Conditional Planning

Manuela Veloso
CMU 15-381, Fall 2000

- Conditional actions
- Conditional planning algorithm
- Planning and execution

These lecture slides should be complemented with the examples discussed and worked out on the blackboard during the lecture.

What's Coming

- Today, Tuesday, Oct 17 - Homework 4 is out - Planning; Due in two weeks, Tuesday, Oct 31. Form your own groups - up to 2 people.
- Thursday - NO lecture here. Instead attend Earthware Symposium, `http://www.cs.cmu.edu/~earthware/` McConomy Auditorium, University Center:
  - 10:20am - Bill Joy (synthetic interview): Will the Future Need Us?
  - 11:00am - Hans Moravec: Mass Utility Robots this Decade: Finally!
  - 11:35am - David Gelernter: A good World in 2050: will computer help or hinder?

What's Coming

- Oct 24 - Nov 2: Learning for Classification
- Nov 7 - Nov 21: Learning to Act (control learning)

Uncertainty in the World

- The goal of planning is *to execute* the plans generated!
- Planning models the world - actions, predicates, objects, choices for plan generation
- Sources of uncertainty
  - State may be unknown
  - Exact effects of actions may be unknown
  - External events may change the world

Handling Uncertainty

- **Replanning:**
  1. Assume model is correct and plan
  2. **Wait** for eventual execution failures
  3. React to failures - Plan again
- **Conditional planning:**
  1. Add/represent observable contingencies
  2. Create a *branching plan* for contingencies
  3. Observe and execute appropriate branch
- **Probabilistic planning:** Plan for the *most probable* contingencies.

Replanning and Conditional Planning

Replanner: *"Why are you taking an umbrella? It's not raining!"*

Replanners can't find plans that require them to have steps *before* the contingency.

Conditional Planner: *"Why are you leaving the house? The class might be cancelled, there may be an earthquake, you may have won the pools,...*

Conditional Planners need to consider *every* contingency in a plan.

The Flat Tire Operators

```
Operator REMOVE (x)
p: On (x)

a: Off (x) and
   ClearHub (x)
d: On (x)
```

```
Operator PUT-ON (x)
p: Off (x) and
   ClearHub (x)
a: On (x)

d: ClearHub (x) and
   Off (x)
```

```
Operator INFLATE (x)
p: Intact (x) and
   Flat (x)
a: Inflated (x)
d: Flat (x)
```

A Flat Tire Problem

- Initial state:
  - On (Tire1) Flat (Tire1),
  - Inflated (Spare), Intact (Spare), and Off (Spare)
- Goal:
  - On (x) and Inflated (x)
- Plan:
  - Remove(Tire1), Put-On (Spare)

Possible Contingencies

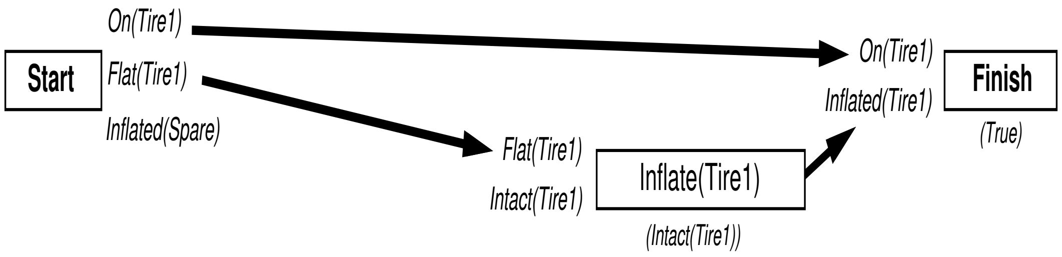
- The tire may just not be inflated.
- We would like a plan:
  - If Intact (Tire1) Then Inflate (Tire1)
  - Else Remove (Tire1) and Put-On (Spare)
- Mark precondition Intact as *unknown* and check.
- Three-valued logic: True, False, Unknown.

The Flat Tire Operators

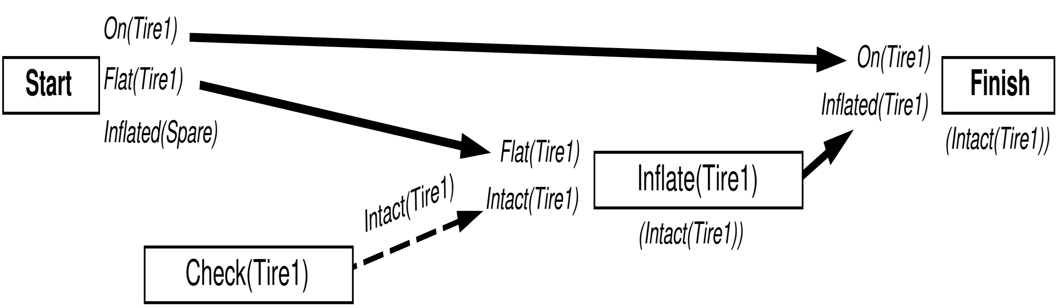
```
Operator REMOVE (x)        Operator PUT-ON (x)
p: On (x)                  p: Off (x) and
                              ClearHub (x)
a: Off (x) and             a: On (x)
   ClearHub (x)
d: On (x)                  d: ClearHub (x) and
                              Off (x)

Operator INFLATE (x)       Operator CHECKTIRE (x)
p: UNK (Intact (x))        p: Tire (x)
   Flat (x)
a: Inflated (x)            a: KnowsWhether
                                   (Intact (x))
d: Flat (x)
```

Initial Plan

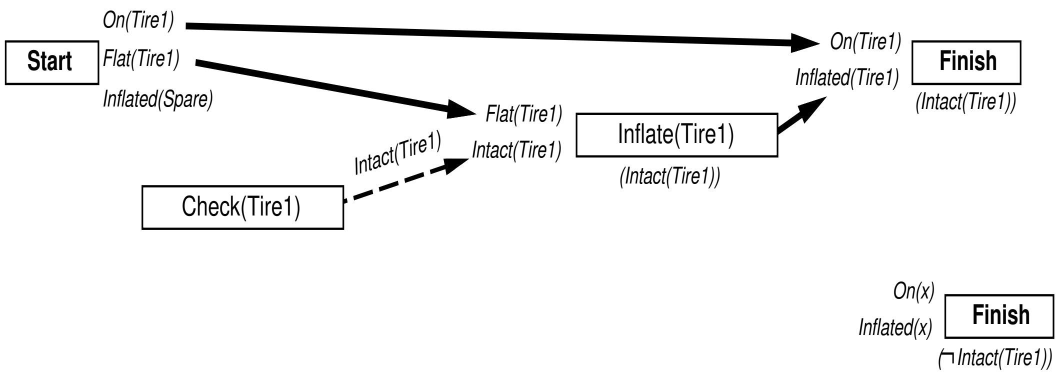
After adding the Inflate step



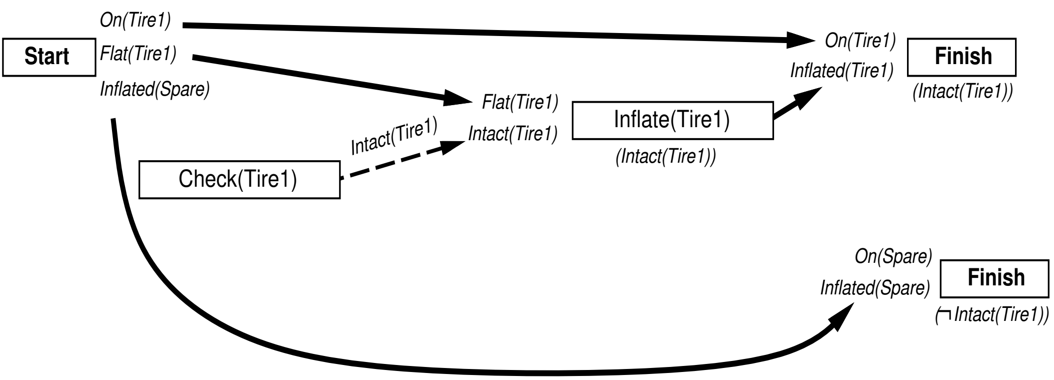
Adding Conditional Link



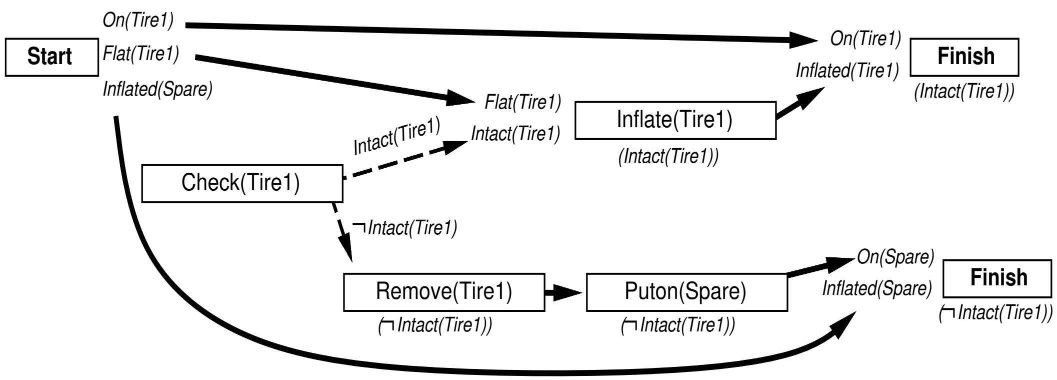
Set the New Branching Goal



Plan for New Branch



Complete Conditional Plan



Extensions to Partial-Order Planning

- Parameterized plans
- Threat resolution
  - Conditioning: can remove threat by separating contexts (branches)
- See CPOP - Conditional Partial-Order Planning
- Page 399 of textbook - Chapter 13

Probabilistic Planning

Planning example: logistics domain

- Objects are *packages*, *trucks*, *places*, *lockers*.
- Operators are *drive*, *load*, *unload*, *store*, *unstore*.
- *load* has uncertain outcomes.
- One event: *lose-package-from-post-office*.

Probabilistic Uncertainty - State

Simple approach: just list possible initial states with their probabilities.

0.4 (at truck1 boston)
(at package pittsburgh)
(at airplane glasgow)

0.4 (at truck1 boston)
(at package pittsburgh)
(at airplane london)

0.2 (at truck1 boston)
(at package philadelphia)
(at airplane london)

Probabilistic Uncertainty - Actions

Just like the initial state: list possible sets of effects:

```
(OPERATOR LOAD-TRUCK
  (preconds
   ((<obj> OBJECT)
    (<truck> TRUCK)
    (<loc>  LOCATION))
   (and (at-obj <obj> <loc>)
        (at-truck <truck> <loc>)))
  (p-effects
   ((0.7 ()
         ((add (inside-truck <obj> <truck>))
          (del (at-obj <obj> <loc>))))
    (0.25 ()
          nil)
    (0.05 ()
          ((add (lost <obj>))
          (del (at-obj <obj> <loc>)))))))
```

Probabilistic Uncertainty - External Events

Jim Blythe, Weaver.

```
(Event
 lose-package-from-post-office
 (params <package> <post-office>)
 (probability 0.05)
 (preconds
  ((<package> Object)
   (<post-office> Post-office))
  (and (~ (protected <package>))
       (at-obj <package> <post-office>)))
 (effects
  ()
  ((del (at-obj <package> <post-office>))
   (add (lost <package>)))))
```

Similar to operators, but not controlled by the planner.

Weaver - Blythe 95; Probabilistic Planning

- Builds an initial plan *ignoring* uncertainty.
- Uses a Bayesian network both:
  1. to evaluate the plan's probability of succeeding
  2. to find parts of the plan that need to be improved.
- Two new plan refinements to improve the plan:
  1. conditional planning (but now uncertainty is not just from the initial state).
  2. "protection" of conditions to defeat bad events and steps.

Example

Initial state:

```
(at-obj package1 pgh-po)
(at-taxi pgh-taxi pgh-airport)
(loc-at pgh-locker pgh-po)
(have-key pgh-locker)
```

Goal state: `(at-obj package1 pgh-airport)`

Simple plan: drive to post office, get the package, drive back to the airport, unload the package.

Summary

- Planning domains very often have uncertainty.
- Re-planning and conditional planning are two extreme ways to cope with uncertainty.
- Conditional planning creates a branching plan for contingencies. *You need to know the conditional planning procedure - chapter 13.*
- Probabilistic planning is in the middle ground.
- We will see probabilistic planning in more detail after we study probabilistic representations and Bayesian networks.