

Introduction to Artificial Intelligence

COMP 3501 / COMP 4704-4

Lecture 4: Game Playing

Prof. Nathan Sturtevant

Online Search Agents with Unknown Environments

- Previous search agents find a solution and then execute
 - What if the environment is initially unknown?
 - What if the environment can only be partially sensed?
- Online agents interleave search and actions
 - But, they have a model of the environment
 - eg may have a heuristic function
 - Able to plan about the world

Nathan Sturtevant

Introduction to Artificial Intelligence

Online agents

- Assume the world is safely explorable
 - A goal can always be reached
 - Undirected edges
- Free space assumption
 - The unexplored world is empty
 - Only add to world representation
- Agent-centered
 - Can only sense in the local area around the agent

Nathan Sturtevant

Introduction to Artificial Intelligence

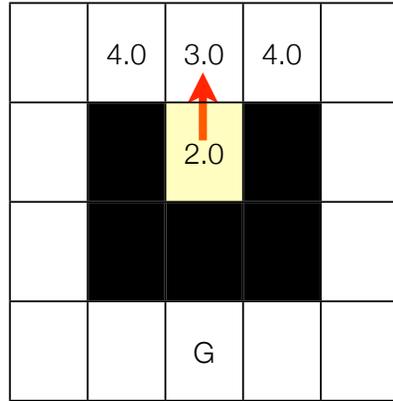
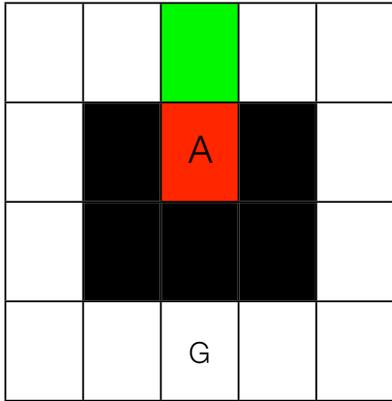
Simple online agents

- Hill-climbing search
- Random walk
- LRTA*
 - Learning Real-Time A*
 - Learns a better heuristic to guarantee completeness

Nathan Sturtevant

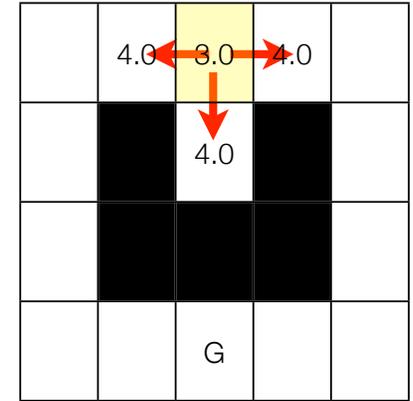
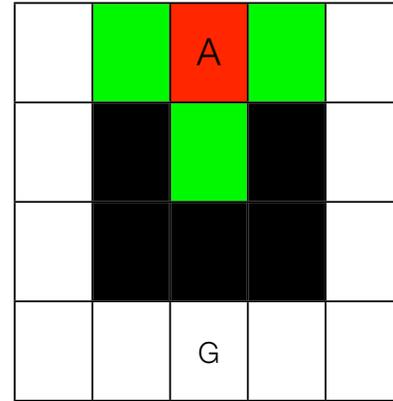
Introduction to Artificial Intelligence

Learning Example



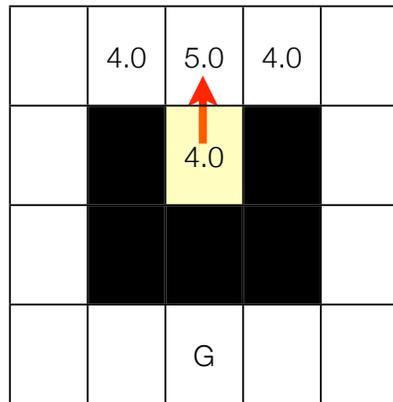
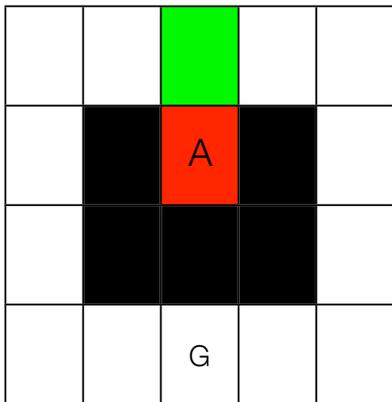
Heuristic

Learning Example



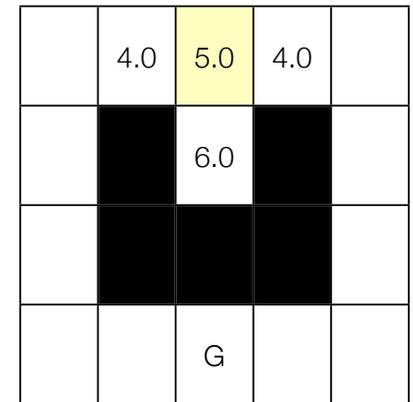
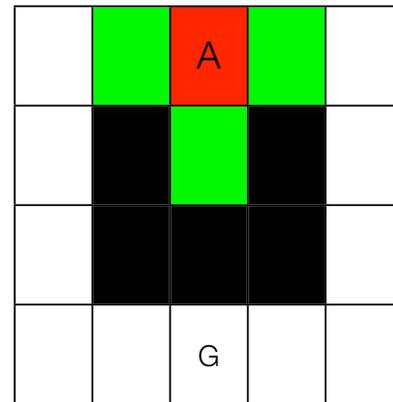
Heuristic

Learning Example



Heuristic

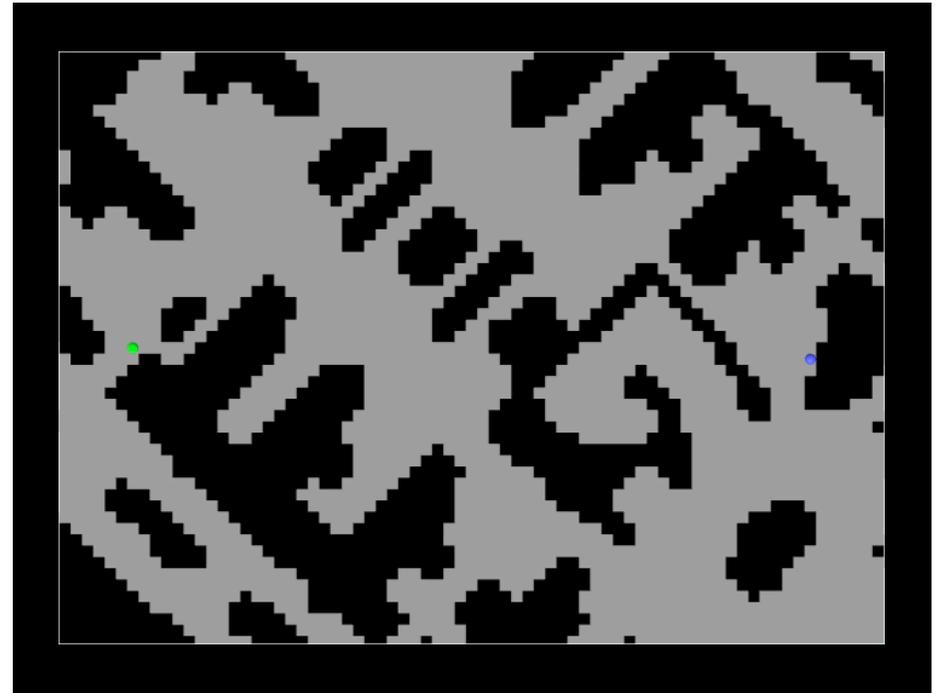
Learning Example



Heuristic

LRTA*

- LRTA* is one of a large class of algorithms
 - Guaranteed to solve a problem in at most $O(n^2)$ steps
 - Actual performance can look very poor
- Sometimes used in episodic framework



Today

- Discuss homework
- Chapter 5:
 - Two-player zero-sum games
 - Minimax / alpha-beta pruning
 - Search enhancements
 - Game history

Project

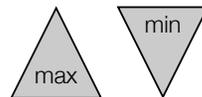
- Graduate Students Only
 - October 5: Proposal & timeline
 - November 2: Status report
 - November 19: Due
- May qualify for advanced programming credit

Overview

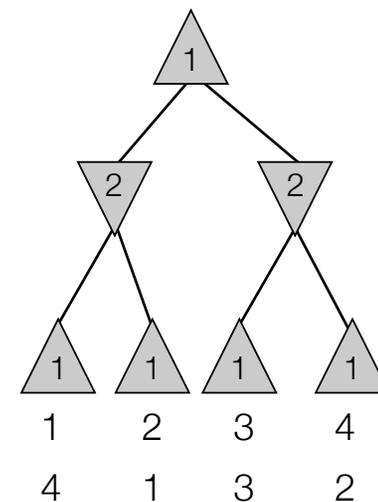
- Two-player zero-sum games
 - Have a strategy guaranteed to win no matter the opponent strategy
 - Nash equilibrium
- Doesn't matter if the game is partially observable
 - Deterministic games have "pure" strategies

Minimax

- Min player and Max player
 - Max player tries to find strategy for maximum score
 - Min player tries to find strategy for minimum score
- Depth-First Search
 - Find minimax value of each state recursively



Example



Minimax Pseudo-Code

Minimax()

 GetMaxVal()

GetMaxVal()

 if (game over) return game value

 currVal $\leftarrow -\infty$

 for each successor s in 1... # successors

 ApplyMove(s)

 currVal = max(currVal, GetMinVal())

 UndoMove(s)

 return currVal

Minimax

- Analysis
 - $O(d)$ memory
 - $O(N) = O(b^d)$ time
- What if we don't have time to do the whole tree?

Minimax Pseudo-Code (2)

Minimax(depth)

 GetMaxVal(depth)

GetMaxVal(depth)

 if (depth == 0) return CutoffEval();

 currVal $\leftarrow -\infty$

 for each successor s in 1... # successors

 ApplyMove(s)

 currVal = max(currVal, GetMinVal(depth-1))

 UndoMove(s)

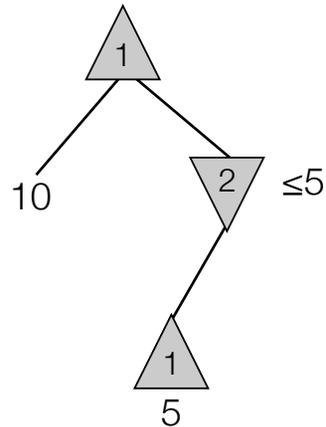
 return currVal

How should we choose our eval?

- Only need an ordering on the preference of leaf values
- But, often normalize values
 - Choose set of useful features & weights
 - $f_1 \cdot w_1 + f_2 \cdot w_2 + \dots + f_n \cdot w_n$
 - In chess, just material value of pieces plus a good search will play somewhat reasonable chess

Can we do better?

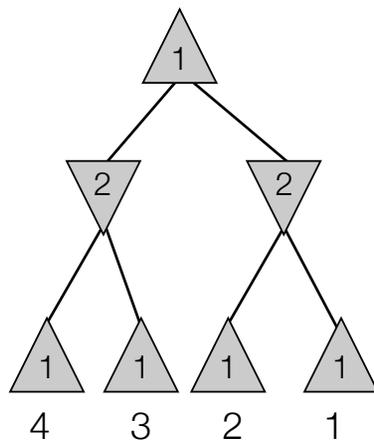
- Don't have to search a whole tree to know the value of the tree
- Simple example:



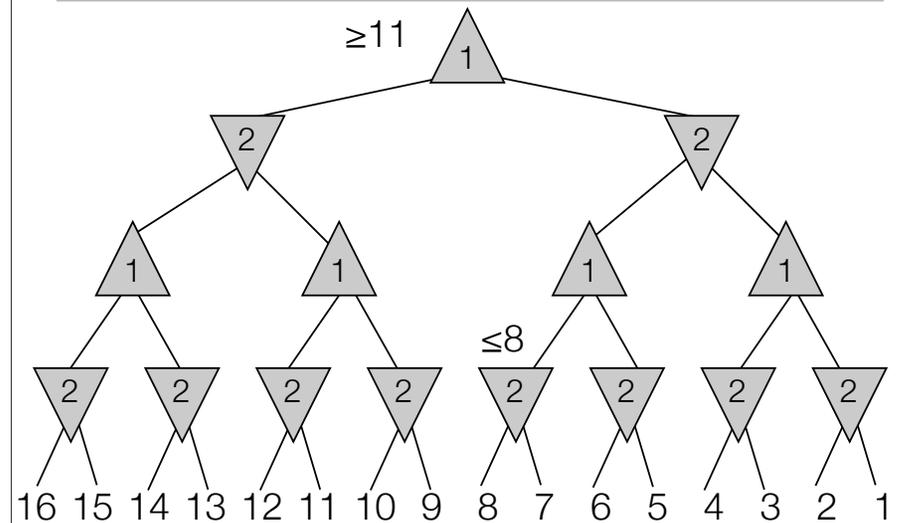
Alpha Beta Pruning

- alpha is the best score achieved by the max player
 - alpha starts at $-\infty$
- beta is the best score achieved by the min player
 - beta starts at ∞
- If $\alpha \geq \beta$, then we can perform a cutoff

Alpha Beta Pruning (shallow)



Alpha Beta Pruning (deep)

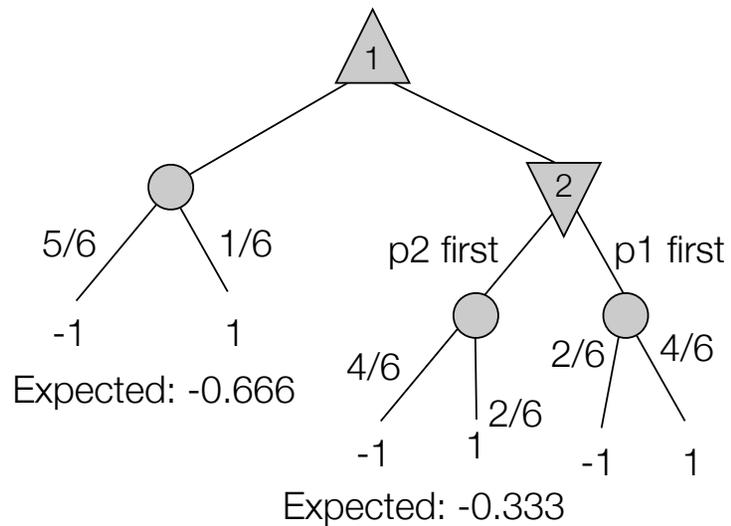


Alpha-beta pruning

- Assuming the game is a perfectly ordered win/loss tree
 - Easily show that alpha-beta expands the min nodes
- Node ordering matters!
 - b^d with the wrong ordering
 - $b^{d/2}$ with perfect ordering
 - $b^{3d/4}$ with “average ordering”
 - Will discuss move-ordering algorithms next week
- Is $b^{d/2}$ a big deal? Yes -- double our search depth

Expecti-Minimax

- What if we have chance nodes?
 - We can use minimax
 - We have to average over chance nodes
- Example:
 - Backgammon
 - Roll 2 die to determine possible moves
 - Simple choice of which Nim game to play



Pruning

- Can do alpha-beta style pruning, but more complicated
 - Need bounds on payoffs
 - Still have to do a lot of the computation

What depth should I search?

- Cannot know proper search depth *a priori*
- Need a method to dynamically choose search depth
 - Possibilities?
- Iterative deepening approach
 - Search all depths!

Null move

- If one player is in a strong position, they could skip their turn and still win
 - Perform “null” move
 - Search to depth 2/3 ply shallower than required
 - If the value of the game is still better on previous branches, it’s a win
 - Otherwise re-search with the full tree
- zugzwang -- sometimes it’s better not to move

Quiescence search

- Quiescence = quiet
 - Searching to a fixed depth may not be advantageous
 - eg if a capture has just been made, and the capture response hasn’t
 - Extend search until position is quiet
 - eg no captures and no check

Horizon Effect

- A result of limited depth knowledge
- Something bad is about to happen, but find a way to delay it until it happens after the search depth
- May turn a minor problem into a catastrophic one
 - Make a bad move now to avoid a worse state by the horizon effect
 - The worse state still happens later

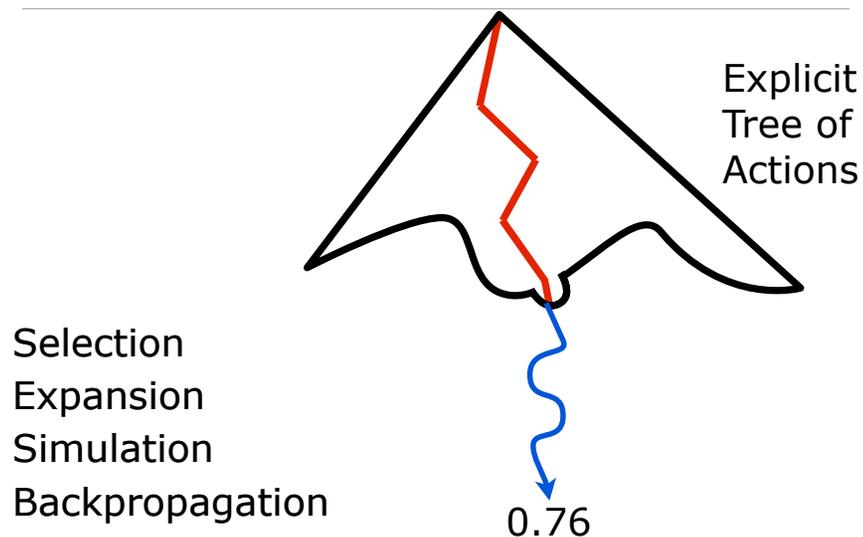
Transpositions

- When should we look for transpositions?
 - Near the top of the tree
 - Large savings
 - Likely to find transpositions
- When shouldn't we look for transpositions?
 - Near the bottom of the tree
 - Minimal savings
 - Unlikely to find transpositions

Transpositions

- What is needed to test for transpositions?
 - Naïve - list of states, and a linear search
 - Better - tree of states log(s) search
 - Best - hash table
- What hash function should we use?

Monte Carlo Tree Search



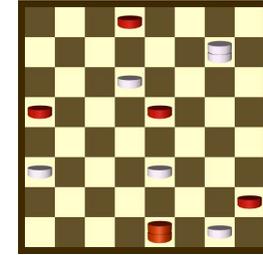
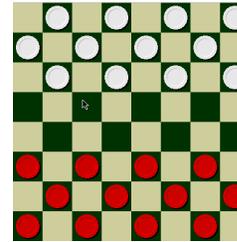
Today

- Finish up 2-player games
- Discuss homework
- Constraint Satisfaction Problems

Focus

- How did the computer AI win?
- How did the humans react?
- What mistakes were made in over-stating performance

Checkers



History

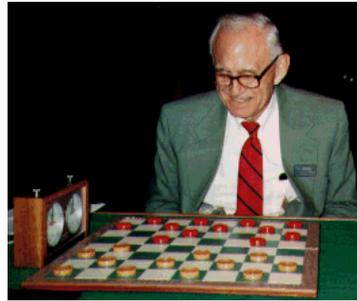
- Arthur Samuel began work on Checkers in the late 50's
 - Wrote a program that "learned" to play
 - Beat Robert Nealey in 1962
 - IBM advertised as "a former Connecticut checkers champion, and one of the nation's foremost players"
 - Nealey won rematch in 1963
 - Nealey didn't win Connecticut state championship until 1966
 - Crushed by human champions in 1966

Reports of success overblown

- "...it seems safe to predict that within ten years, checkers will be a completely decidable game." Richard Bellman, Proceedings of the National Academy of Science, 53(1965): p. 246.
- "So whereas computers can 'crunch' tick-tack-toe, and even checkers, by looking all the way to the end of the game, they cannot do this with chess." Lynn Steen, "Computer Chess: Mind vs. Machine," Science News, November 29, 1975.
- "Although computers had long since been unbeatable at such basic games as checkers...." Clark Whelton, Horizon, February 1978.
- "Computers became unbeatable in checkers several years ago." Thomas Hoover, "Intelligent Machines," Omni magazine, 1979, p. 162.
- "...an improved model of Samuel's checkers-playing computer today is virtually unbeatable, even defeating checkers champions foolhardy enough to 'challenge' it to a game." Richard Restak, The Brain: The Last Frontier, 1979, p. 336.
- "...the Duke program, Bierman believes, is already 'knocking at the door' of the world championship. Jensen and Truscott regard it as now being about the 10th strongest player in the world." Martin Gardner, Scientific American, January 1980, p. 25.

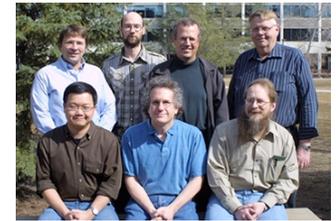
Human Champ: Marion Tinsley

- Closest thing to perfect human player
- Over 42 years loses only 3(!) games of checkers.



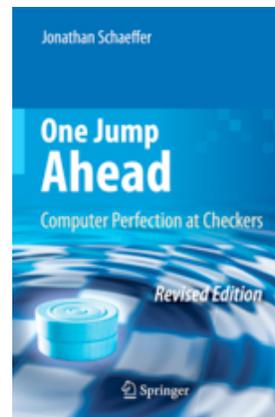
Computer Challenger: Chinook

- Have to overcome the stigma of checkers being “solved” in 1963.
- Project takes five years, 10 people, > 200 computers working around the clock, and terabytes of data.



Outcome

- The first computer to win a human world championship (1994)
- Checkers is solved (2007)!
- Perfect play leads to a draw
- Humans will never win against Chinook again

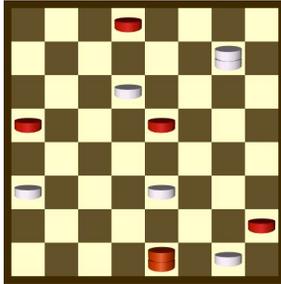


Secret: Endgame Databases

- Endgame databases
 - Searched all positions with 10 or fewer pieces
 - Each identified with perfect win, loss, draw info
 - 39 trillion positions in the program’s memory
 - Exceeds human abilities
 - Introduces perfect knowledge into the search
 - Factual knowledge, but without the ability to generalize it

The 100(?) -year position

The 100-Year Position (white to move)
Give it to humans for 100 years... win!

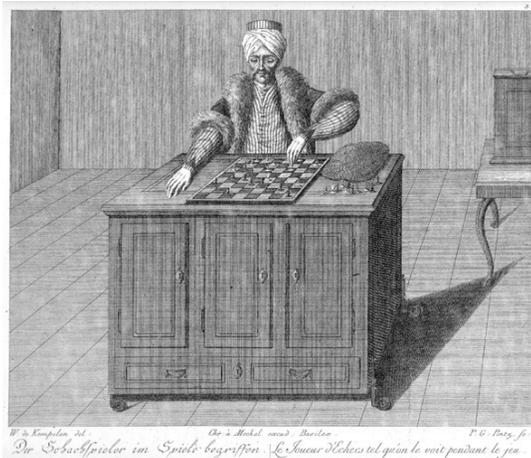


Give it to Chinook for one I/O... draw!
The 197-Year Position

Chess

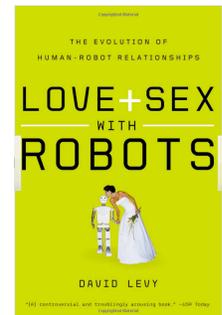


1770 - The Turk



Further Work

- 1910 - El Ajedrecista plays King+Rook vs. King endgames
- 1950's - Claude Shannon, Alan Turing, John McCarthy begin work on Chess
- 1968, David Levy bets that no computer program would win a chess match against him within 10 years
 - Wins his bet 10 years later



Human Champ: Garry Kasparov

- Holds the record for the longest time as the #1 rated player (1986-2005)
- Reached a 2851 Elo rating, the highest rating ever achieved
 - Beaten by Magnus Carlsen, 2881 Elo rating as of March 2014



Computer Challenger: Deep Blue

- 2,400 lbs
- 512 processors
- 200,000,000 pos/sec



The result of second match in 1997

- Kasparov won game 1
- Kasparov lost game 2
- Kasparov self-destructed in game 6 and lost the match
- In the video he rails on about game 2. He was crushed in the game but in the final position there is a miracle that saves the game. No one saw it at the time, and certainly not Kasparov, who resigned.
- Note that Deep Blue lost game 1 in a drawn position due to a bug.

Kasparov's Response



- Who is better?

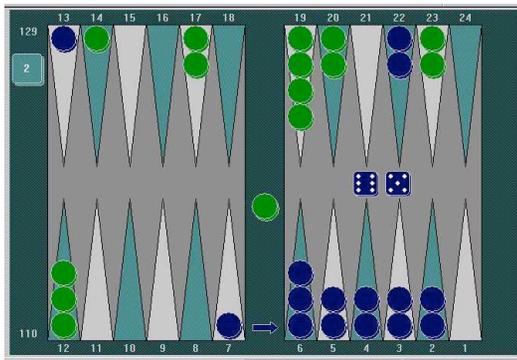
Post-analysis

- Exhibition match; scientific data point can't be repeated.
- Man was superior in 1997 but by 2006 it appears that man is no longer competitive
- Deep Fritz played world chess champion Vladimir Kramnik in November 2006
- Used a personal computer containing two Intel Core 2 Duo CPUs, capable of evaluating only 8 million positions per second
- Searched to an average depth of 17 to 18 plies

Secret: Brute-Force

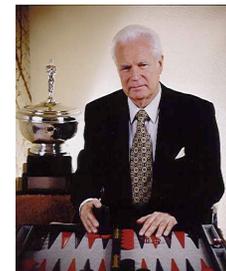
- Brute-force search
 - Consider all moves as deeply as possible
 - Some moves can be provably eliminated
 - 200,000,000 per second versus Kasparov's ~2
 - 99.99% of the positions examined are silly by human standards
- Lots of search... and little knowledge
- Tour de force for engineering

Backgammon



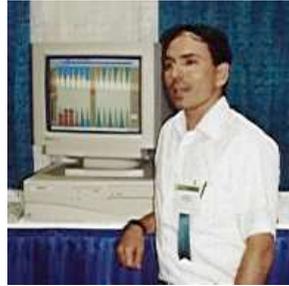
Human Champ: Malcolm Davis

- World backgammon champion.
- Agrees to play exhibition matches against a computer; narrowly avoids becoming part of computing history.



Computer Challenger: TDGammon

- Gerry Tesauro builds TDGammon over 8 years. Learned to play strong backgammon
- Unable to beat champion in match; too many games needed for statistical significance



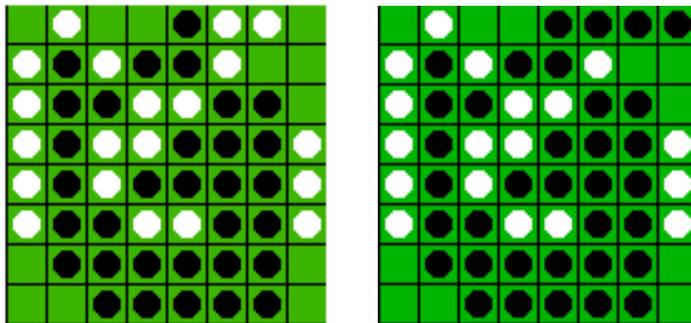
Nathan Sturtevant

Secret: TD-Learning

- Pioneering success for temporal difference learning
- Backgammon happens to be very well suited for these techniques
- Combination of search, expert knowledge, and a neural net tuned using TD learning
- Tour de force for artificial intelligence

Nathan Sturtevant

Othello (Reversi)



Human Champ: Takeshi Murakami

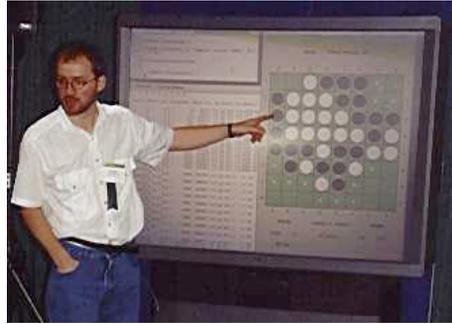
- World Othello Champion



Nathan Sturtevant

Computer Challenger: Logistello

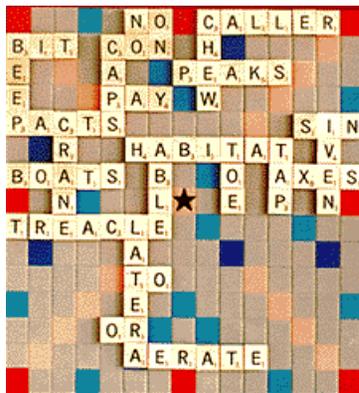
- Had to overcome the stigma of Othello being “solved” in 1980 and 1990.
- Michael Buro’s one-man effort for five years produces Logistello.
- 6 game match
 - Aug. 4-7, 1997
- Logistello wins 6-0



Secret: Machine Learning

- Automatically discovered and tuned knowledge
 - Samples patterns to see if its presence in a position can be correlated with success
- Tuned 1.5 million parameters using self-play games with feedback
- “Knowledgeable” program but no one understands the knowledge

Scrabble



Human Champion: Adam Logan

- Math professor.
- 1997 Canadian and North American scrabble champion

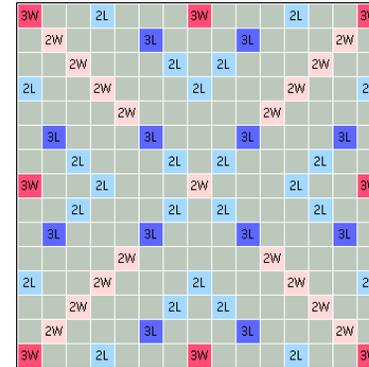


Computer Challenger: Maven



Brian Sheppard spends 14 years developing his Scrabble program.

Maven versus Logan: A Classic



Brian Sheppard's commentary:

- The following game is in the author's opinion the best Scrabble board game ever played in a tournament or match.
- The game is the 12th game in the AAAI-98 exhibition match between MAVEN and Adam Logan.
- After losing three of the first four games, MAVEN had come back strongly to take a 7 to 4 lead.
- In total, there were 14 games scheduled.
 - *First player to win 8 games wins the match.*

Scrabble

Maven v. Logan

M	O	U	T	H	A	R	T		2L		3W		
A	E				3L		Q				2W		
T		2W			2L	2L	U		G				
H	U	R	T		2L	A		2W	R		2L		
N	E	O	N			I	S	E		L			
	3L	D	O	Z	Y		3L	P	A	X	E		
		E		E	2L	J	A	W	S		I		
I	A	M	B	C	A	V	Y	N	2L	E	3W		
W	E		R		2L	K			2L				
	3L	N	F	3L	L		B			3L			
		D	E	O		O	R						
2L	D	E	V	I	A	N	C	E	S		2L		
		2W	D	G	G	O	2L			2W			
		2W		N	3L	F		3L			2W		
P	I	L	I	S		T	U	T	O	R	I	A	L

MAVEN: 440
last word: mouthpart
worth : 100

LOGAN: 438
last word: quai
worth : 35

E | I | O | I

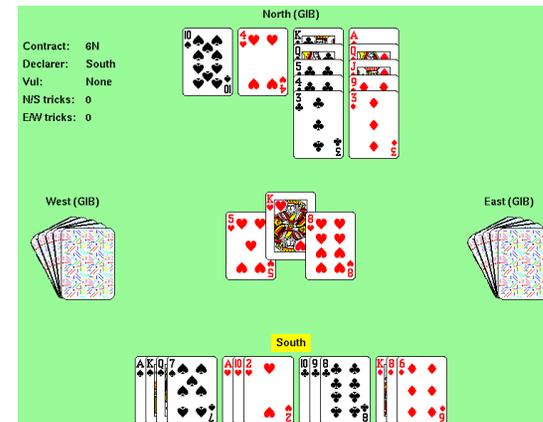
Definitions

MOUTHPART
part of the mouth of an insect or other arthropod

The Secret?

- Memory
 - Maven has the entire dictionary in its memory
 - over 100,000 words
- Simulations
 - Simulates 1,000 game scenarios per decision
 - Typically 700 legal moves (more with a blank)!

Bridge



Human Champ: Zia Mahmood

- In 1990 offers £1,000,000 bet that no program can defeat him.
- December 1, 1996
 - Cancels bet when faced with a possible challenger.



Computer Challenger: GIB

- Matt Ginsberg develops the first expert-level bridge program, GIB (1998).



- Finishes 12th in the World Championship.

The Verdict...

- Man is better than machine!
- Likely to remain that way for a while yet
 - Difficulties in understanding the bidding

The Secret?

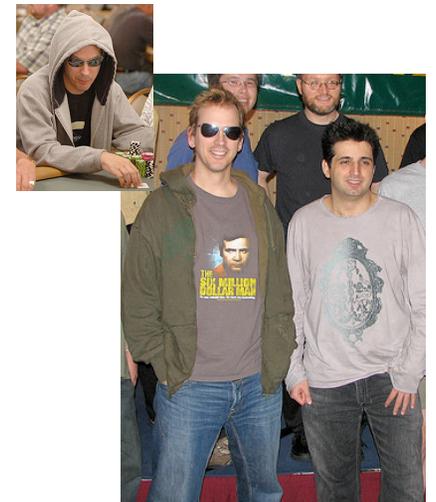
- GIB does 100 simulations for each decision
 - Deals cards to opponents consistent with available information
 - Chooses the action that leads to the highest expected return
 - Program does not understand things like “finesse” or “squeeze”
 - Simulations contain implicit knowledge

Poker



Human Champion: Phil Laak

- Phil Laak (aka the unibomber) holds a World Poker Title
 - Stronger at no-limit texas hold'em
- Ali Eslami was invited by Phil to play against University of Alberta computers



Computer Challenger: Polaris

- Poker is a hard problem because of multiple opponents, imperfect information, and deception
- Ongoing project at the UofA (~20 years)



from Wired Magazine

The result (part 1)

- 2007 first man-machine match
- Narrow loss for UofA programs



The result (part 2)

- 2008, second match
 - Played against a team of 2-player experts
 - Polaris wins



Matt Hawrilenko



IJay Palansky

The Secret?

- Precise probability calculations
- Game theoretic solutions
- Use short-term and long-term statistics to model each opponent

- **Not playing most popular form of game**

Go



Human Champion: Zhou Junxun

- Ranked 9-dan (professional)
- Winner of 43 domestic and 2 international titles



Computer Challenger: Fuego

- Written by Markus Enzenberger and Martin Müller
 - Both had strong Go programs
 - Teamed up to write stronger program



Result

- Fuego was the first computer program to win an official game of 9x9 Go against a 9-Dan professional player in 2009
 - Thought to be impossible 10 years ago
 - Not yet playing at this strength 19x19 board
 - Collaborating with IBM and Gerry Tesaro

The secret?

- Monte-Carlo Tree Search
 - Use heuristic to choose good *actions*
 - Play out millions of games guessing the best actions for each player
 - Working with IBM on massively parallel hardware to improve performance

Jeopardy



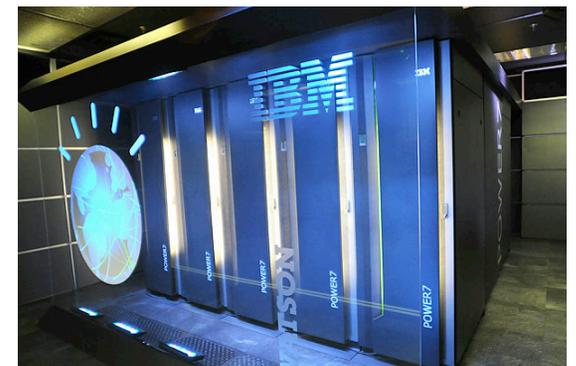
Human Champion: Ken Jennings

- Won 74 games straight
 - Lost the 75th game
 - Won a total of \$2,522,700
- Has won \$3.8 million on game shows



Computer Challenger: Watson

- 2880 POWER7 processor cores
- 16 Terabytes of RAM
- ~\$3 million
- Stored copy of wikipedia in memory



Result

- Watson scores \$77,147
- Jennings scores \$24,000
- Rutter scores \$21,600

- Watson can buzz in faster and more accurately than humans
- Watson still misses many basic questions

Response to Watson's win



The Secret?

- Massive hardware optimization reduced days of computation to a few seconds
- Many tuned experts able to answer particular question types
- A high-level controller which weights experts

- The ability to 'learn' from answers in a category
- Some similarities to PROVERB program which solves crossword puzzles