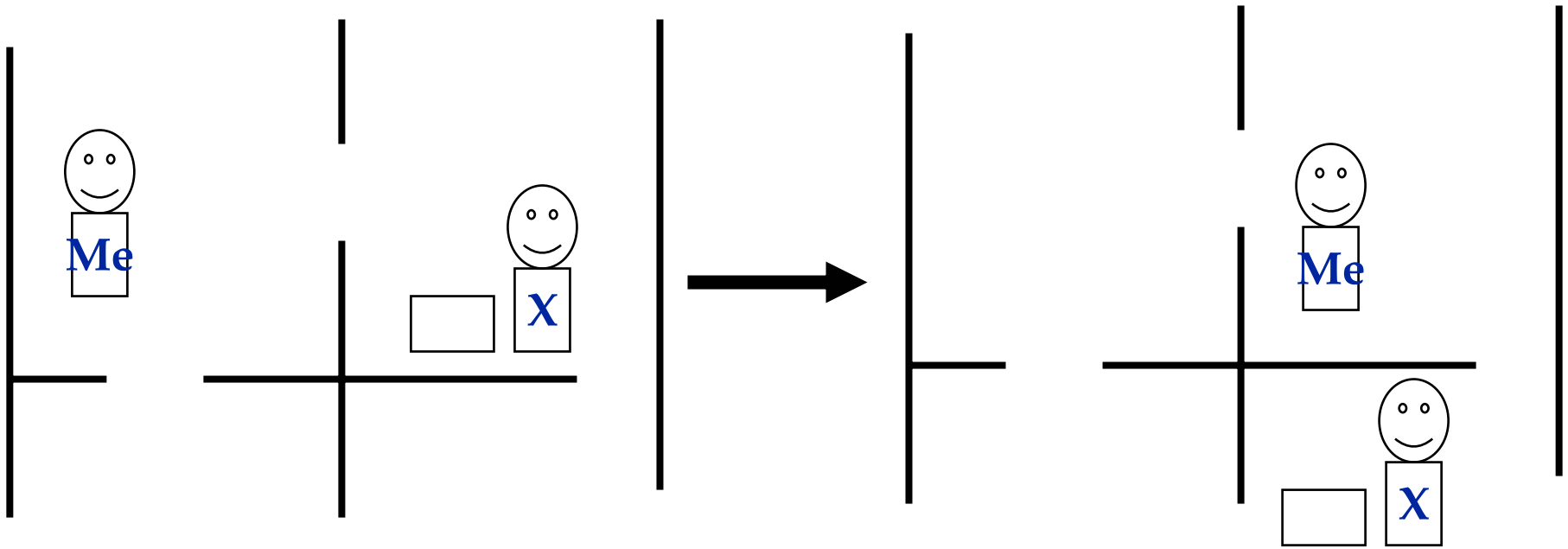
Probabilistic Planning 2: Exogenous events

Jim Blythe

November 11th

Recap: uncertainty from external change

External agents might be changing the world while we execute our plan.



Representing external sources of change

Model actions that external agents can take in the same way as actions that the planner can take.

(*event* oil-spills
  (probability 0.1)
  (preconds
    (and (oil-in-tanker <sea-sector>)
         (poor-weather <sea-sector>)))
  (effects
    (del (oil-in-tanker <sea-sector>))
    (add (oil-in-sea <sea-sector>))))

Random external processes

- Some agents, like robot agent X, have intentions, beliefs and desires, and their actions are based on planning
  - May be co-operative, neutral or adversarial
- Some “external agents” like weather, can be thought of as random processes
  - Not affected by knowledge of our goals
  - Can’t argue with forces of nature
  - But sometimes we can influence random processes indirectly, through states of the world that affect their outcomes.

Impact of random events on planning

- Many random events are constantly taking place in most domains in which we execute plans
- Most do not affect the plans we execute
- Given a plan being considered
  - (e.g. move a barge to some location, use it to clean up spilled oil),

  we can find the random events that do matter
  - (e.g. the weather at that location, how spread out the oil is)

Difficulty of handling random events

- Harder than uncertain action outcomes
  - Have to find the relevant events
  - Effects take place asynchronously
- Easier than co-operative or adversarial planning in general
  - No communication of goals, plans
  - No second-guessing other agents
- Question: does having uncertain external events increase the expressivity of a planner that already has uncertain action outcomes?

Improving plans affected by random events

- Add a conditional branch
- Try to decrease the probability of a bad event, by decreasing the probability of its preconditions or shortening the time during which it can happen.
- Sometimes select a random event as part of a plan (e.g. to wash a car, leave it outside and wait for rain)

  then try to increase probability by increase probability of preconditions or waiting longer.

Example events governing an oil-spill cleanup problem

- The `oil-spills` event from an earlier slide, and:

(*event* weather-brightens
  (probability 0.25)
  (preconds (poor-weather))
  (effects
    (del (poor-weather))
    (add (fair-weather))))

Semantics of STRIPS-style representation of external events

- Many different interpretations might be possible
- Here, assume that at each time point, any event that *could* take place does so with the probability given in the event.

Evaluating a plan in the oil-spill domain

- Given this non-deterministic operator:

```
(operator move-barge
  (preconds (at <barge> <from>))
  (effects
    (0.667
      (del (at <barge> <from>))
      (add (at <barge> <to>)))
    (0.333
      (del (at <barge> <from>))
      (add (at <barge> <to>))
      (del (operational <barge>)))))
```

Consider this conditional plan:

(move barge1 dock spill-site)

IF (operational barge1)

THEN

(pump oil barge1)

ELSE

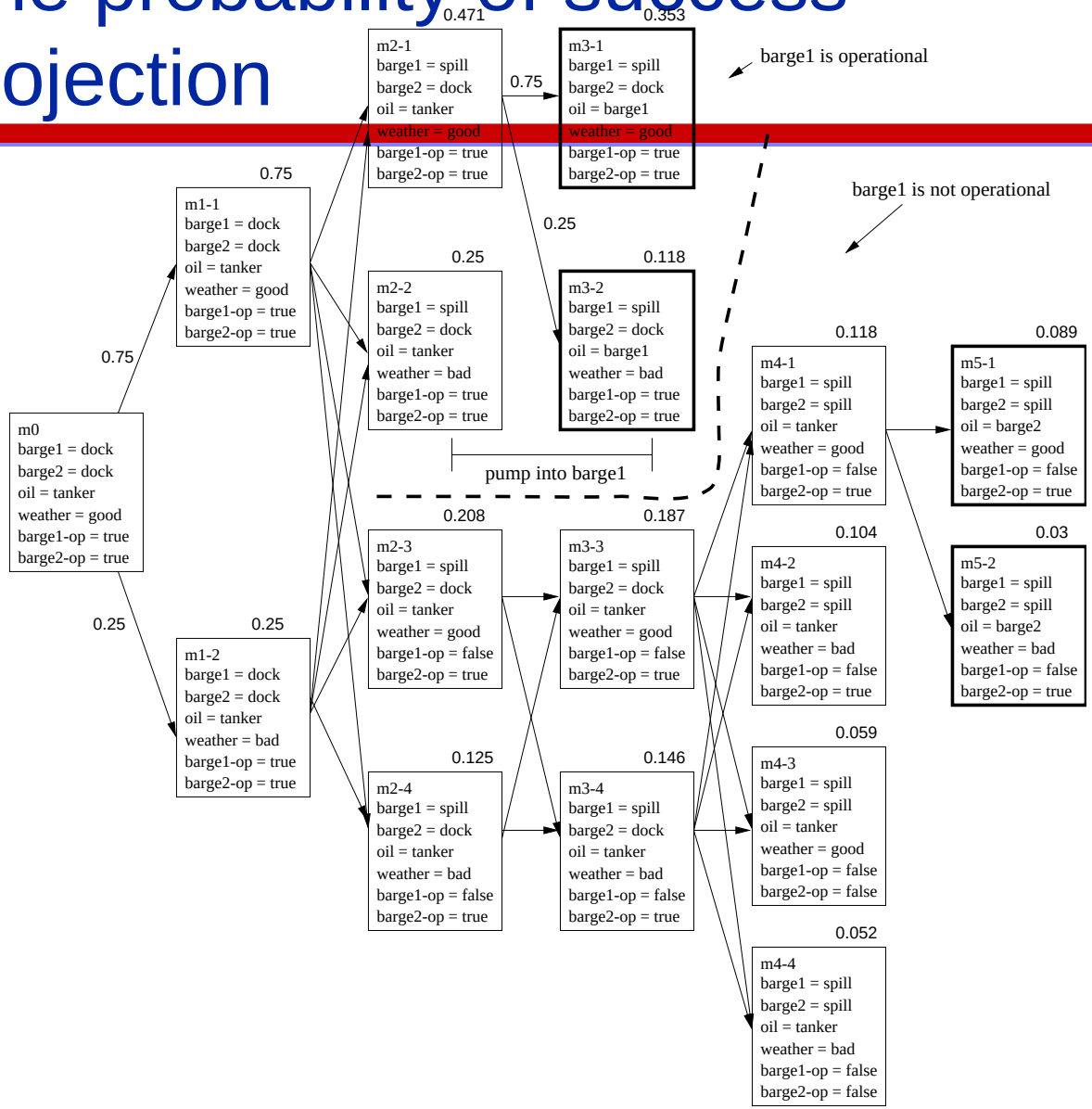
(move barge2 further-dock spill-site)

(pump oil barge2)

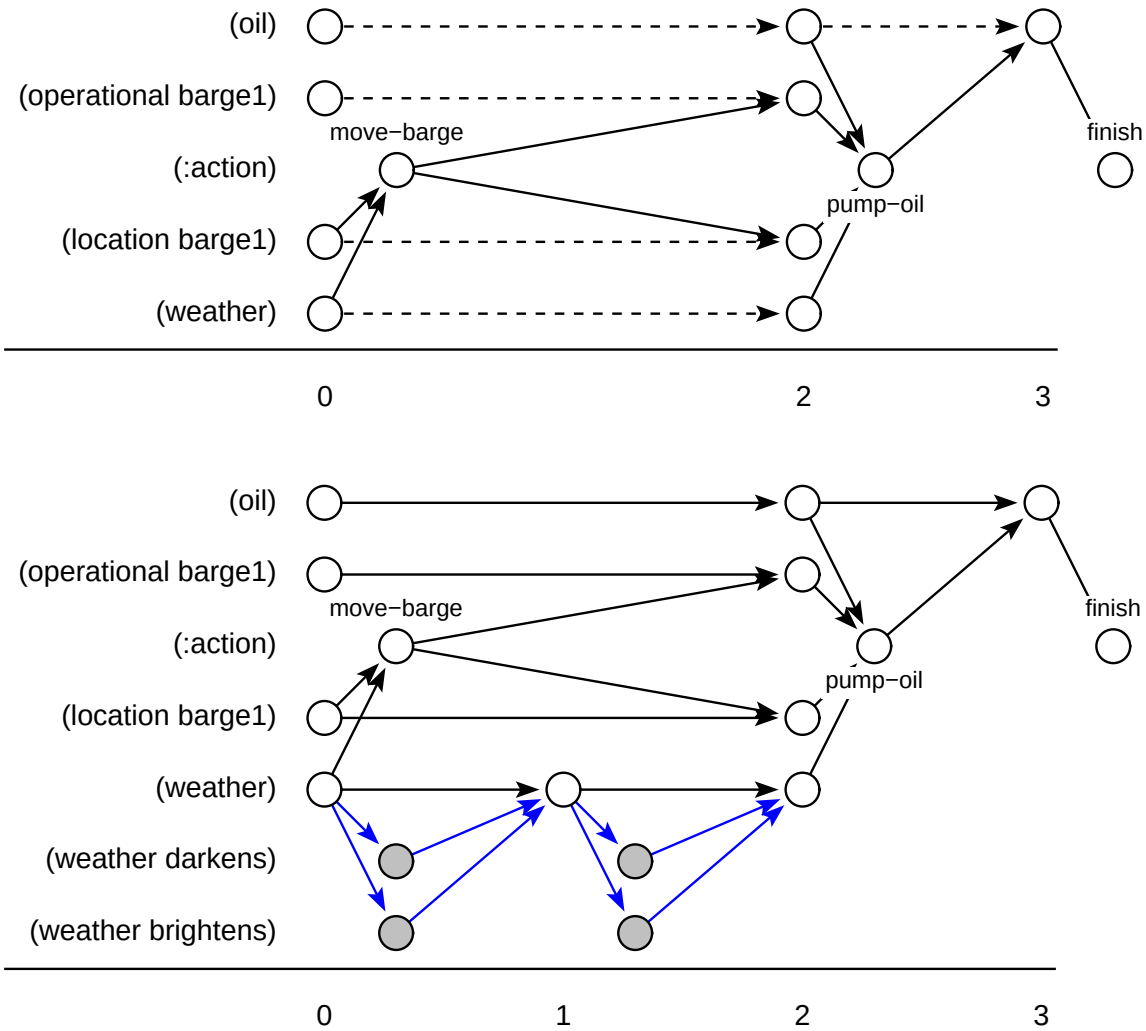
`Pump-oil` has preconds `(operational <barge>)` and `(fair-weather)`.

`Move` takes some time depending on the distance.

Computing the probability of success 1: forward projection

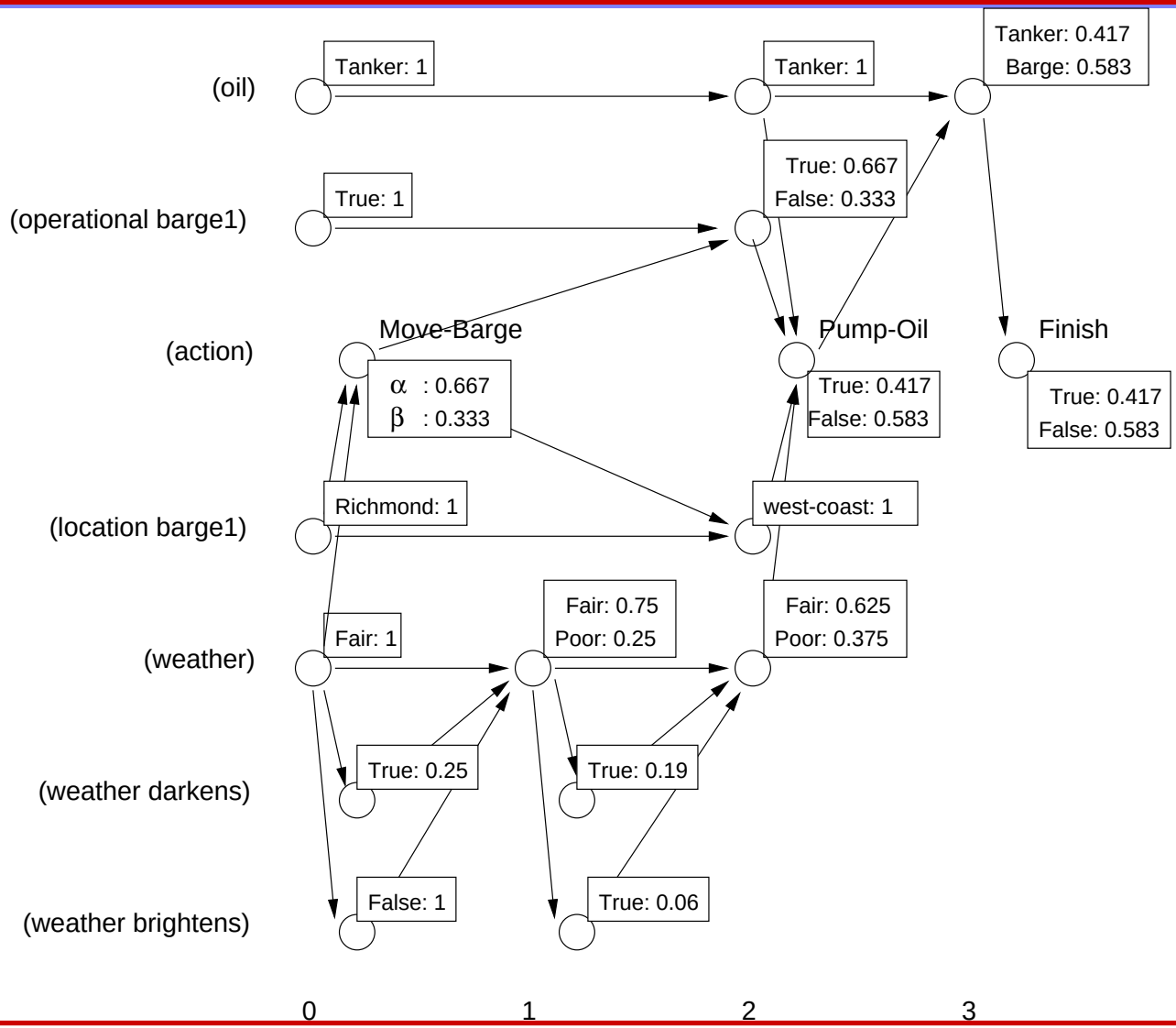


Computing probability of success 2: constructing a belief net from the plan



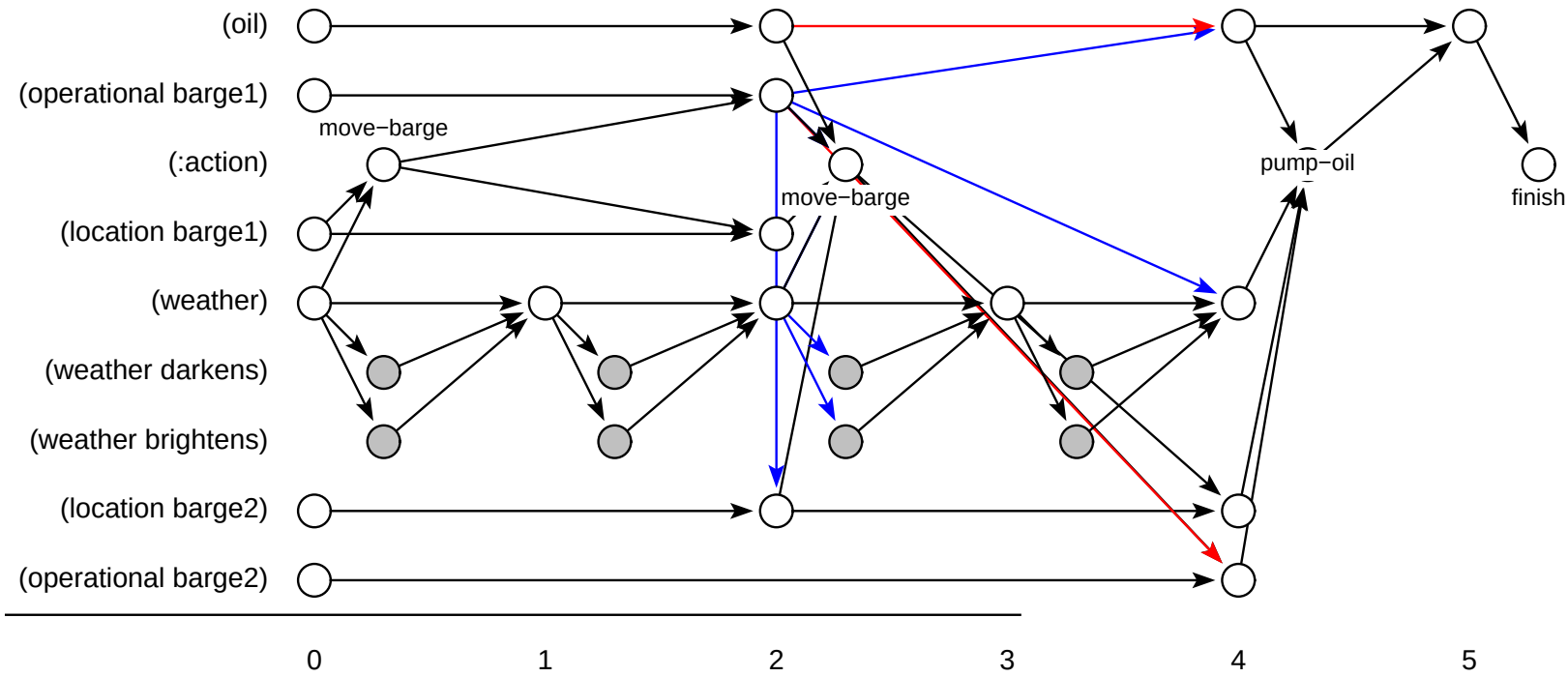
- Add nodes for actions and literals, then investigate "persistence intervals".
- Add any events that might affect persistence intervals in the plan.

Belief net with marginal probabilities



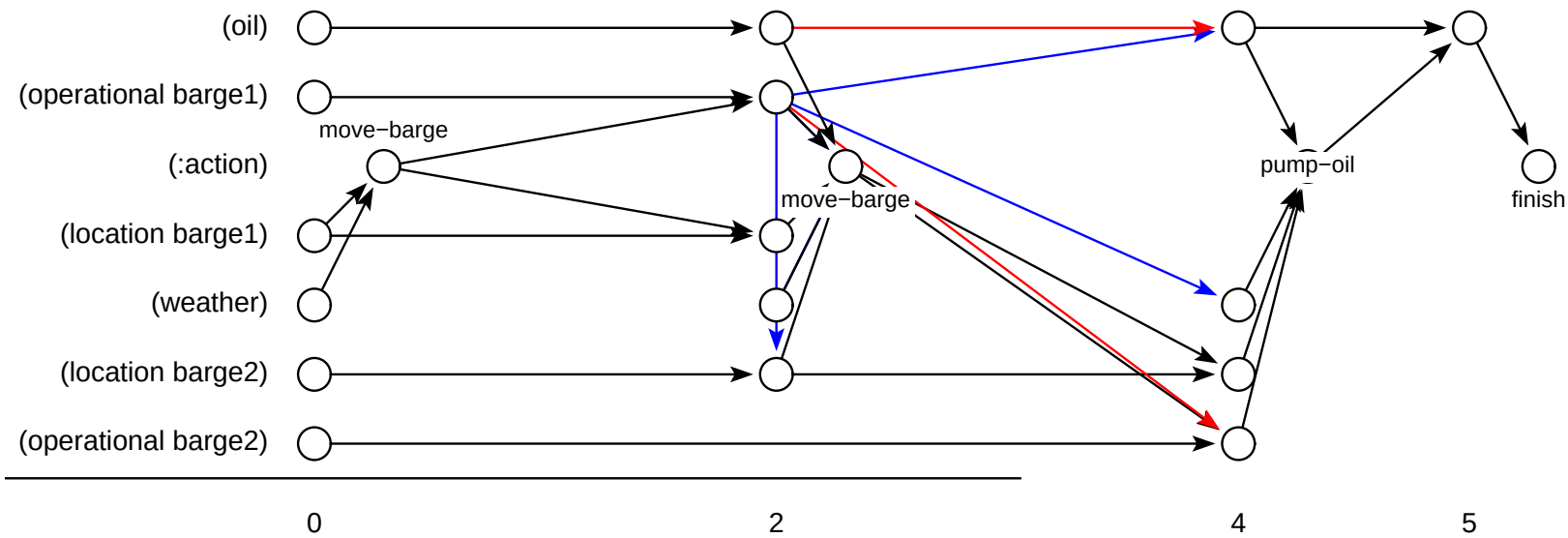
The “explicit events” construction quickly gets expensive:

- This is the second branch of the conditional plan being evaluated.

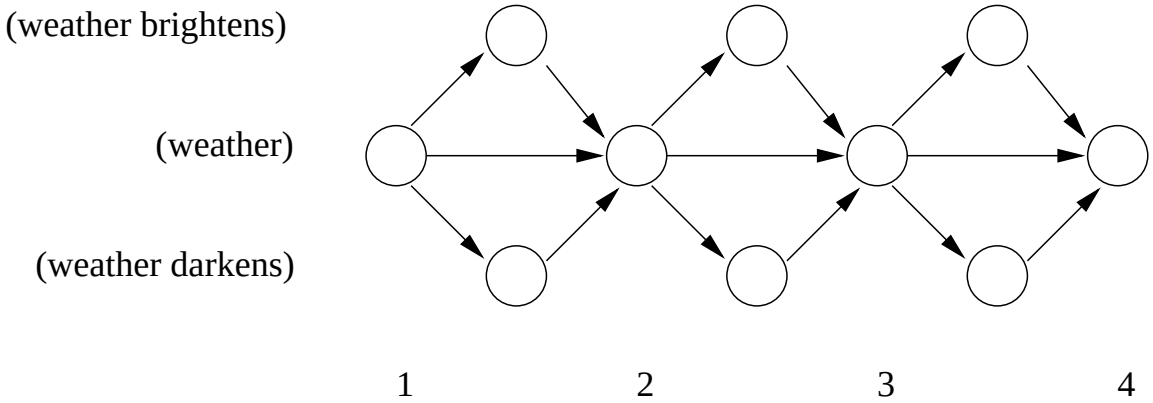


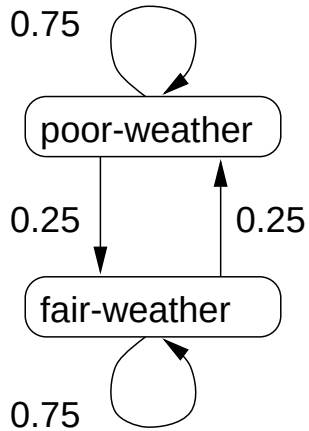
Constructing a cheaper belief net using markov chains.

- The semantics given to events lead them to have a markov chain structure, so the explicit event nodes can be replaced by single arcs as shown here.



Example: the weather events and the corresponding markov chain

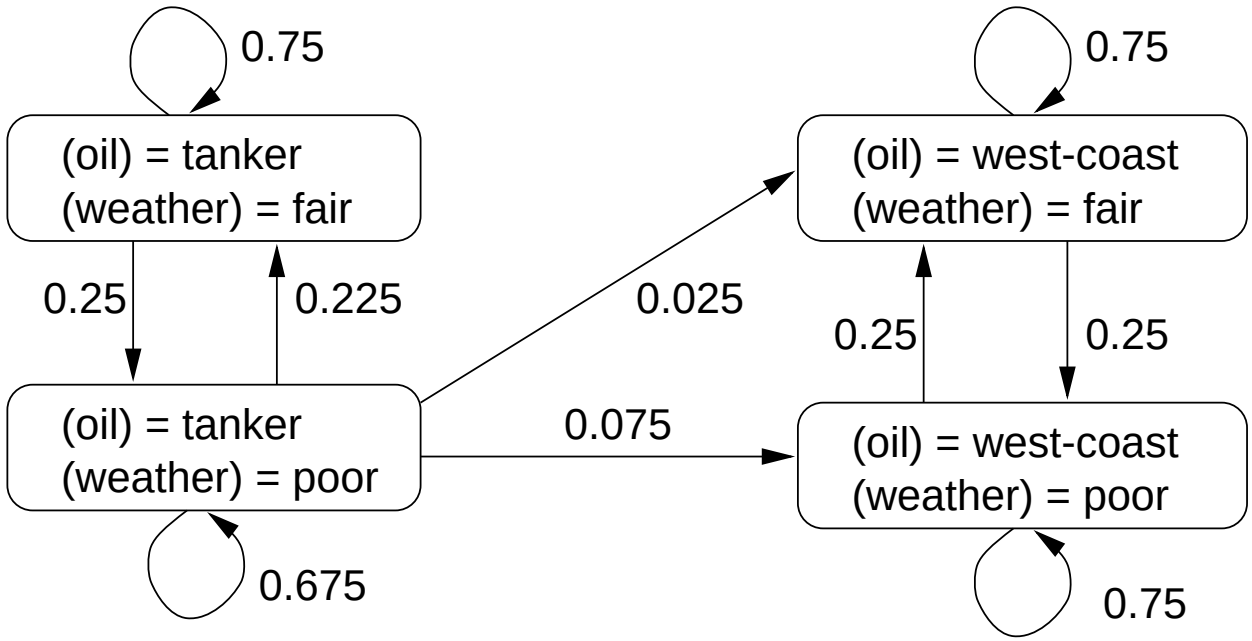




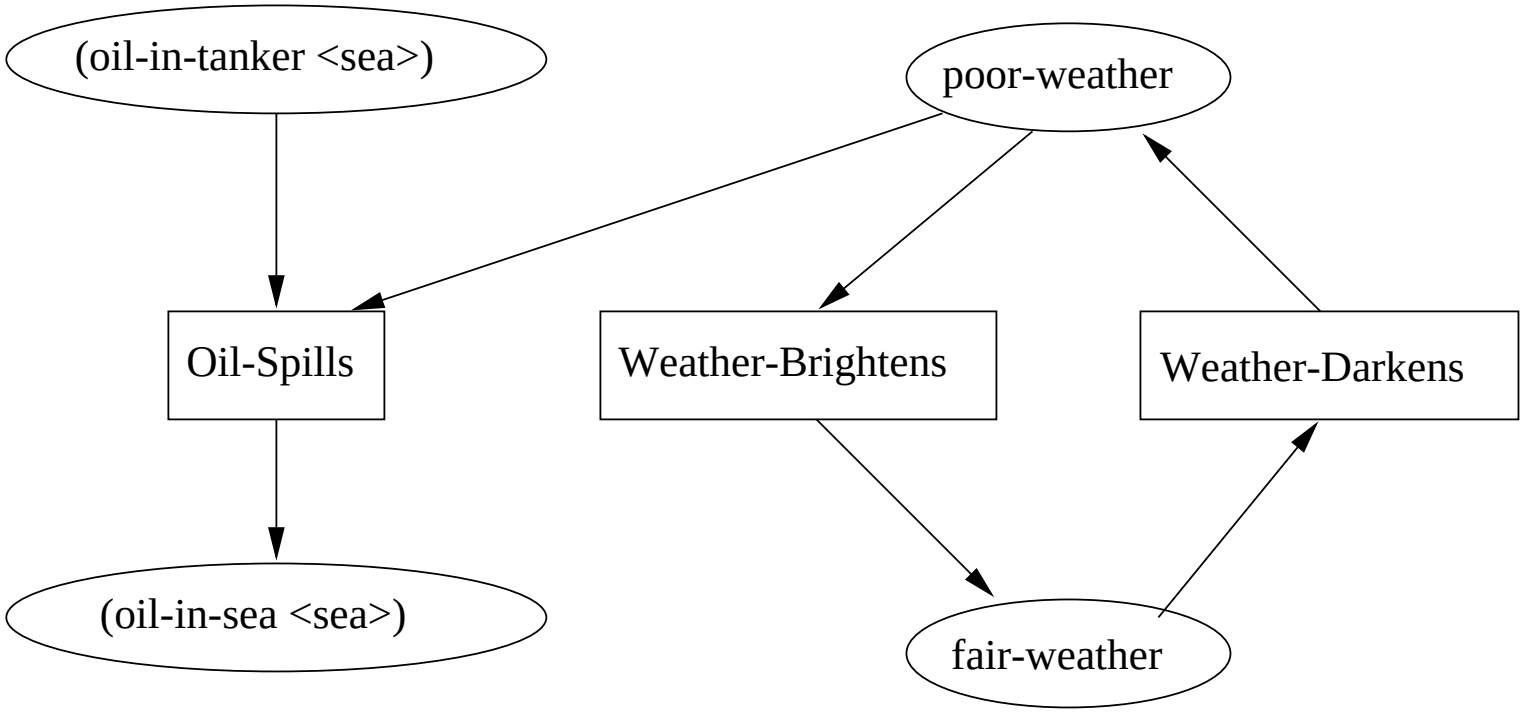
- The markov chain shows possible states independent of time.
- As long as transition probabilities are independent of time, the probability of the state at some future time t can be computed in logarithmic time complexity in t.
- The computation time is polynomial in the number of states in the markov chain.

Wrinkle: how do we know which states need to be included in the markov chain?

- The markov chain to compute the probability of oil spill needs to have four states. Why?



The event graph



- Captures the dependencies between events needed to build *small* but *correct* markov chains.
- Any event whose literals should be included will be an ancestor of the events governing objective literals.

General ideas

- To capture uncertainty from different forms, we can use structures like Markov chains that take advantage of the time-independence of STRIPS-style operators.

- To make computations efficient, we can make use of the *structure* of the problem to remove irrelevant calculations.
  - The same idea is used in efficient planning techniques, e.g. Knoblock's abstraction hierarchies, Etzioni's machine learning.
  - The same idea is also used to try to make MDP planning efficient as we will see next class.