Search In Combinatorial Spaces:
Fun with puzzles and games

Bart Massey (Physics '87)
Asst. Prof. Computer Science
Portland State University
bart@cs.pdx.edu

Why puzzles and games?

- Vernor Vinge: Usenix 2005 Keynote Talk: *Possible Futures of Software*
  - exponential collapse
  - indefinite exponential growth (!)
  - exponential saturation
- Key point – two kinds of computation
  - boring polytime
  - NP-hard and worse
- Puzzles and games "don't scale"
  - given P≠NP, strong polytime thesis

Topics

- About the problem
- Search spaces induced by puzzles & games
- Single-agent search
  - complete
  - local
- Adversary search
- Some p&g I've been playing with
  - a word puzzle
  - optimal Boggle boards
  - optimal 2-player Yahtzee

P vs NP

- Instance vs Problem vs Class
- Decision problems and instance size
- Completeness for class: reductions
- P: Class of problems solvable in time (bounded by) polynomial in size of instance
- NP: Class of problems with a cert *checkable* in P – “guess and check”
- P=NP? NP=co-NP? NP-complete?
- PSPACE, EXPTIME

Why Search

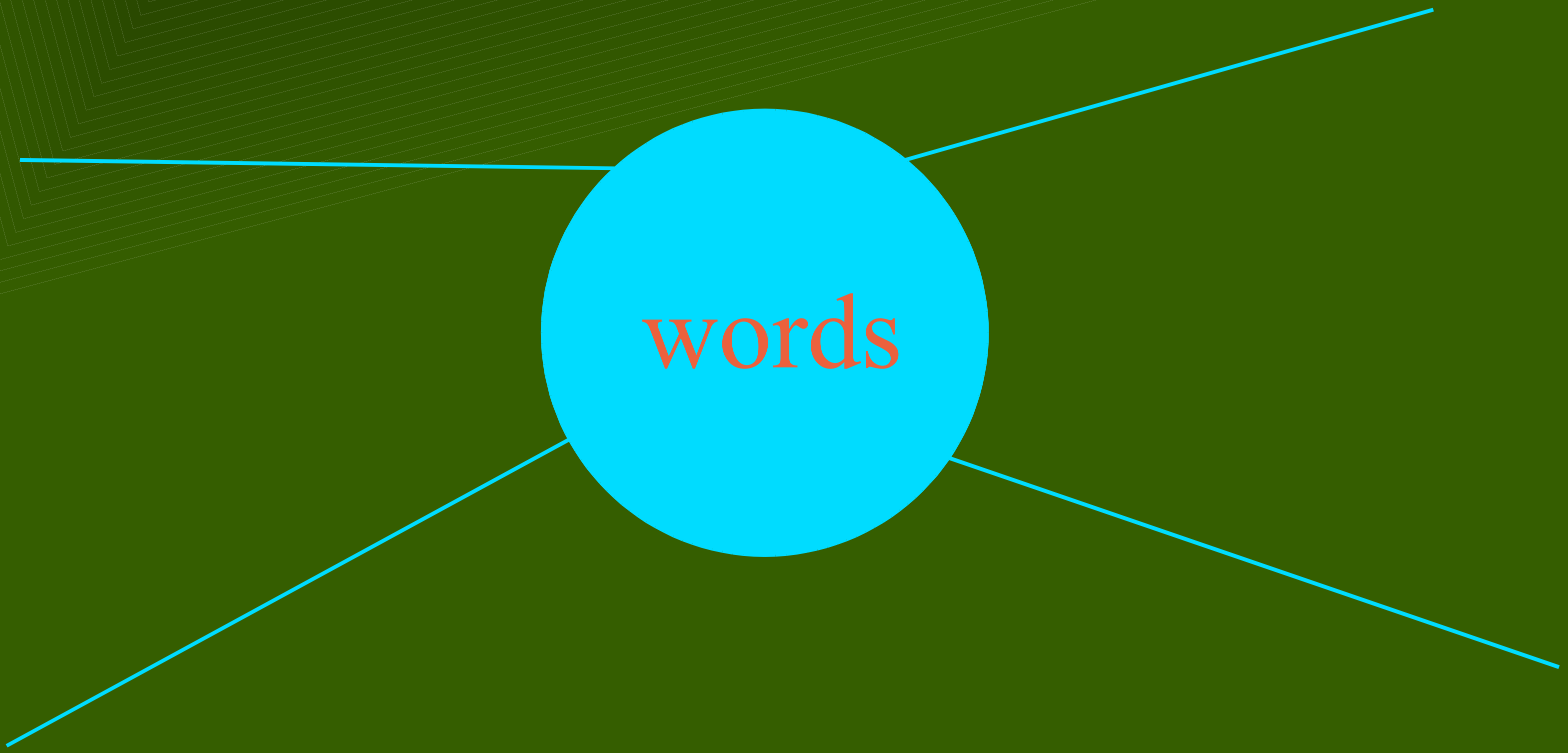
- Idea: Examine only tractable instances of hard problems!
  - everything is constant time/space :-)
  - these instances matter most anyhow
- But how?
  - clever mathematics (too hard for me)
  - brute force (too slow)
  - brute force and trickery

Example: Magnetic Letters

- Found on the side of a file cabinet:
  L = wretch sprightly plumb divvy smudge
  off knock jazz wjxxbqqn
- Question: Can all the letters be used simultaneously to make words?
- Consider the various large spaces
  - 2**|D| sets of words (omitting dupes!):
    find a largest collection covered by L
  - huge number of ordered partitions of L:
    find one best covered by D
- Brute force won't work: no time or space

A search space

- Consider graph with
  - node = words + unmatched letters
  - edge = “small” change to node state



State space search

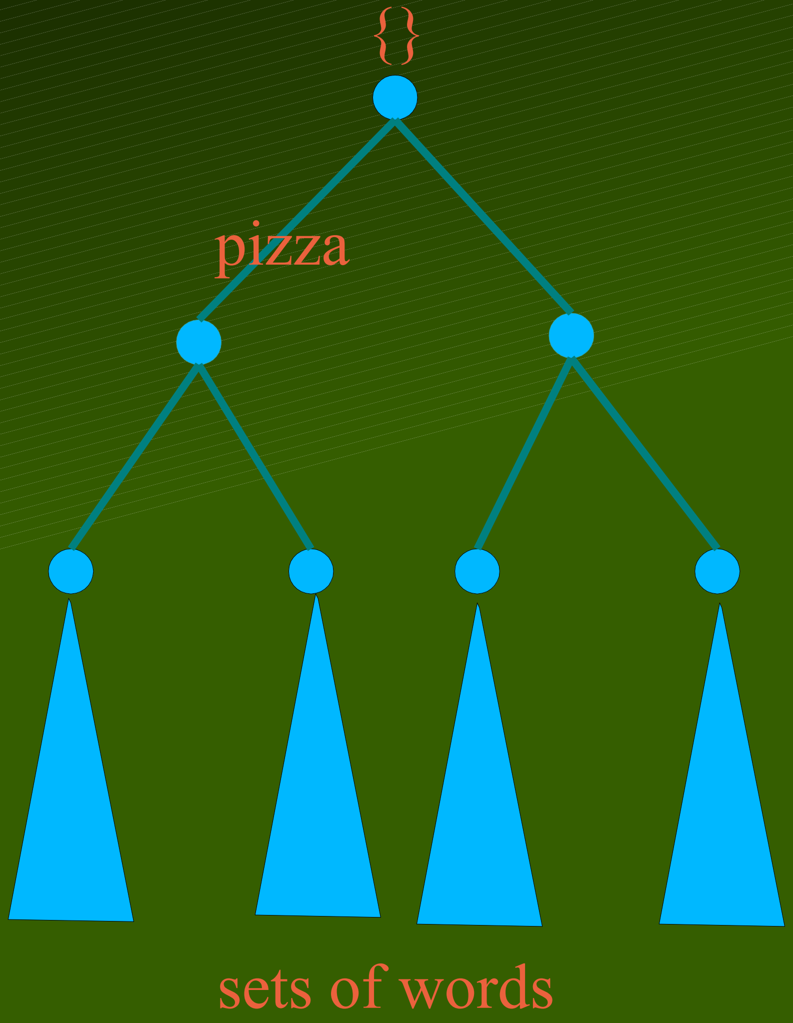
- Traverse the graph from initial condition
  - deterministically
  - stochastically
- Use algorithmic tricks that are
  - general search tricks: e.g. local search
  - problem specific tricks: e.g. dictionary ordering
  - instance specific tricks: e.g. vowel valuation
- May not terminate expeditiously: anytime

An answer

- Here's a better letter puzzle soln

  rhythms crypts blindfold knock fuzz bump wigwag vex vex jjqq

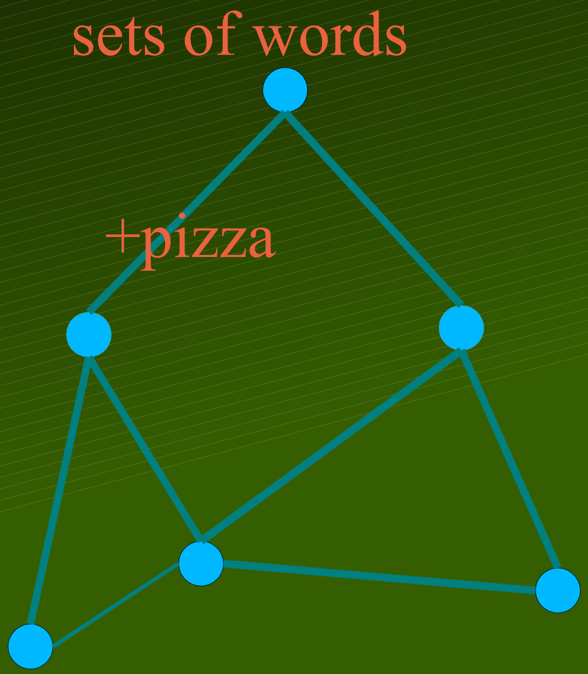
Complete Search



Speeding up complete search

- Search control
  - iterative deepening/broadening
  - limited discrepancy search
- Heuristic search
- Pruning
  - branch-and-bound
  - A*

Local Search



Speeding up local search

- Heuristic functions
- Evading local minima
  - restarts
  - simulated annealing
  - noise moves

Optimal Boggle

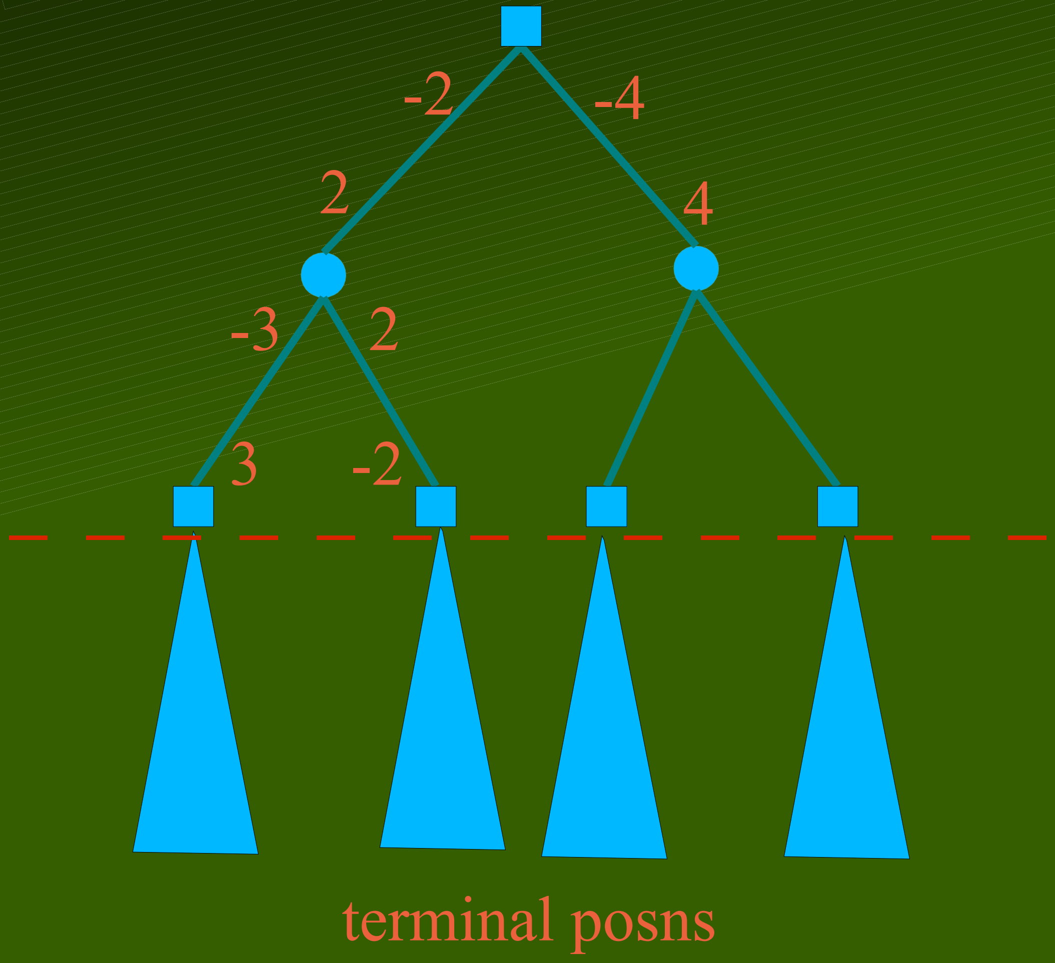
A C G Qu
T E A N
P I B U
R A T A

- Lead: Micah Sheller (PSU undergrad)
  - easy search: score Boggle board fast
  - harder search: find high-scoring board
  - hardest search: find high-scoring dice
- Monte-Carlo local search

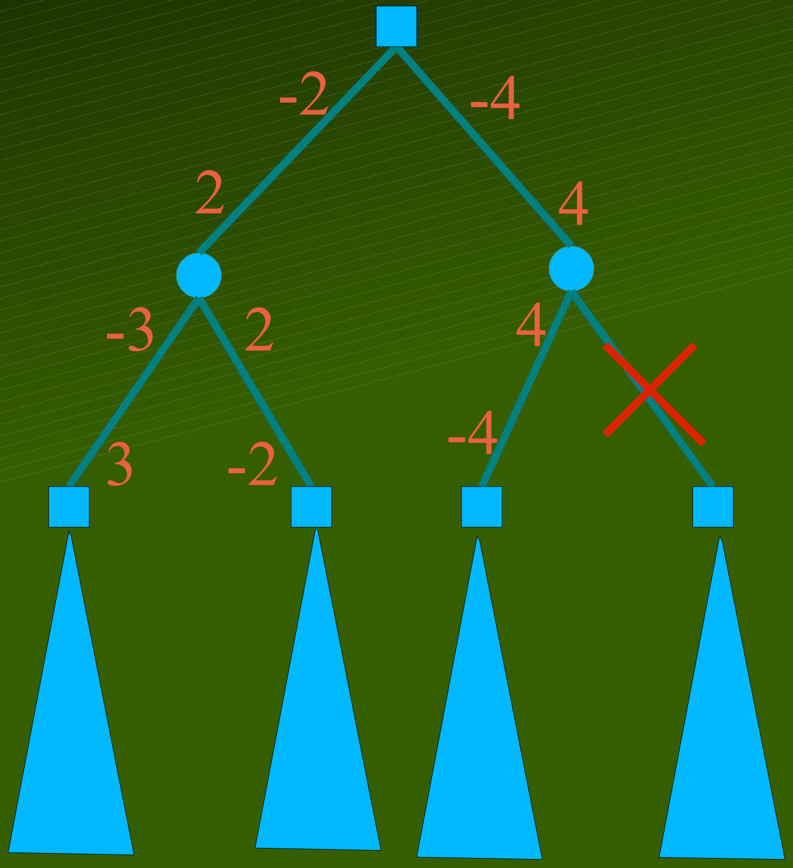
Puzzle vs. Game

- “Adversary is nature” vs “Adversary is intelligent”
- Basic game-theory trick: minimax
  - usually negamax
  - two-player alternating deterministic zero-sum games with no hidden information or likelihood
- produces best outcome vs. best opp. play
  - vs expectimax
  - given complete search

Backing up game tree values



αβ pruning



terminal posns

Speeding up game tree search

- Transposition tables
- Move ordering
- Zero-window search

Optimal 1-player Yahtzee

- Single-agent puzzle
- Probability means large branching factor
- Can use a set of tricks to speed up
- Folks do this from time to time

Optimal 2-Player Yahtzee

- Probabilistic “race game”: c.f. backgammon
- Not the same as 1-player!
- Classic adversary search, except
  - forward search requires hidden-info analysis
  - this would mean solving many LP problems
  - instead, retrograde analysis
- Work in progress (for Yacht)

Comments

- Computers are dumb + fast, so use emergent behavior
- Outstanding challenges: puzzles
  - search based theorem proving
  - search based GP planning
- Outstanding challenges: games
  - Go
  - Bridge
  - 3+ player games
  - nonzero-sum “games”