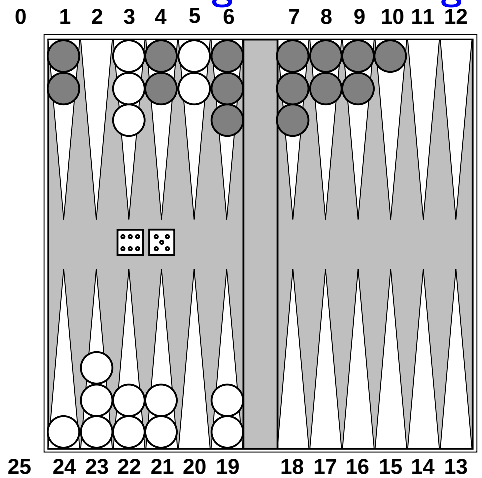
Today

See Russell and Norvig, chapter 6

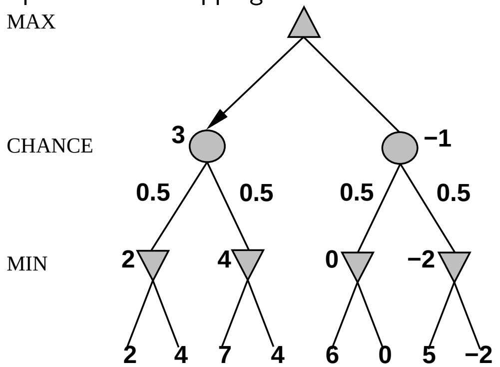
- Game playing
- Nondeterministic games
- Games with imperfect information

Nondeterministic games: backgammon



Nondeterministic games in general

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



Algorithm for nondeterministic games

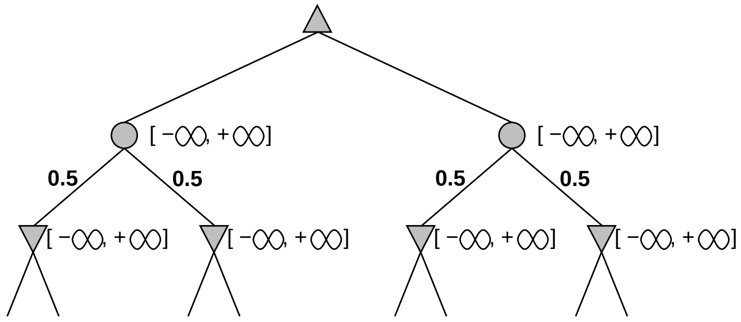
EXPECTIMINIMAX gives perfect play

Just like MINIMAX, except we must also handle chance nodes:

. . .
if *state* is a MAX node **then**
  return the highest EXPECTIMINIMAX-VALUE of SUCCESSORS(*state*)
if *state* is a MIN node **then**
  return the lowest EXPECTIMINIMAX-VALUE of SUCCESSORS(*state*)
if *state* is a chance node **then**
  return average of EXPECTIMINIMAX-VALUE of SUCCESSORS(*state*)
. . .

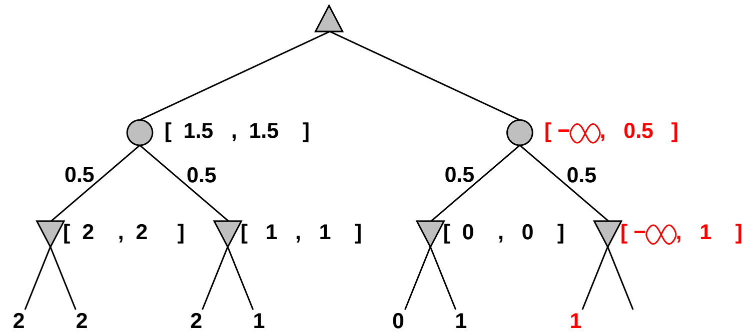
Pruning in nondeterministic game trees

A version of α-β pruning is possible:



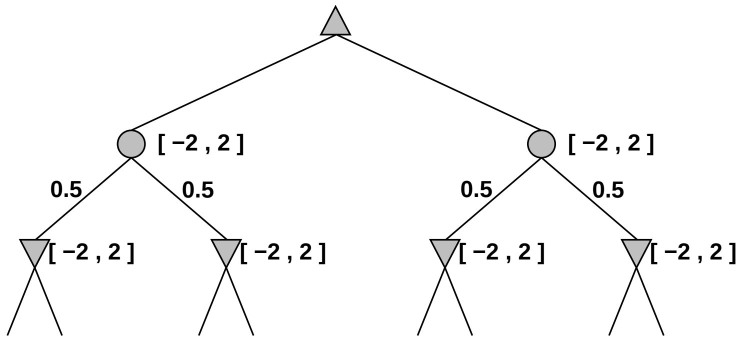
Pruning in nondeterministic game trees

A version of α-β pruning is possible:



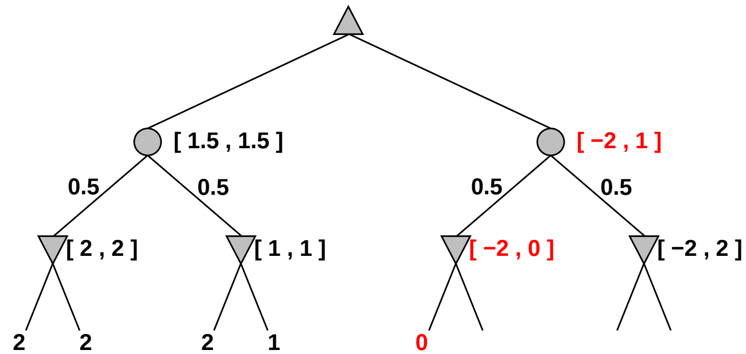
Pruning contd.

More pruning occurs if we can bound the leaf values



Pruning contd.

More pruning occurs if we can bound the leaf values



Nondeterministic games in practice

Dice rolls increase b: 21 possible rolls with 2 dice
Backgammon $\approx$ 20 legal moves (can be 6,000 with 1-1 roll)

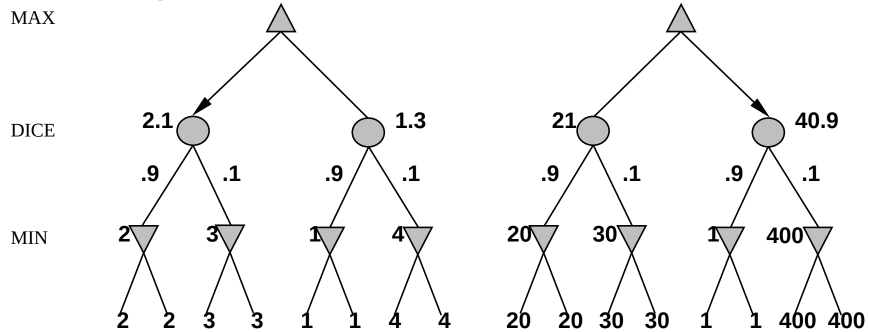
$$\text{depth } 4 = 20 \times (21 \times 20)^3 \approx 1.2 \times 10^9$$

As depth increases, probability of reaching a given node shrinks
$\Rightarrow$ value of lookahead is diminished

α–β pruning is much less effective

TDGAMMON uses depth-2 search + very good EVAL
$\approx$ world-champion level

Digression: Exact values DO matter



Behaviour is preserved only by *positive linear* transformation of EVAL

Hence EVAL should be proportional to the expected payoff

Games of imperfect information

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

Typically we can calculate a probability for each possible deal

Seems just like having one big dice roll at the beginning of the game

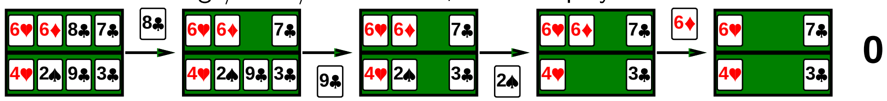
Idea: compute the minimax value of each action in each deal,
then choose the action with highest expected value over all deals

Special case: if an action is optimal for all deals, it's optimal.

GIB, current best bridge program, approximates this idea by
1) generating 100 deals consistent with bidding information
2) picking the action that wins most tricks on average

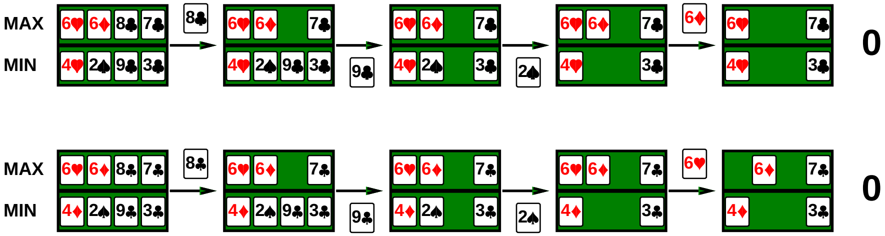
Example

Four-card bridge/whist/hearts hand, MAX to play first



Example

Four-card bridge/whist/hearts hand, MAX to play first



Example

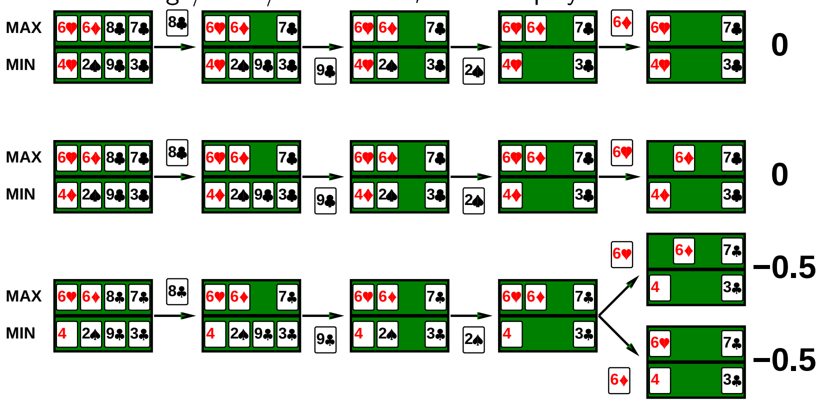
So far, we have seen the **optimal** play from Max in two different situations.

Now suppose that Max knows that Min has one or other of the two hands, but does not know which one.

Is the same play still optimal?

Example

Four-card bridge/whist/hearts hand, MAX to play first



Commonsense example

Road A leads to a small heap of gold pieces
Road B leads to a fork:
  take the left fork and you'll find a mound of jewels;
  take the right fork and you'll be run over by a bus.

Commonsense example

Road A leads to a small heap of gold pieces
Road B leads to a fork:
 take the left fork and you'll find a mound of jewels;
 take the right fork and you'll be run over by a bus.

Road A leads to a small heap of gold pieces
Road B leads to a fork:
 take the left fork and you'll be run over by a bus;
 take the right fork and you'll find a mound of jewels.

Commonsense example

Road A leads to a small heap of gold pieces
Road B leads to a fork:
 take the left fork and you'll find a mound of jewels;
 take the right fork and you'll be run over by a bus.

Road A leads to a small heap of gold pieces
Road B leads to a fork:
 take the left fork and you'll be run over by a bus;
 take the right fork and you'll find a mound of jewels.

Road A leads to a small heap of gold pieces
Road B leads to a fork:
 guess correctly and you'll find a mound of jewels;
 guess incorrectly and you'll be run over by a bus.

Proper analysis

* Intuition that the value of an action is the average of its values in all actual states is WRONG

With partial observability, value of an action depends on the information state or belief state the agent is in

Can generate and search a tree of information states

Leads to rational behaviors such as
- ♢ Acting to obtain information
- ♢ Signalling to one's partner
- ♢ Acting randomly to minimize information disclosure

Summary

Games are fun to work on! (and dangerous)

They illustrate several important points about AI

- ♢ perfection is unattainable $\Rightarrow$ must approximate
- ♢ good idea to think about what to think about
- ♢ uncertainty constrains the assignment of values to states

Games are a good field to experiment with AI techniques and develop new approaches.