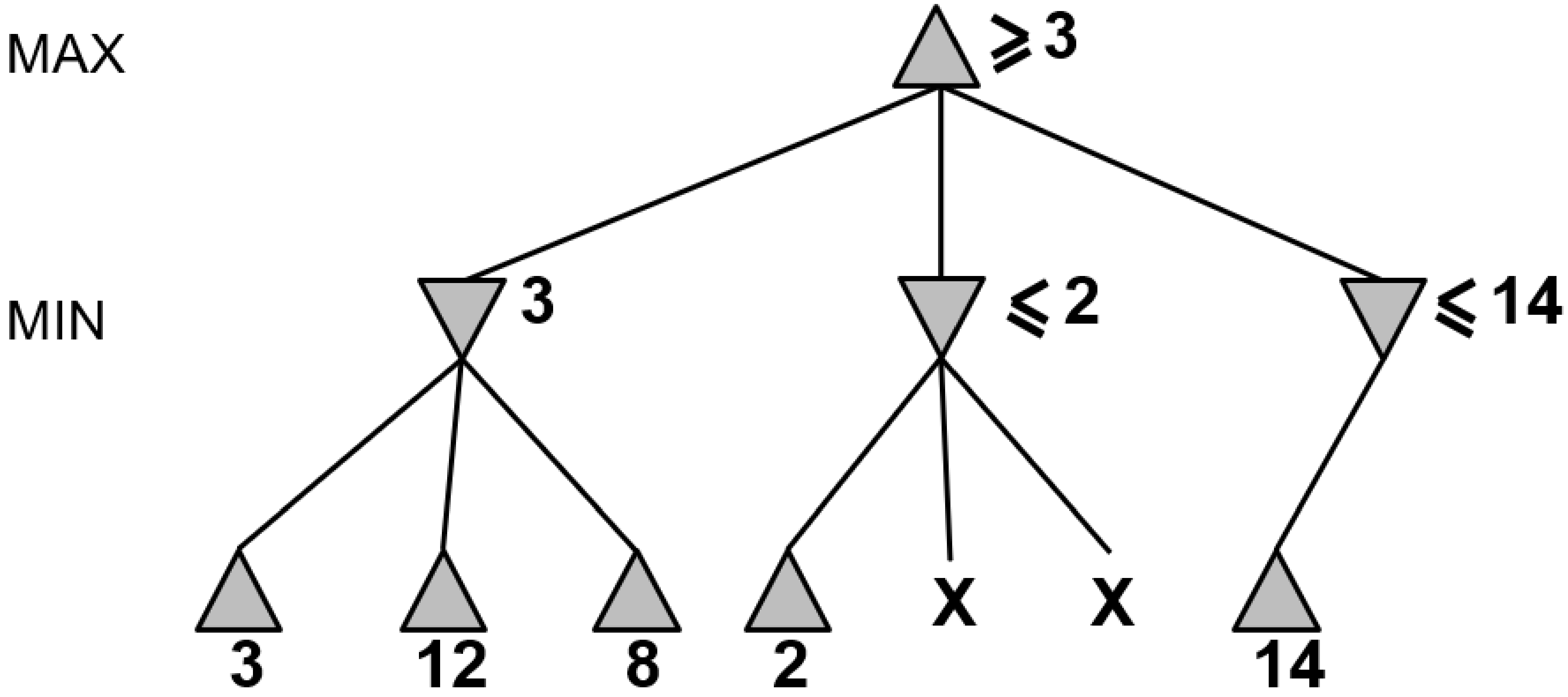
First step improvement (αlpha-βeta pruning)



First step improvement (αlpha-βeta pruning)

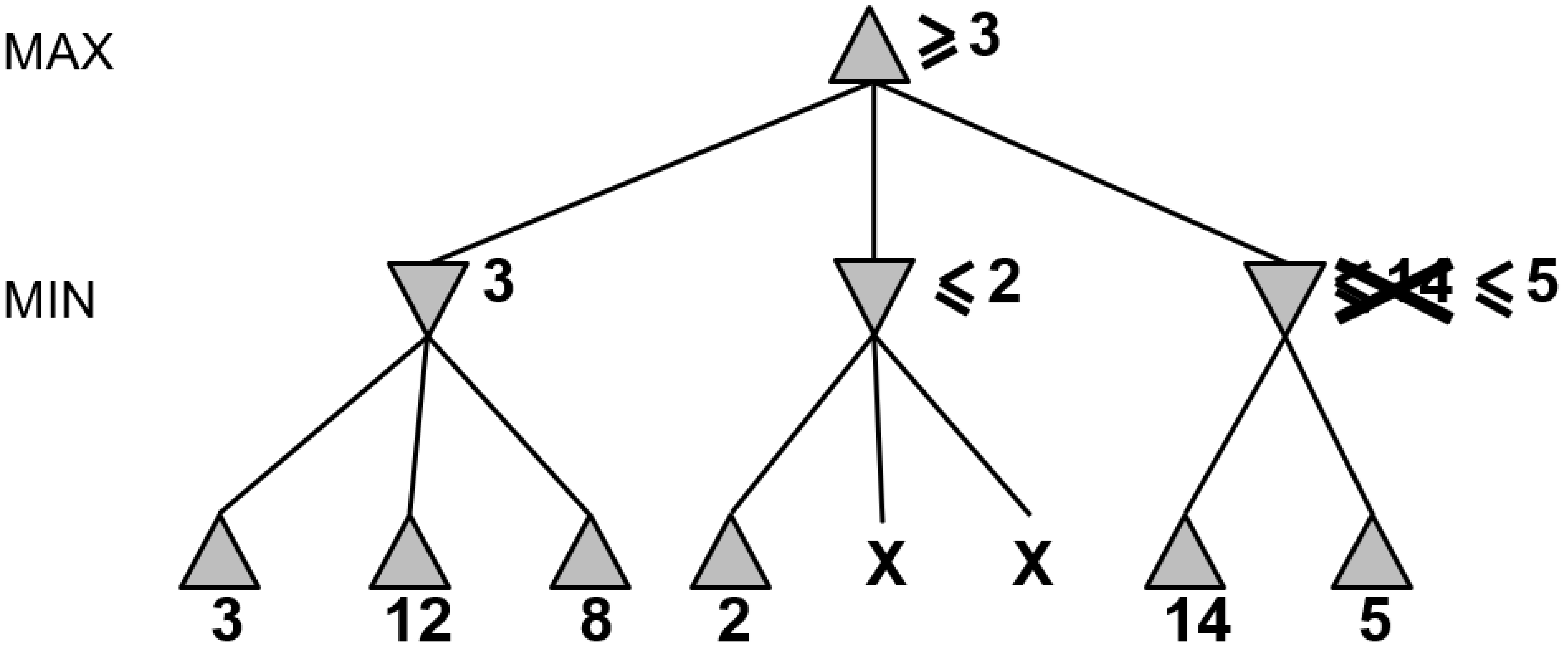


First step improvement (αlpha-βeta pruning)

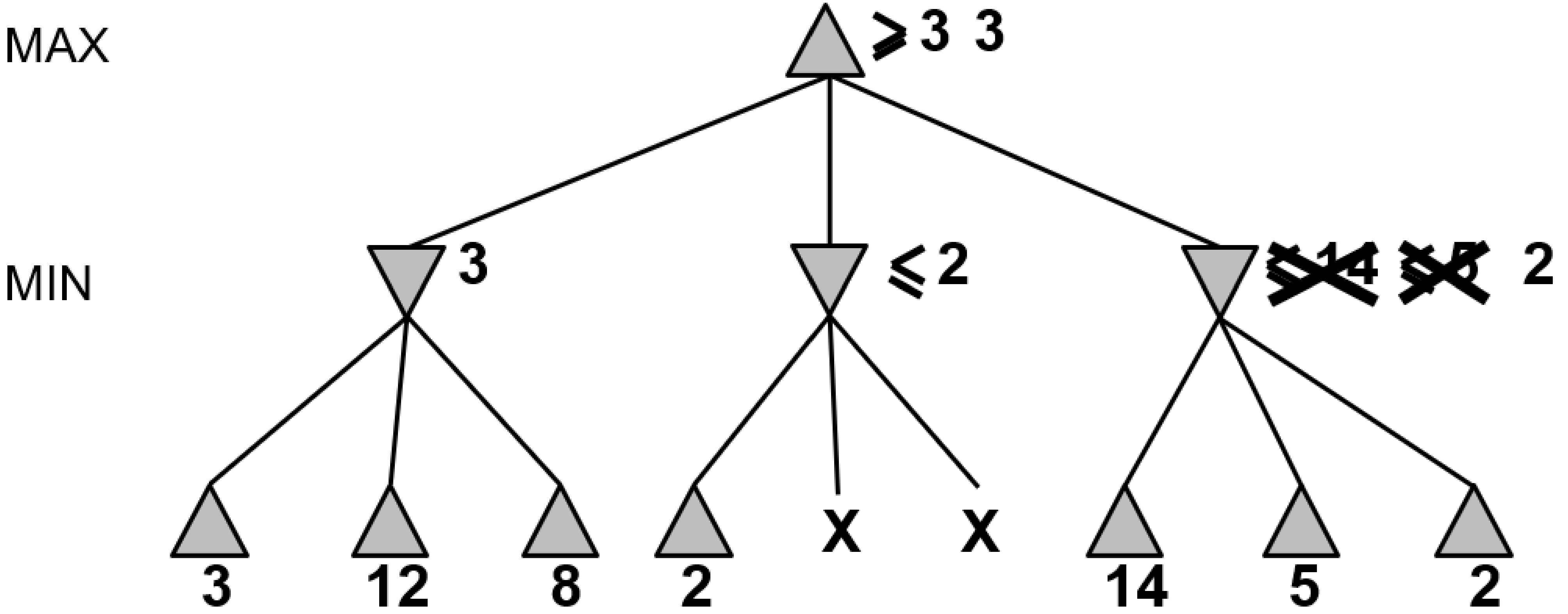


UNIVERSITY OF CALGARY

First step improvement (αlpha-βeta pruning)



First step improvement (αlpha-βeta pruning)



UNIVERSITY OF CALGARY

State of Games

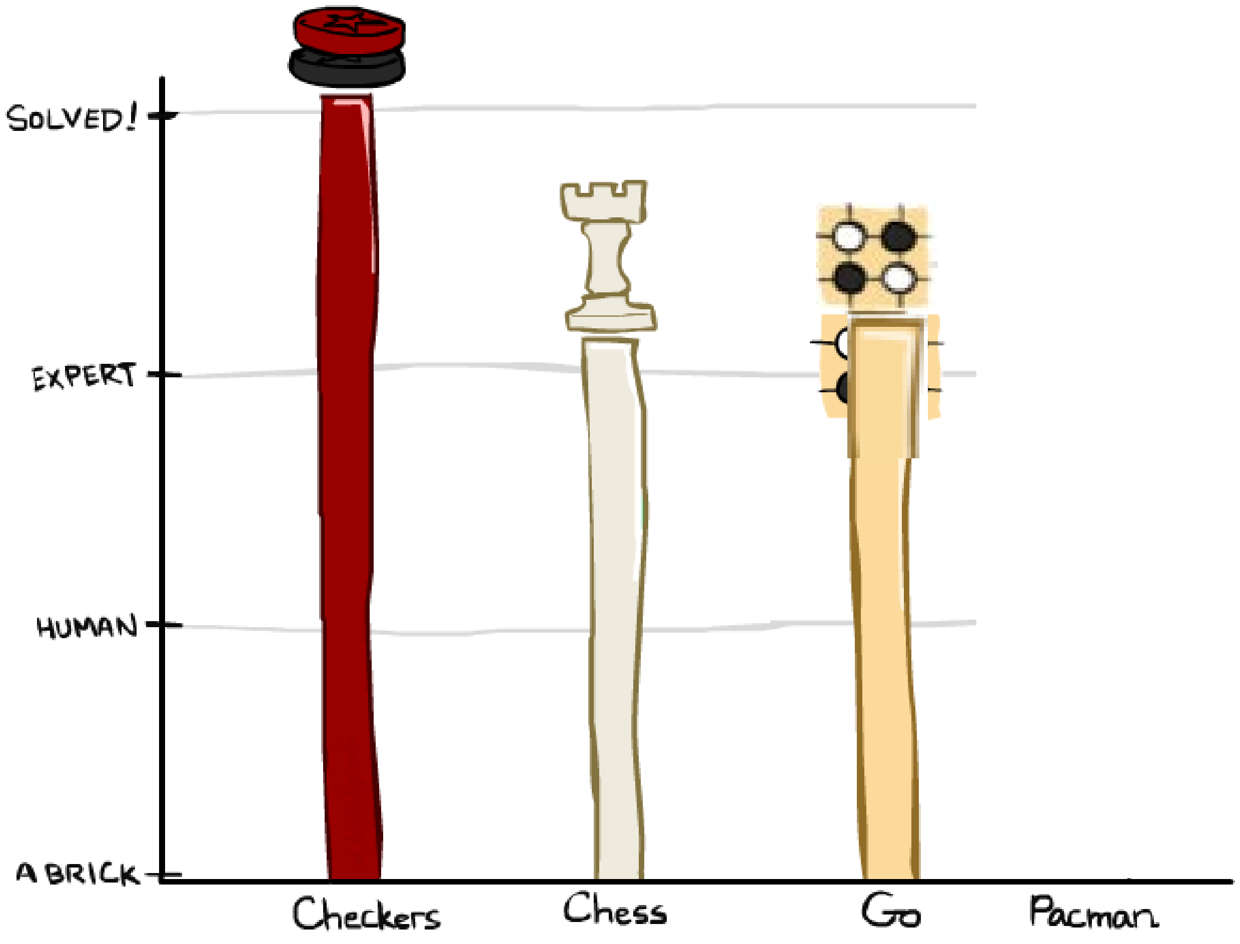
Subtitle

Zero-Sum Game Games ☺

- **Checkers:** 1950: First computer player. 1994: First computer champion: UofA Jonathan Schaeffer's Chinook ended 40-year-reign of human champion Marion Tinsley using complete 8-piece endgame. 2007: Checkers solved!

- **Chess:** 1997: Deep Blue defeats human champion Gary Kasparov in a six-game match. Deep Blue examined 200M positions per second, used very sophisticated evaluation and undisclosed methods for extending some lines of search up to 40 ply. Current programs are even better, if less historic.

- **Go :2016: Alpha GO defeats human champion. Uses Monte Carlo Tree Search, learned evaluation function.**

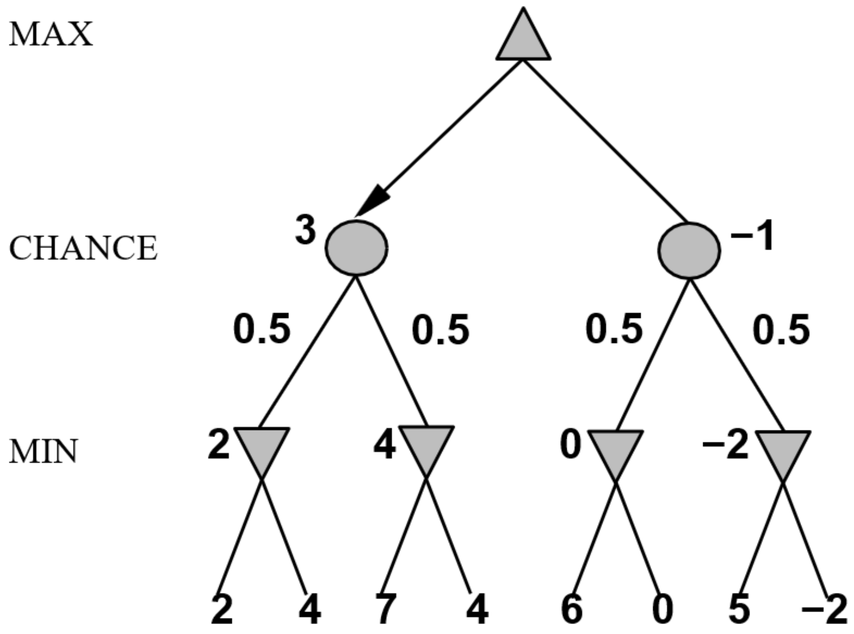


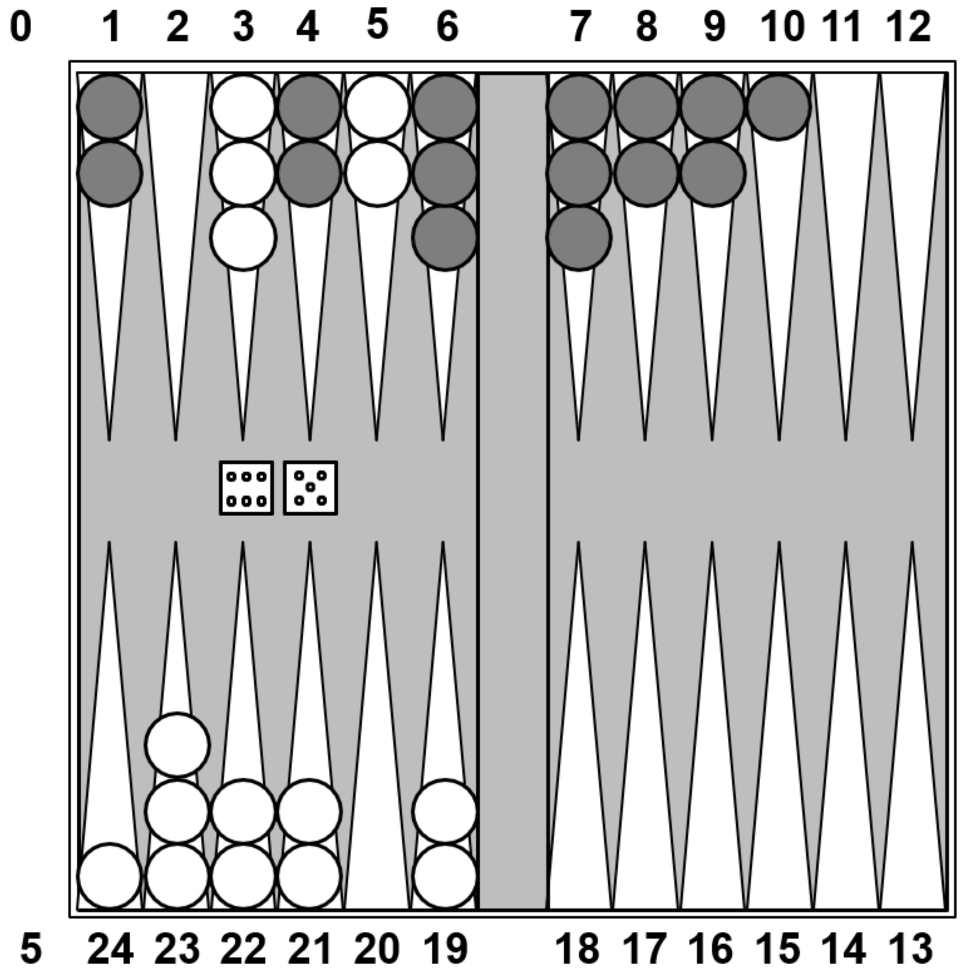
UNIVERSITY OF CALGARY

Other game types

Non-deterministic Games

- In nondeterministic games, chance introduced by dice, card-shuffling
- Simplified example with coin-flipping:





- Need average case reasoning

Monte Carlo Tree Search (MCTS)

- Monte Carlo Tree Search (MCTS)
  - Strategy does not use a heuristic evaluation function.
  - Value of a state is estimated as the average utility over number of simulations
  - Evaluations are estimations instead of determinate UTILITY

Games of imperfect information (Poker, Bridge)

- E.g., card games, where opponent's initial cards are unknown

- Idea: compute the minimax value of each action in each deal,
  - then choose the action with highest expected value over all deals

Games of imperfect information (Poker)

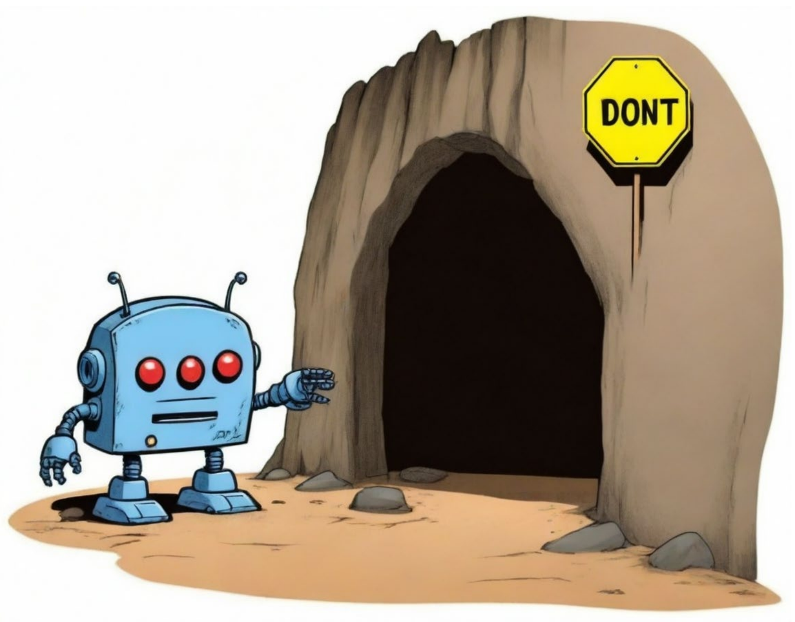
- Texas Two-player Hold’em Poker UofA Jonathan’s Schaeffer “””solved”””
  - on average the program is so good that a human would have no chance of ever edging ahead of it, even if the two played 60 million hands.
  - "for all purposes that anyone would ever care about, we've solved the game“
  - https://www.science.org/content/article/texas-hold-em-poker-solved-computer

Common Sense

The Dangers of Optimism and Pessimism

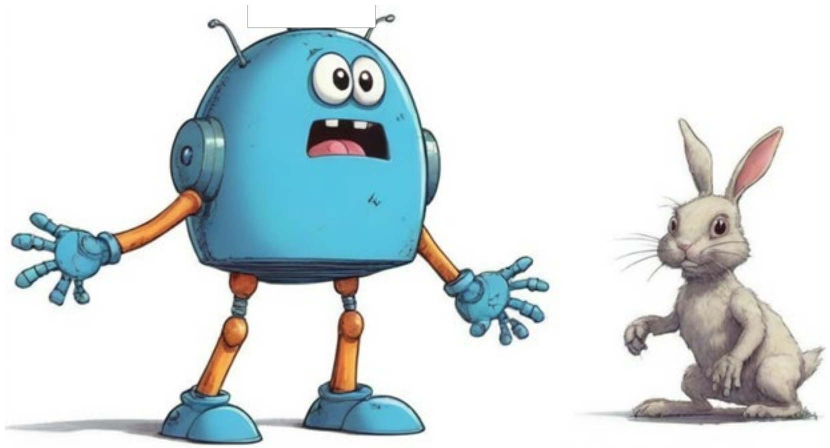
Dangerous Optimism

Assuming chance when the world is adversarial



Dangerous Pessimism

Assuming the worst case when it's not likely



UNIVERSITY OF CALGARY

Summary

Subtitle

Summary

- Minimax algorithm: selects optimal moves by a depth-first enumeration of the game tree.
- Alpha–beta algorithm: greater efficiency by eliminating subtrees
- Evaluation function: a heuristic that estimates utility of state.
- Monte Carlo tree search (MCTS): no heuristic, play game to the end with rules and repeated multiple times to determine optimal moves during playout.

Next...advanced local search

Jonathan Hudson, Ph.D.
jwhudson@ucalgary.ca
https://cspages.ucalgary.ca/~jwhudson/

UNIVERSITY OF CALGARY