

Adversarial Search

CS51A
David Kauchak
Spring 2022

Some material borrowed from :
Sara Owsley Sood and others

1

Admin

Assignment 10

2

A quick review of search

Problem solving via search:

- To define the state space, define three things:
 - is_goal
 - next_states
 - starting state

Uninformed search vs. informed search

- what's the difference?
- what are the techniques we've seen?
- pluses and minuses?

3

Why should we study games?

Clear success criteria

Important historically for AI

Fun ☺

Good application of search

- hard problems (chess 35^{100} states in search space, 10^{40} legal states)

Some real-world problems fit this model

- game theory (economics)
- multi-agent problems

4

Types of games

What are some of the games you've played?

5

Types of games: game properties

single-player vs. 2-player vs. multiplayer

Fully observable (perfect information) vs. partially observable

Discrete vs. continuous

real-time vs. turn-based

deterministic vs. non-deterministic (chance)

6

Strategic thinking = intelligence

For reasons previously stated, two-player games have been a focus of AI since its inception...



Important question: Is strategic thinking the same as intelligence?

7

Strategic thinking = intelligence

Humans and computers have different relative strengths in these games:

humans

?



computers

?

8

Strategic thinking ? = intelligence

Humans and computers have different relative strengths in these games:

humans

good at evaluating the strength of a board for a player



computers

good at looking ahead in the game to find winning combinations of moves

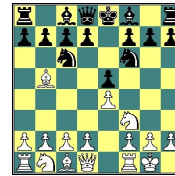
9

Strategic thinking ? = intelligence

How could you figure out how humans approach playing chess?

humans

good at evaluating the strength of a board for a player



10

How humans play games...

An experiment was performed in which chess positions were shown to novice and expert players...



- experts could reconstruct these perfectly
- novice players did far worse...

11

How humans play games...

Random chess positions (not legal ones) were then shown to the two groups

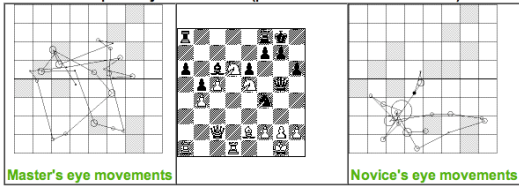


experts and novices did just as badly at reconstructing them!

12

People are still working on this problem...

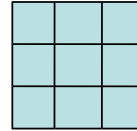
Example of eye movements (presentation time = 5 seconds)



http://people.brunel.ac.uk/~hsstffg/frg-research/chess_expertise/

13

Tic Tac Toe as search

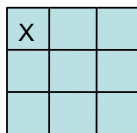
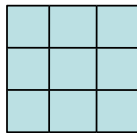


If we want to write a program to play tic tac toe, what question are we trying to answer?

Given a state (i.e. board configuration), what move should we make!

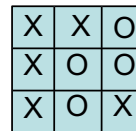
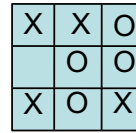
14

Tic Tac Toe as search



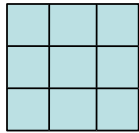
15

Tic Tac Toe as search



16

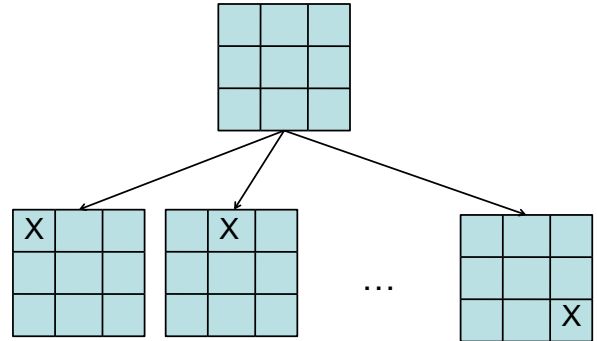
Tic Tac Toe as search



How can we pose this as a search problem?

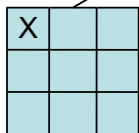
17

Tic Tac Toe as search



18

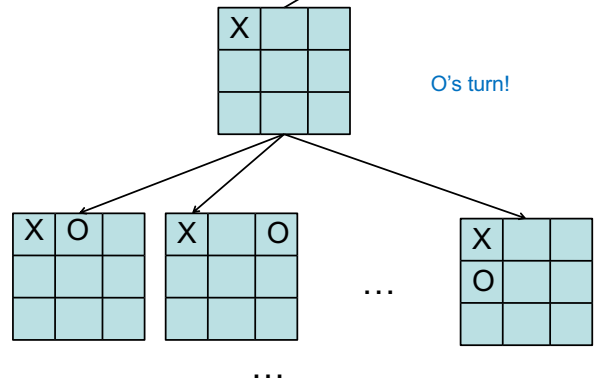
Tic Tac Toe as search



Now what?

19

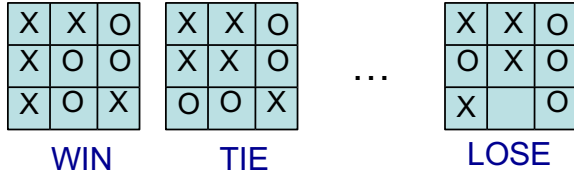
Tic Tac Toe as search



20

Tic Tac Toe as search

Eventually, we'll get to a leaf



How does this help us?

Try and make moves that move us towards a win, i.e. where there are leaves with a WIN.

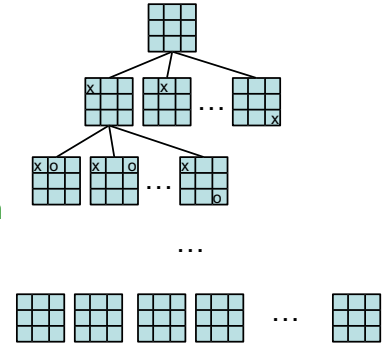
21

Tic Tac Toe

X's turn

O's turn

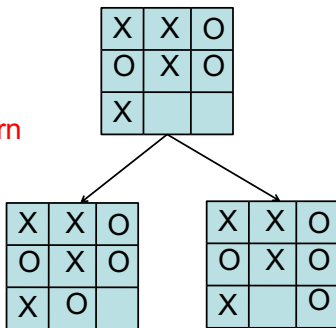
X's turn



22

I'm X, what will 'O' do?

O's turn



qq12314567890-!@;lkjhgfdsazxvbbvnm,./

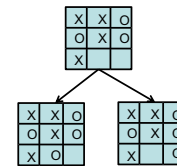
23

Minimizing risk

The computer doesn't know what move O (the opponent) will make

It can assume that it will try and make the best move possible

Even if O actually makes a different move, we're no worse off. Why?



24

Optimal Strategy

An **Optimal Strategy** is one that is at least as good as any other, no matter what the opponent does

- If there's a way to force the win, it will
- Will only lose if there's no other option

25

Defining a scoring function

X	X	O		X	X	O		X	X	O
X	O	O		X	X	O	...	O	X	O
X	O	X		O	O	X		X		O
WIN			TIE			LOSE				
+1			0			-1				

Idea:

- define a function that gives us a "score" for how good each state is
- higher scores mean better

26

Defining a scoring function

Our (X) turn

X	X	O
	O	O
X	O	X

What should be the score of this state?

+1: we can get to a win

27

Defining a scoring function

Opponent's (O) turn

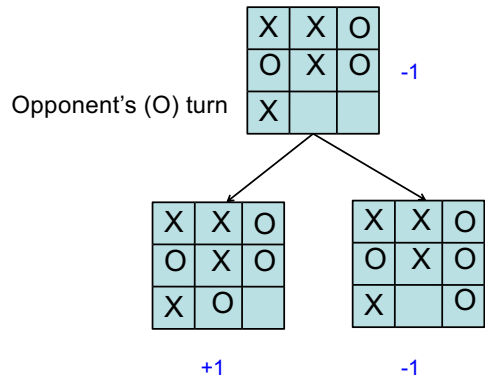
X	X	O
O	X	O
X		

What should be the score of this state?

-1: opponent can get to a win

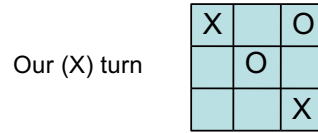
28

Defining a scoring function



29

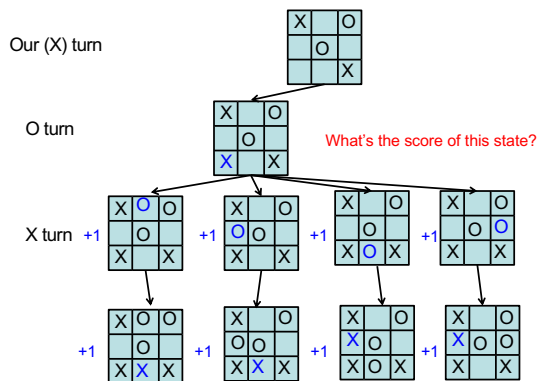
Defining a scoring function



What should be the score of this state?

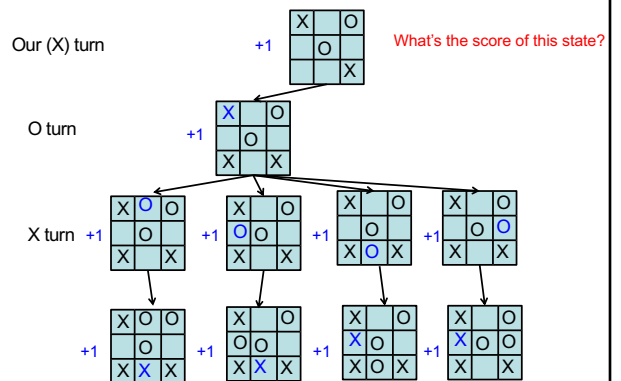
30

Defining a scoring function



31

Defining a scoring function



32