Mod 2&3 search + game

search
- Uninformed — systematic strategies
- Informed — domain knowledge to narrow search scope
- Game — minimax, state pruning; probabilistic games

Uninformed Search Algos

> General Search
pop from list and expand

> BFS
add all children to list
$1+b+b^2+\dots+b^d$ time
$O(b^{d+1})$ d=depth of goal — space
complete

> DFS
add children to front of list
$O(b^m)$ m=max depth — time
$O(bm)$ — space
complete on finite
suboptimal

> DLS
DFS to a defined depth
$O(b^l)$ — time
$O(bl)$ — space
not complete
suboptimal

> Iterative Deepening
DLS on increasing depths
1: 1,2,3 : 1,2,3,4,5,6,7 : 1,2,3...
$O(b^d)$ — time
$O(bd)$ — space
complete
optimal

> BiDirectional BDS
BFS from start and goal
$O(b^{d/2})$

Mod 4 Informed search + Iterative Improvement

Informed -> evaluation function

best first search - order nodes by f
Greedy bestfs - go best
space & time $O(b^m)$
incomplete
sub optimal

$A^* = g(n) + h(n)$ } best first heuristic makes A*
- $g(n)$: path so far
- $h(n)$: estimated distance to goal
- $h(n) \le h^*(n)$

complete
optimal
↓space

IDA* - ID but depth = $g(n)+h(n)$

Mod 5 Iterative Improvement

Simulated Annealing

$p(\Delta E) = e^{-\Delta E / kT}$

p - of taking an upward step based on ΔE from state to state and a decreasing T

Constraint Satisfaction
- forward checking
- backtracking
- heuristics
  - variable selection
    - most constrained — few possible values
    - most constraining — involved in most rules
  - value assignment
    - least constraining — reduce b or leave future freedom

Mod 6 Game Playing

	deterministic	chance
perfect info	chess, checkers, Go, battleship	backgammon, monopoly, bridge, poker, scrabble
imperfect info		

minimax

```
max(s, α, β):   (α: -∞ max, β: +∞ min)
    for child in s:
        v = max(v, min(s, α, β))
        if v ≥ β: return v
        α = max(α, v)
    return v
```

```
min(s, α, β):   (α: max, β: min)
    for child in s:
        v = min(v, max(s, α, β))
        if v ≤ α: return v
        β = min(β, v)
    return v
```

expectiminmax $v = \sum x p(x)$

Mod 8 Prop Logic

- commutative (order)
- associative (ignore '()' on like v or ^)
- identity ($A \vee F = A$, $A \wedge T = A$)
- universal bound ($\vee T = T$, $\wedge F = F$)
- negation ($A \vee \neg A = T$, $B \wedge \neg B = F$)
- double neg ($\neg\neg A = A$)
- DeMorgan (distribute ¬, flip ∨/∧)
- equivalence ($A \leftrightarrow B = A \to B \wedge B \to A$)
- implication ($A \to B = \neg A \vee B$)
- idempotent ($A \wedge A = A$)

Modus Ponens — $(A \to B \wedge A \to B)$
Unit Resolution — $(A \vee B \wedge \neg B \to A)$
Resolution — $(A \vee B \wedge \neg B \vee C \to A \vee C,\ \neg A \to B \wedge B \to C \to \neg A \to C)$

CNF — C1 ∧ C2 ∧ C3
DNF — T1 ∨ T2 ∨ T3
Horn - ≤1 pos literal

1. equiv & implication
2. double neg & DeM
3. distributive + others

CNF → G if $\neg(\text{CNF} \to G)$ inconsistent

1943 McCulloch & Pitts invent NN

1956 Newell & Simon develop "The Logic Theorist"

1958 Rosenblatt invent perceptron, first trainable NN

1959 general problem solvers

1960 Widrow & Hoff dev ADELINE, gd NN

56-62 McCarthy invent AI programing lang LISP

64 STUDENT algebra prob solver

68 ANALOGY analogy prob solver

69 Minsky & Papert pub. ↓ of perceptron

70s mostly hardcoded AI

76 Newell & Simon propose physical system hypothesis

80s symbolic logic AI (prolog, neocognitron NN, conv NN, back prop)

90s Learning from data (ML, prob Bayesian & hidden Markov) SVM

00s Stats natural lang processing

learning from unstructured data

IBM Watson (Jeopardy)

Human based computation (recaptcha, amazon mechanical turk)

10s age of DL

computer vision - conv NN

speech - recurrant NN

NLP - recurrant NN -> transformers

DRL - games & driving

Game History

52 Samuel's Checker Program 10Kb

92 Backgammon Tesauro's NN

97 Chess IBM Deep Blue

15 Atari DeepMind

16 AlphaGo beats GM

19 Atari + Board Games

19 Starcraft 2