1 handouts: slides

Concurrent Actions

2^k vs incremental

The Planning Graph

Simple Heuristics

1. $h(n) = 0$
2. number of unachieved goals
3. H_1 max
4. H_1 sum

The 'Planning Graph'

2 types of layers: fact and action
track both positive and negative grounded literals
'no-op' frame actions

actions a and b mutex iff:

inconsistency: a deletes add of b
interference: a deletes precondition of b
competing needs: inconsistent preconditions

literals a and b mutex iff:

inconsistent: a is $\neg b$
inconsistent support: all ways of achieving them are mutex

Cake World

Initial: Have(Cake)

Eat: Pre: Have(Cake)
Post: ¬ Have(Cake), Eaten(Cake)

Bake: Pre: ¬Have(Cake)
Post: Have(Cake)

Goal: Have(Cake), Eaten(Cake)

Break

- asst 3
- project proposals: talk with me this week or next
- office hours
- exam 1

Relaxed Plan

building a plan:

- choose no-op when possible
- re-use previously chosen action when possible

H_1 and basic PG assume parallelism: serial planning graph
optimal relaxed plan is admissible but NP-hard
need actions if optimizing costs (not makespan)

Comparison

level-based heuristics

1. poor if many 'concurrent' actions at one level

max vs sum

1. sum poor if positive interactions

h^n

1. poor if negative interactions

Heuristics

1. 0
2. number of unachieved goals
3. H_1 max
4. H_1 sum
5. planning graph max
6. planning graph sum
7. relaxed plan

EOLQs

- What question didn't you get to ask today?
- What's still confusing?
- What would you like to hear more about?

Please write down your most pressing question about AI and put it in the box on your way out.
Thanks!