6.825 Techniques in Artificial Intelligence

**Graph Plan**

- Overview
- PO Planning – “human-like” but very slow
- Graph Plan
  - Simplified planning model
  - Efficient algorithm

**Graph Plan**

- A propositional planner, that is, there are no variables
- Simpler – don’t have to worry about matching
- Bigger – if you have six blocks, you need 36 propositions to represent all $On(xy)$ assertions

1. Make a plan graph of depth $k$
2. Search for a solution
3. If succeed, return a plan
4. Else $k = k + 1$
5. Go to 1.

**Plan Depth**

A plan of depth $k$
- has $k$ times steps
- may have multiple parallel actions per time step

<table>
<thead>
<tr>
<th>$t$</th>
<th>Action Level 1</th>
<th>Action Level 2</th>
<th>Action Level 3</th>
<th>Action Level 4</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>$D_{QA}$</td>
<td>$D_{OB}$</td>
<td></td>
<td></td>
</tr>
<tr>
<td>2</td>
<td></td>
<td>$D_{C}$</td>
<td></td>
<td></td>
</tr>
<tr>
<td>3</td>
<td>$D_{D}$</td>
<td>$D_{E}$</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

**Planning vs Scheduling**

Planning: find steps and schedule
- PSPACE-complete

Graph Plan: find plans of a given depth

Scheduling: tasks are fixed
- NP-Complete

**Plan Graph**

- Prop. Level 0: 0
- Action Level 1: 1
- Prop. Level 2: 2
- Action Level 3: 3
- Prop. Level 4: 4
Making the Plan Graph
- Start with initial conditions
- Add actions with satisfied preconditions
- Add all effects of actions at previous levels
- Add maintenance actions

Mutually Exclusive Actions
- Two action instances at level i are mutex if:
  - **Inconsistent effects**: effect of one action is negation of effect of another
  - **Interference**: one action deletes the precondition of the other
  - **Competing needs**: the actions have preconditions that are mutex at level i-1

Mutually Exclusive Propositions
- Two propositions at level i are mutex if:
  - **Negation**: they are negations of one another
  - **Inconsistent support**: all ways of achieving the propositions at level i-1 are pairwise mutex.

Solution Extraction
- If all the literals in the goal appear at the deepest level and not mutex, then search for a solution for each subgoal at level i
- For each subgoal at level i:
  - Choose an action to achieve it
  - If it’s mutex with another action, Fail
- Repeat for preconditions at level i-2

Birthday Dinner Example
- Goal: garb ∧ dinner ∧ present
- Init: garb ∧ clean ∧ quiet
- Actions:
  - **Cook**
    - Pre: clean
    - Effect: dinner
  - Wrap
    - Pre: quiet
    - Effect: present
  - Carry
    - Pre: garb
    - Effect: garb ∧ ¬ clean
  - Doily
    - Pre: garb
    - Effect: garb ∧ ¬ quiet
Extensions

- Lots of time optimizations
- Disjunctive preconditions
- Universally quantified (sort of) preconditions and effects
- Conditional planning