Artificial Intelligence Recap

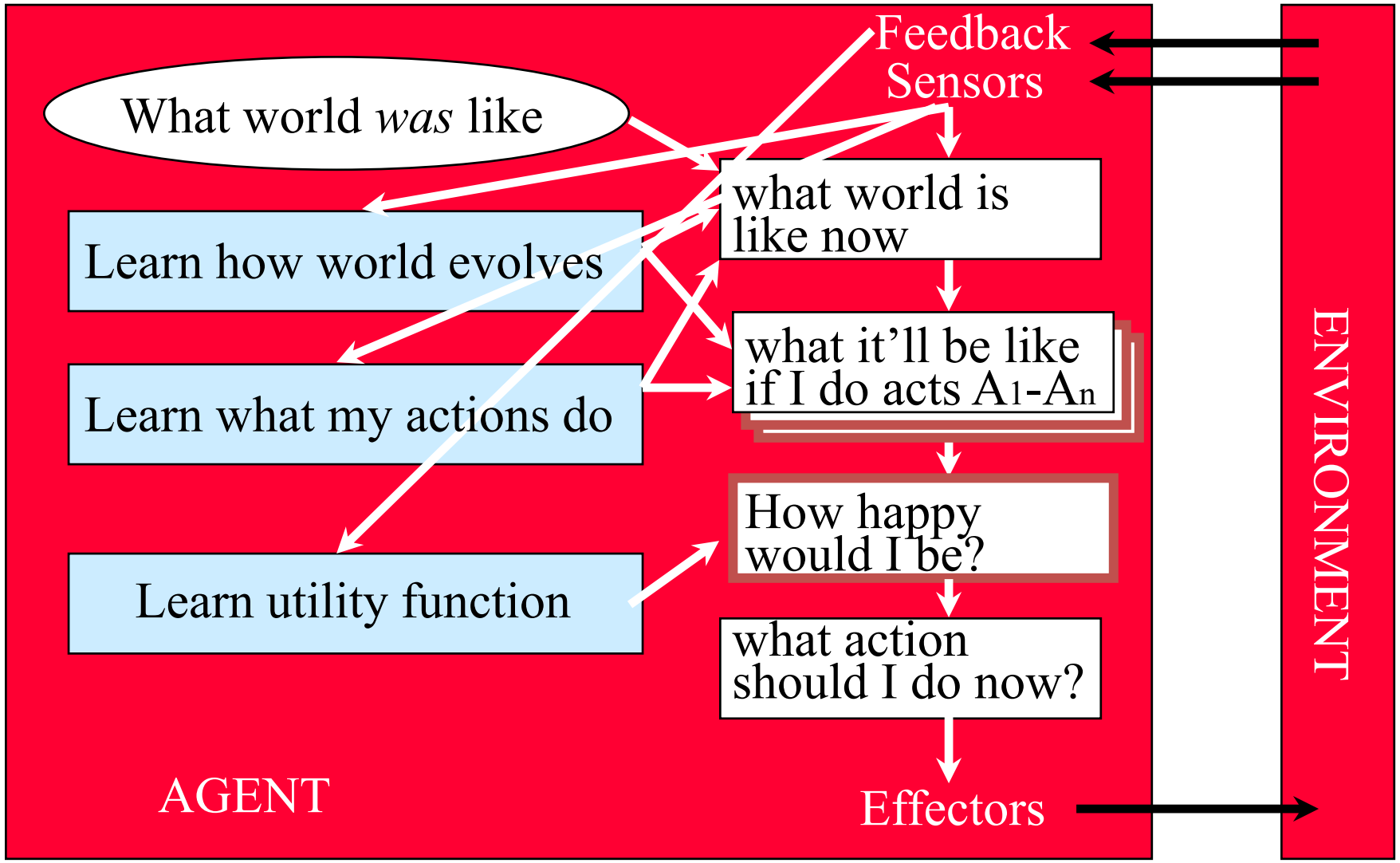
Mausam

What is intelligence?

- (bounded) Rationality
  - We have a performance measure to optimize
  - Given our state of knowledge
  - Choose optimal action
  - Given limited computational resources

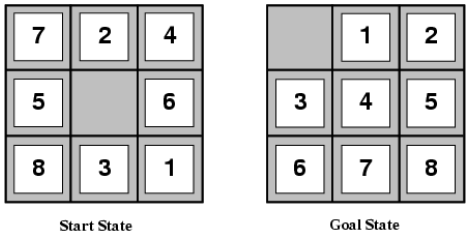
- Human-like intelligence/behavior

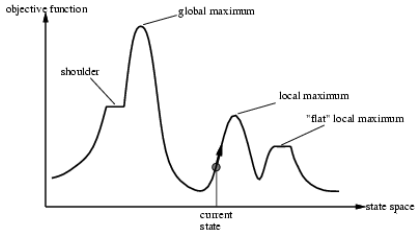
Learning agents



Search in Discrete State Spaces

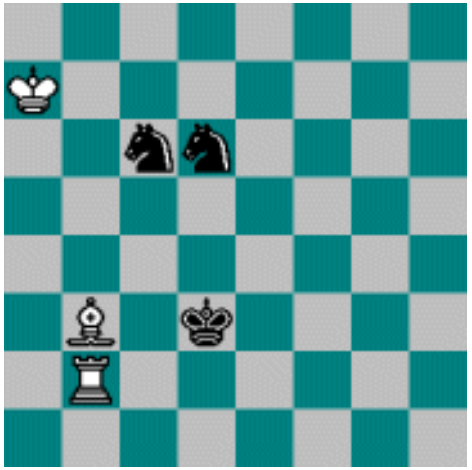
- This is different from Web Search ☺
- Every discrete problem can be cast as a search problem.
- (states, actions, transitions, cost, goal-test)
- Types
  - uninformed systematic: often slow
    - DFS, BFS, uniform-cost, iterative deepening
  - Heuristic-guided: better
    - Greedy best first, A*
    - relaxation leads to heuristics
  - Local: fast, fewer guarantees; often local optimal
    - Hill climbing and variations
    - Simulated Annealing: global optimal
    - Genetic algorithms: somewhat non-local due to crossing over
  - (Local) Beam Search





Search Example: Game Playing

- Game Playing
  - AND/OR search space (max, min)
  - minimax objective function
  - minimax algorithm (~dfs)
    - alpha-beta pruning
  - Utility function for partial search
    - Learning utility functions by playing with itself
  - Openings/Endgame databases
    - Secondary search/Quiescence search



Knowledge Representation and Reasoning

- Representing: what I know
- Reasoning: what I can infer

- Logic

- CSPs

- Bayes Nets

KR&R Example: Propositional Logic

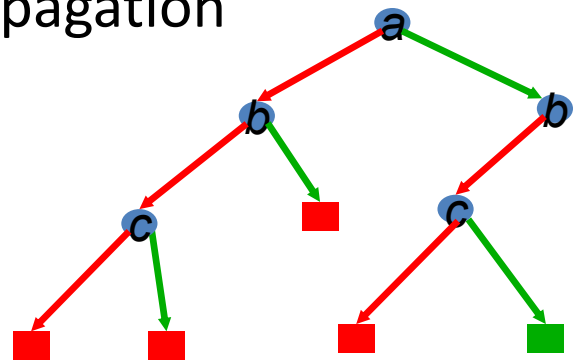
- Representation: Propositional Logic Formula
  - CNF, Horn Clause,…
- Reasoning: Deduction
  - Forward Chaining
  - Resolution
- Model Finding
  - Enumeration
  - SAT Solving

Search+KR&R Example: CSP

- Representation
  - Variables, Domains, Constraints
- Reasoning: Constraint Propagation
  - Node consistency, Arc Consistency, k-Consistency
- Search
  - Backtracking search: partial var assignments
    - Heuristics for choosing which var/value next
  - Local search: complete var assignments
- Tree structured CSPs: polynomial time
- Cutsets: vars assigned → converts to Tree CSP

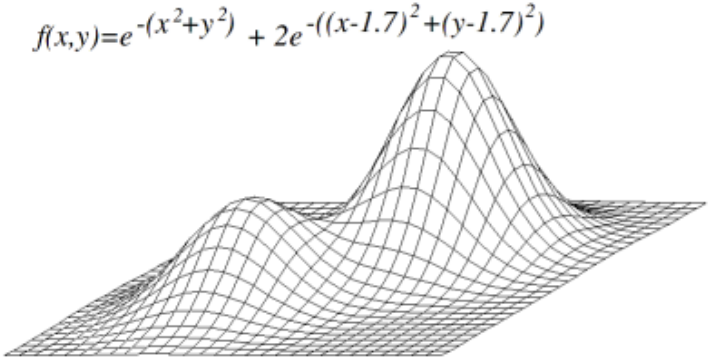
Search+KR&R Example: SAT Solving

- Representation: CNF Formula
- Reasoning
  - pure literals; unit clauses; unit propagation
- Search
  - DPLL (~ backtracking search)
    - MOM's heuristic
  - Local: GSAT, WalkSAT
- Advances
  - Clause Learning: learning from mistakes
  - Restarts in systematic search
  - Portfolio of SAT solvers; Parameter tuning



- Phase Transitions in SAT problems

KR&R Part 2: Continuous Spaces

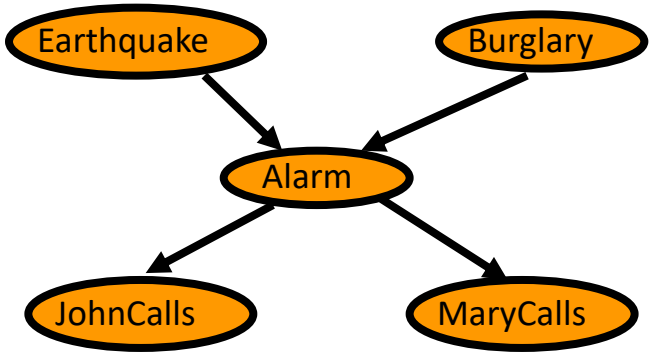


- Search
  - Gradient Descent
  - Newton Raphson
  - Optimization (convex/non-convex…)

- Constraint Optimization (we didn't study this)
  - Linear Programming
  - Integer Linear Programming
  - Mixed Integer Linear Programming

KR&R: Probability

- Representation: Bayesian Networks
  - encode probability distributions compactly
    - by exploiting conditional independences



- Reasoning
  - Exact inference: var elimination
  - Approx inference: sampling based methods
    - rejection sampling, likelihood weighting, Gibbs sampling

KR&R: One-step Decision Theory

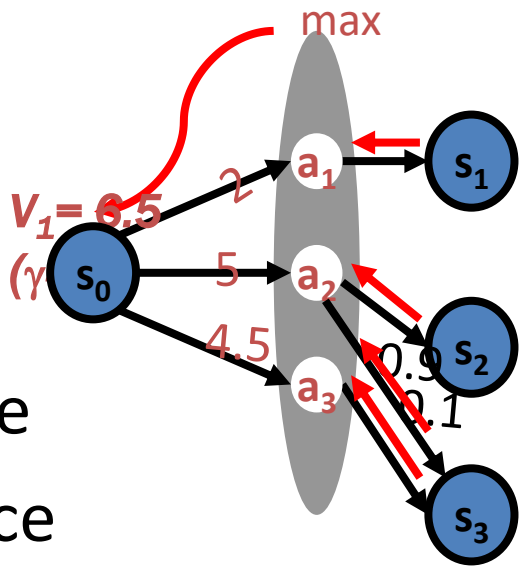
- Representation
  - actions, probabilistic outcomes, rewards
- Reasoning
  - expected value/regret of action
  - Expected value of perfect information

	States of Nature	
Actions	**Favorable Market**	**Unfavorable Market**
Large plant	**$200,000**	**-$180,000**
Small plant	**$100,000**	**-$20,000**
No plant	**$0**	**$0**

- Non-deterministic uncertainty
  - Maximax, maximin, eq likelihood, minimax regret..
- Utility theory: value of money…

KR&R: Markov Decision Process

- Representation
  - states, actions, probabilistic outcomes, rewards
  - ~AND/OR Graph (sum, max)
- Reasoning: V*(s)
  - Value Iteration: search thru value space
  - Policy Iteration: search thru policy space

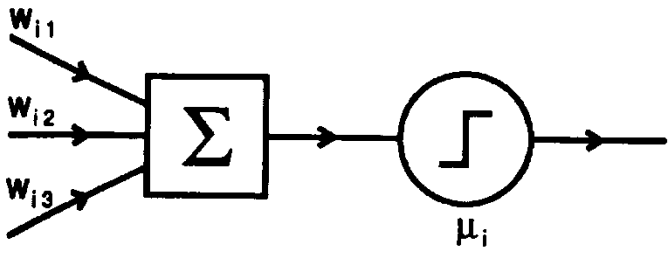


Learning: BNs/NB

- ML estimation. max $P(D|\theta)$
  - counting; smoothing
- MAP estimation max $P(\theta|D)$..
  - Gradient descent

Learning: Neural Nets

- Representation
  - Network of weighted sum + non-linearities
- Reasoning
  - Forward pass
- Learning
  - Gradient descent
  - Backpropagation through network



- Advantage: feature learning

Learning: Reinforcement Learning

- Learn model while taking actions
- What to learn
  - T and R: model based
  - Policy: Model free
- Which actions to take
  - Exploration – Exploitation
- Large state spaces
  - function approximation

AI we didn't cover

- Temporal models: HMMs, Kalman filters…
- Ontologies
- Robotics
- Vision
- Mechanism design
- Multi-agent systems
- Sensor Networks
- Computational Neuroscience
- …

AI is about problems.

- It is an application-driven field
- Happy to beg, borrow, steal ideas from anywhere
- Traditionally discrete … more and more cont.
- Traditionally logic… almost all probability
  - Recent close connections with EE/Stat due to ML
- HUGE field

Applications of AI

- Mars rover: planning
- Jeopardy: NLP, info retrieval, machine learning
- Puzzles: search, CSP, logic
- Chess: search
- Web search: IR
- Text categorization: machine learning

- Self-driving cars: robotics, prob. reasoning, ML…

AI-Centric World ☺