Artificial Intelligence

Games

Daniel Polani

Games

More Precisely:

- two-person (not multi-person; no gang-ups)
- perfect information (no card games)
- deterministic (no backgammon)
- alternating moves (no rock/scissors/paper)
- zero-sum (no prisoner's dilemma)

games

Game Structure

Conditions: game is over when *terminal* position reached where game ends (no successor moves).

Possible Outcomes: consider *win/loss/draw*. Other, intermediate outcomes also possible.

Games and AND/OR-trees (Recap)

Game	AND/OR
game position	problem
terminal won position	goal node, trivially solved
terminal lost	unsolvable
non-terminal won	solved problem
us-to-move (player A)	OR node
them-to-move (player B)	AND node

Position Utilities

Motivation: since, in general, game trees are too big to be completely solved, use a *utility* (value) function to indicate which positions are more promising than another.

Implication: quality of a game state characterized by its value (utility) U, a real-valued number

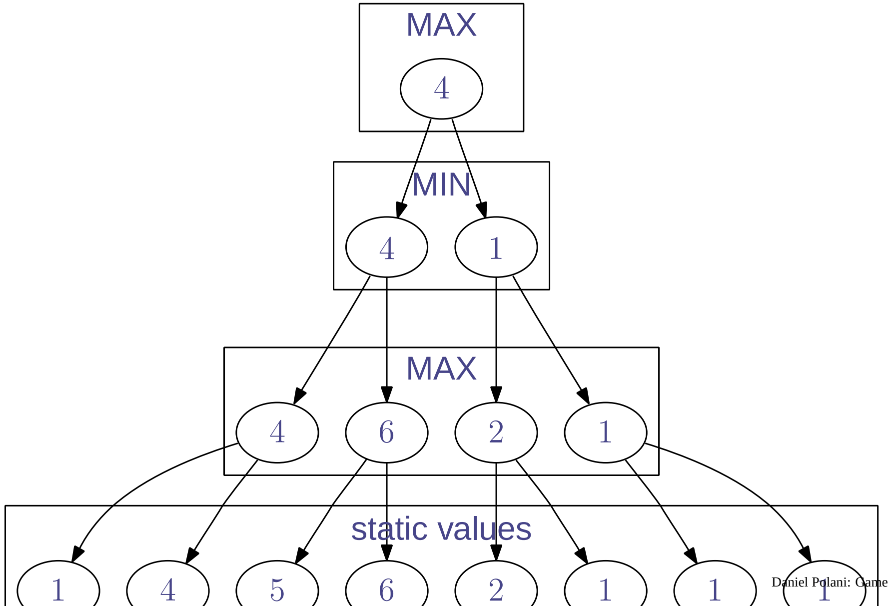
Note: “promising” subtrees are indicated by a high value of U for starting states.

Position Utilities II

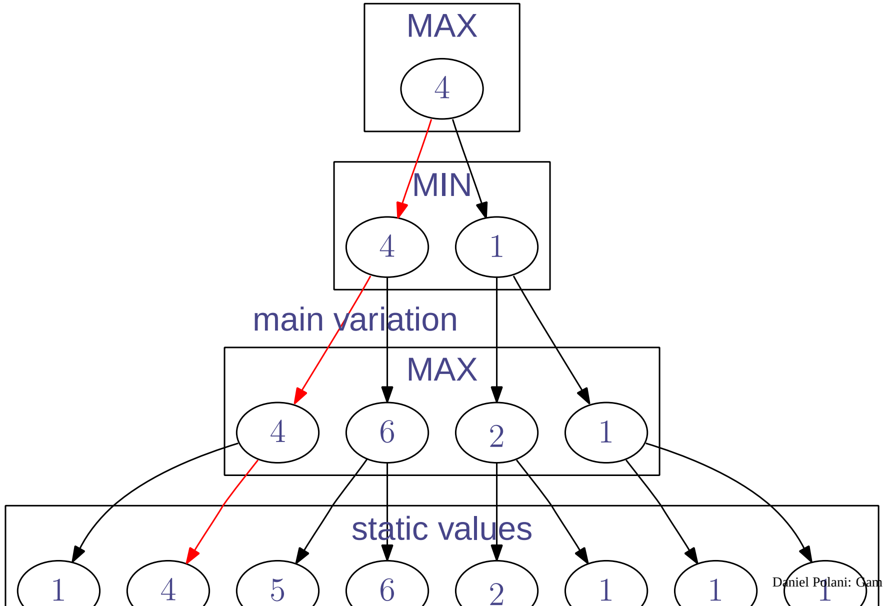
Note: the *true* value U of a position indicates the state of the position won/lost/draw, e.g.

1. $U = 100$: current position allows player A to win (on optimal game from both sides)
2. $U = -100$: current position is lost for player A (on optimal game from both sides)
3. $U = 0$: position is a draw (no player can force a win)

Minimax Principle



Minimax Principle (Main Variation)



Minimax view of utilities

Consider: $U(P)$, the utility of a position

Let: $S(P) = \{P_1, P_2, \dots, P_n\}$ be the set of successors for position P

Minimax Utility: define

$$U(P) = \begin{cases} U_{\text{static}}(P) & \text{if } P \text{ terminal, i.e. } S(P) = \{\} \\ \max\limits_{P_i \in S(P)} U(P_i) & \text{if } P \text{ is a MAX-to-move position} \\ \min\limits_{P_i \in S(P)} U(P_i) & \text{if } P \text{ is a MIN-to-move position} \end{cases}$$

The Alpha-Beta Algorithm

Observation:

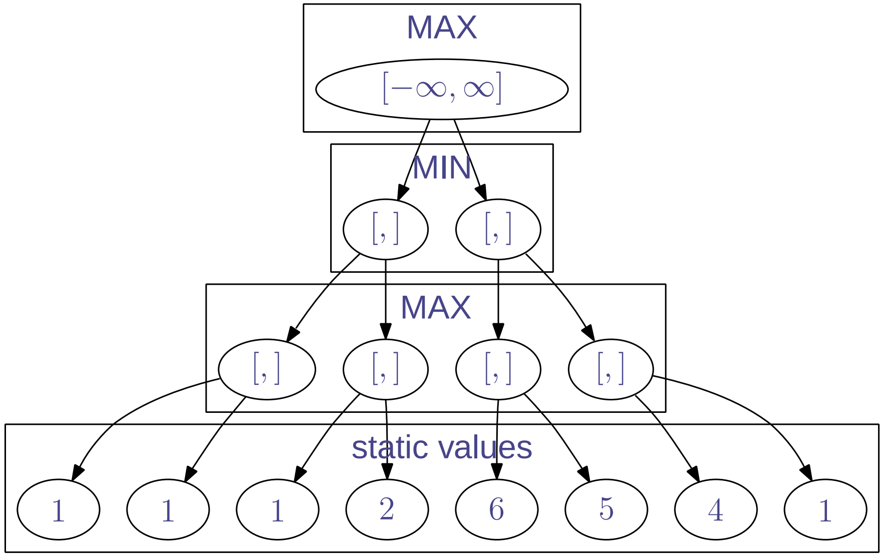
- sometimes we know a move is not good and will never be covered
- in that case, the exact utility of the node is not needed

α-β principle:

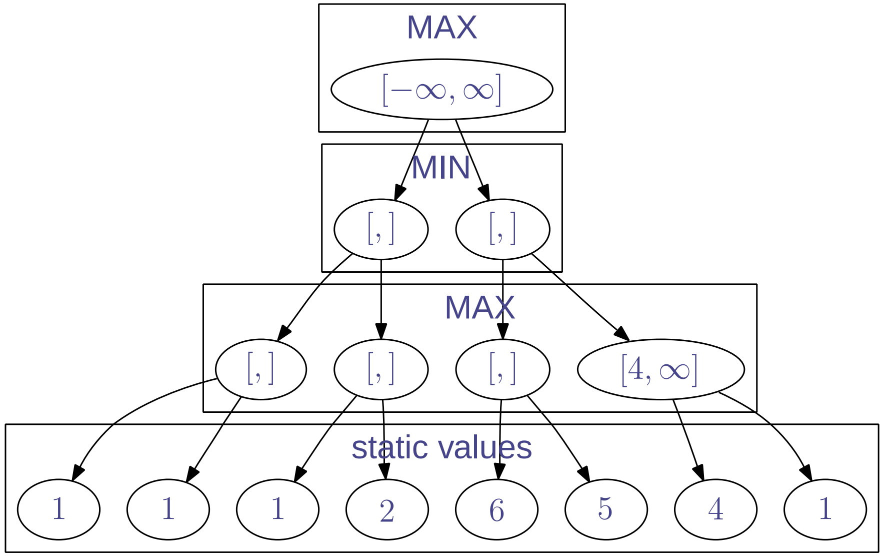
- search for the utility of a position but only if in the interval $[\alpha, \beta]$
- if it is outside, its exact value is not important, we will be prevented from taking that path anyway

Illustration: see following slides

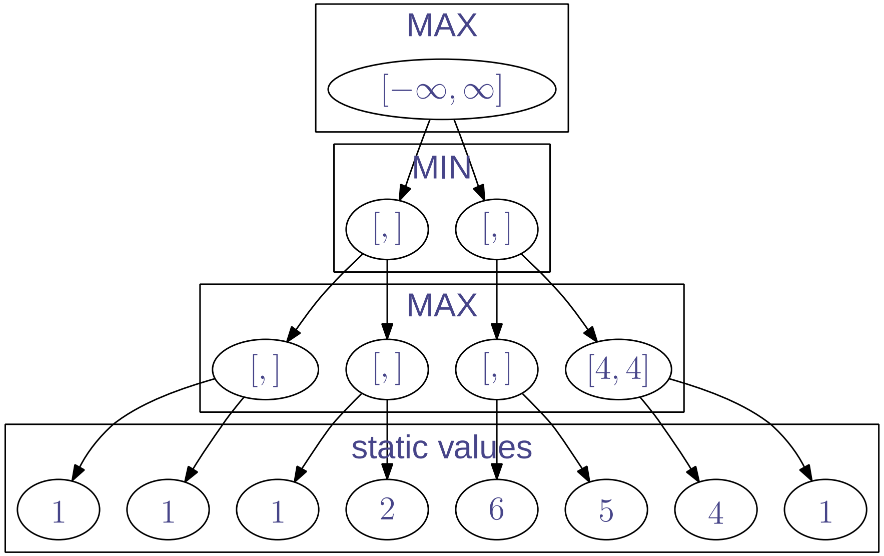
The Alpha-Beta Algorithm



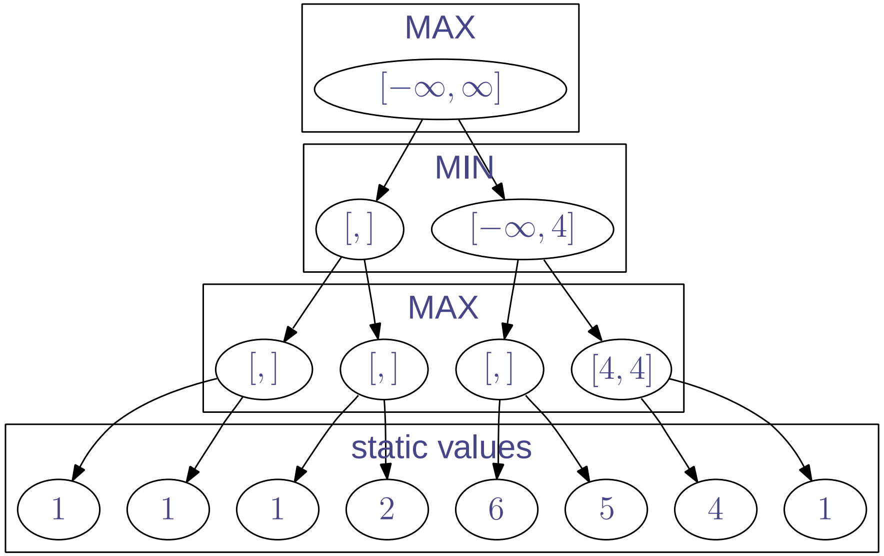
The Alpha-Beta Algorithm



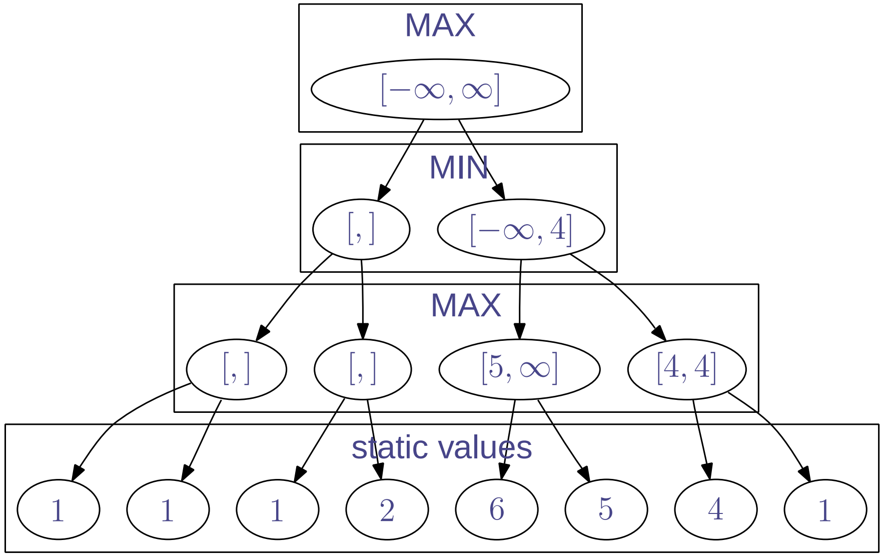
The Alpha-Beta Algorithm



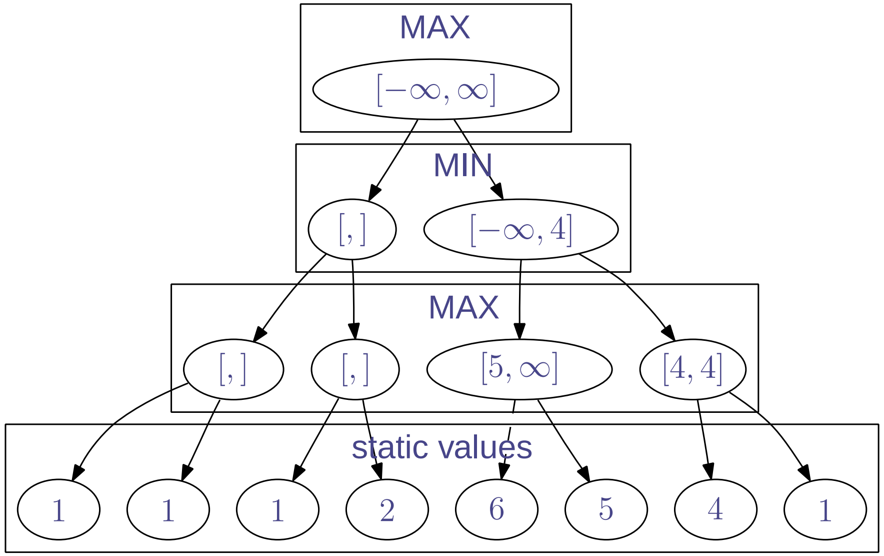
The Alpha-Beta Algorithm



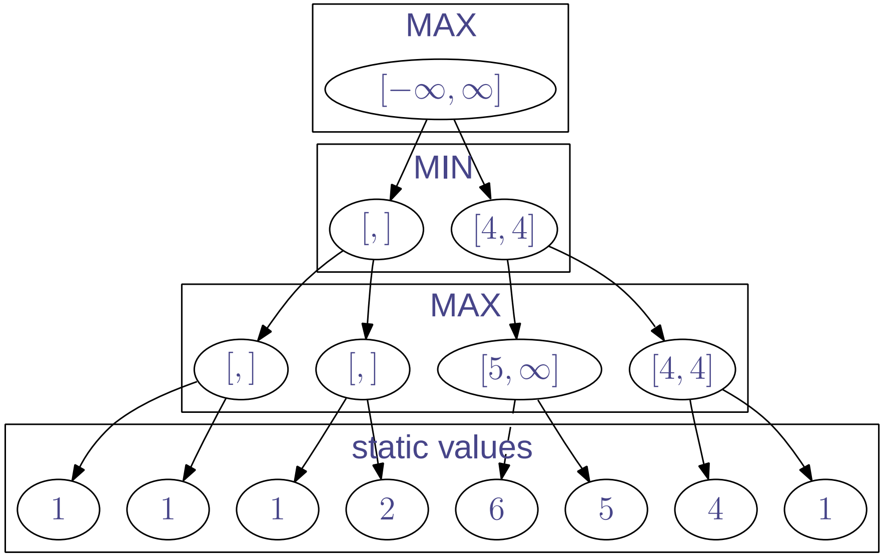
The Alpha-Beta Algorithm



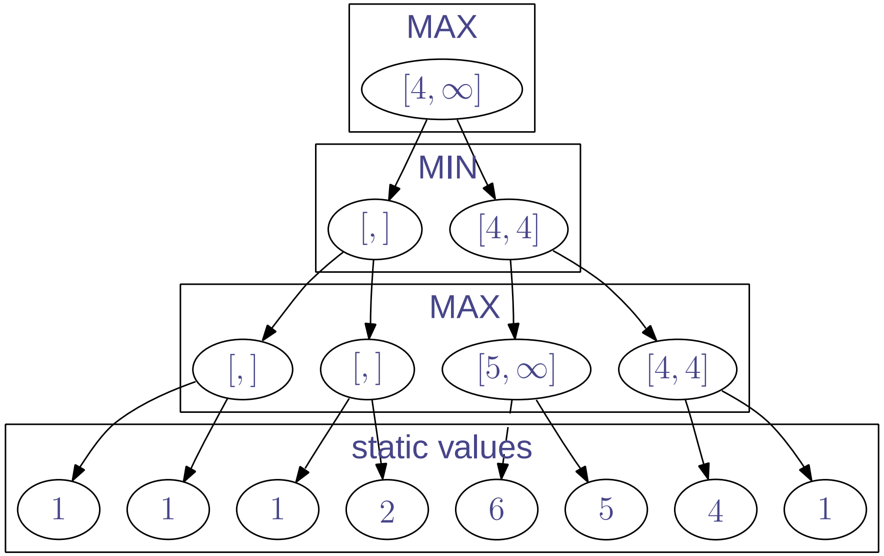
The Alpha-Beta Algorithm



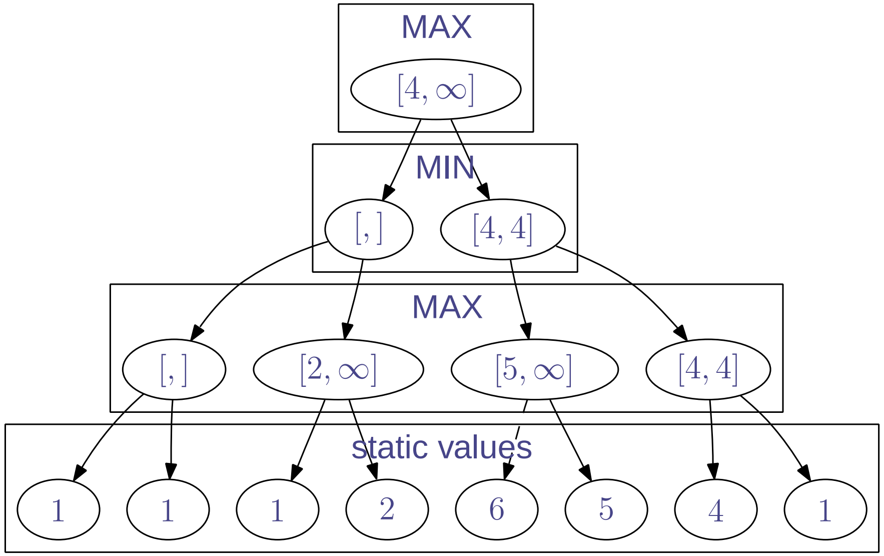
The Alpha-Beta Algorithm



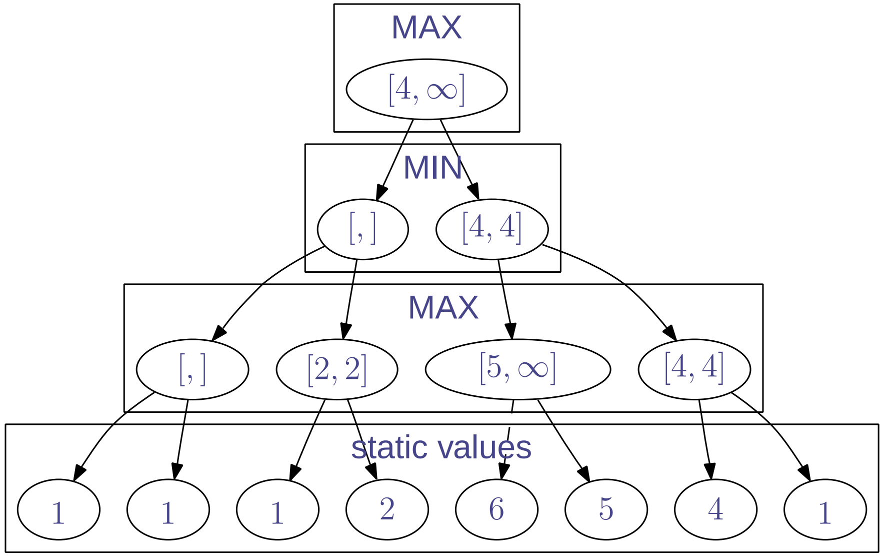
The Alpha-Beta Algorithm



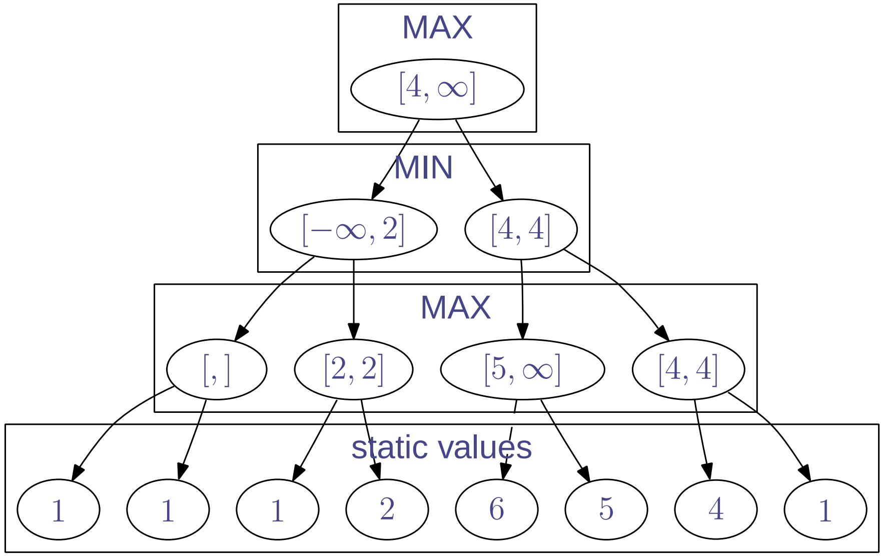
The Alpha-Beta Algorithm



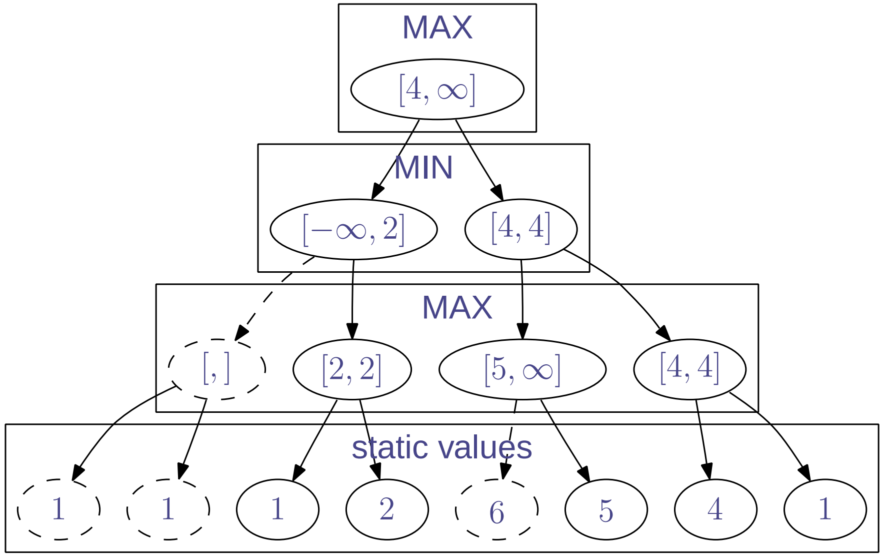
The Alpha-Beta Algorithm



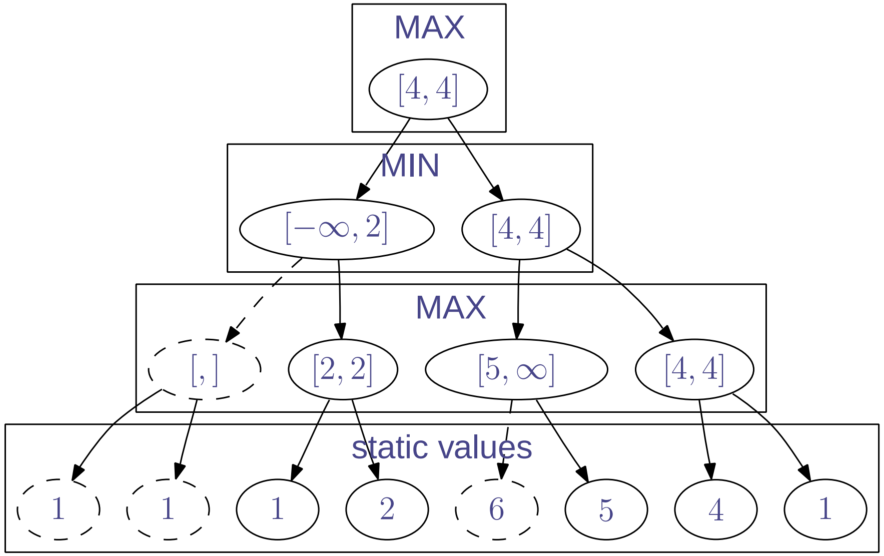
The Alpha-Beta Algorithm



The Alpha-Beta Algorithm



The Alpha-Beta Algorithm



Alpha-Beta Algorithm: Properties

α: worst guaranteed utilty for MAX (and best achievable value for MIN)

β: worst guaranteed utility for MIN (and best achievable value for MAX)

Good Enough Utility: a utility $U(P, \alpha, \beta)$ is a utility such that

$$\begin{aligned} U(P, \alpha, \beta) &< \alpha && \text{if } U(P) < \alpha \\ U(P, \alpha, \beta) &= U(P) && \text{if } \alpha \leq U(P) \leq \beta \\ U(P, \alpha, \beta) &> \beta && \text{if } U(P) > \beta\,. \end{aligned}$$

In Particular: $U(P, -\infty, \infty) = U(P)$

Remark: in the best case, this reduces the search branching factor from b for minimax to $\sqrt{b}$

Thus: can search twice as deeply as with minimax with the same evaluation effort

Further Improvements

- limitation of move selection
- heuristic value function (cutoff before final state)
- quiescence heuristics

Other Game Types

Prisoner's Dilemma: game with

1. simultaneous moves
2. non-zero-sum payoff

${}_{P1}\backslash^{P2}$	defect	cooperate
defect	${}_{-6}\backslash^{-6}$	${}_{0}\backslash^{-10}$
cooperate	${}_{-10}\backslash^{0}$	${}_{-1}\backslash^{-1}$

Dominance

Def. (strong dominance): a strategy s for a player p *strongly dominates* s' if the payoff using s is better than using s' for every *fixed* choice of strategy for other players.

Def. (weak dominance): a strategy *weakly dominates* if it is better on (at least) one strategy of other players and no worse on any other.

Def.: A *dominant strategy* dominates all others.

Pareto optimality/dominance

Def. (Pareto optimality): an outcome is *Pareto optimal* if no other outcome would be preferred by *all* the players.

Def. (Pareto dominance): an outcome is strongly *Pareto dominated* if all players would prefer some other outcome

Def. (weak Pareto dominance): an outcome is weakly Pareto dominated, if some players would prefer another outcome to which all others would not mind switching

Dominance in Prisoner’s Dilemma

Note: both Alice and Bob have a dominant strategy, i.e. we have a dominant strategy *equilibrium*

Def. (Nash equilibrium): a selection of strategies for each player such that no player can benefit by switching his/her strategy if all other players’ strategies are unchanged.

Remark: the *dilemma* in the prisoner’s dilemma is due to the fact that the Nash equilibrium $(-6, -6)$ of both prisoners defecting is Pareto dominated by $(-1, -1)$ of both prisoners cooperating.

Note: a Nash equilibrium can arise even without the existence of a dominant strategy.

Comments on Game Theory

Remark: if

- the prisoner’s dilemma game is being iterated
- the players are allowed to have memories and identify their opponent

this can lead to solutions which avoid the equilibrium.

Note: Tit-For-Tat and very related strategies prove to be remarkably stable and robust solutions.

Remark: if one has a Pareto-optimal point which is also a Nash equilibrium, then we call that a *solution* of the game.

Back to Zero-Sum Games

Consider: simultaneous zero-sum games. Need to consider only the payoff P for one of the players, the other will follow as $-P$.

2-Finger Morra: payoff matrix:

$_{E}\backslash^{O}$	1	2
1	$_{2}\backslash^{-2}$	$_{-3}\backslash^{3}$
2	$_{-3}\backslash^{3}$	$_{4}\backslash^{-4}$

Goal: find *solution*

Zero-Sum Games: Solution

Scenario 1: force E to begin, O to follow. This is an advantage for O. It is easy to see that E is guaranteed an outcome of $U_E \geq -3$.

Scenario 2: force O to begin, E to follow. O can ensure an outcome with $U_E \leq 2$.

Mixed Strategy

Note: revealing a strategy gives the second player an advantage. For, if second player plays $[p:1;(1-p):2]$ (notation: lottery where outcome 1 is selected with probability p and outcome 2 is selected with probability $1-p$), the expected utility for E is

$$pU_E(O=1)+(1-p)U_E(O=2)$$

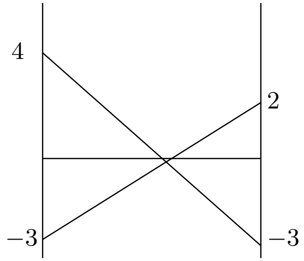
If $U_E(O=1)$ and $U_E(O=2)$ are different, O should pick the best as *pure* strategy.

Utilities for Mixed Strategies I

Assume: E moves first, without O knowing the move, but knowing p in the strategy $[p:1;(1-p):2]$ of E. Then if

1. O chooses 1, then $\mathbf{E}(U)=2p-3(1-p)=5p-3$
2. O chooses 2, then $\mathbf{E}(U)=-3p+4(1-p)=4-7p$

O will always pick the minimum of both. E will pick p such that is maximal, $U=-\frac{1}{12}$.



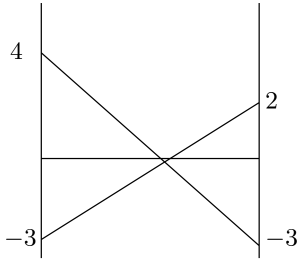
Utilities for Mixed Strategies II

Assume: O moves first, probabilites $[q:1;(1-q):2]$. If

1. E picks 1, then $\mathbf{E}(U)=2q-3(1-q)=5q-3$
2. E picks 2, then $\mathbf{E}(U)=-3q+4(1-q)=4-7q$

E picks the maximum of both. O will pick q such that is minimal, $U=-\frac{1}{12}$.

Note: The two U values enclose the true value, which is therefore $U=-\frac{1}{12}$. It turns out that $p=\frac{7}{12}=q$.



Minimax Equilibria

Bottom Line: there exists an *equilibrium*, a *minimax* equilibrium which is Nash equilibrium.

von Neumann: every two-player zero-sum game has a minimax equilibrium on mixed strategies. Also, in zero-sum games, Nash equilibria are minimax equilibria.