Extending PDDL to Model Stochastic Decision Processes

Håkan L. S. Younes

Carnegie Mellon University

Introduction

- PDDL extensions for modeling stochastic decision processes
  - Only full observability is considered
- Formalism for expressing probabilistic temporally extended goals
- No commitment on plan representation

Simple Example

```
(:action flip-coin
    :parameters (?coin)
    :precondition (holding ?coin)
    :effect (and  (not (holding ?coin))
                  (probabilistic 0.5 (head-up ?coin)
                                 0.5 (tail-up ?coin))))
```

Stochastic Actions

- Variation of factored probabilistic STRIPS operators [Dearden & Boutilier 97]
- An action consists of a precondition ϕ and a consequence set $C = \{c_1, \ldots, c_n\}$
- Each c_i has a trigger condition ϕ_i and an effects list $E_i = \langle p_1^i, E_1^i; \ldots; p_k^i, E_k^i \rangle$
  - $\sum_j p_j = 1$ for each E_i

Stochastic Actions: Semantics

- An action is enabled in a state s if its precondition ϕ holds in s
- Executing a disabled action is allowed, but does not change the state
  - Different from deterministic PDDL
  - Motivation: partial observability
  - Precondition becomes factored trigger condition

Stochastic Actions: Semantics (cont.)

- When applying an enabled action to s:
  - Select an effect set for each consequence with enabled trigger condition
  - The combined effects of the selected effect sets are applied atomically to s
  - Unique next state if consequences with mutually consistent trigger conditions have commutative effect sets

Syntax of Probabilistic Effects

```
<effect>      ::= <d-effect>
<effect>      ::= (and <effect>*)
<effect>      ::= (forall (<typed list(variable)>) <effect>)
<effect>      ::= (when <GD> <d-effect>)
<d-effect>    ::= (probabilistic <prob-eff>+)
<d-effect>    ::= <a-effect>
<prob-eff>    ::= <probability> <a-effect>
<a-effect>    ::= (and <p-effect>*)
<a-effect>    ::= <p-effect>
<p-effect>    ::= (not <atomic formula(term)>)
<p-effect>    ::= <atomic formula(term)>
<p-effect>    ::= (<assign-op> <f-head> <f-exp>)
<probability> ::= Any rational number in the interval [0, 1]
```

Correspondence to Components of Stochastic Actions

- Effects list:

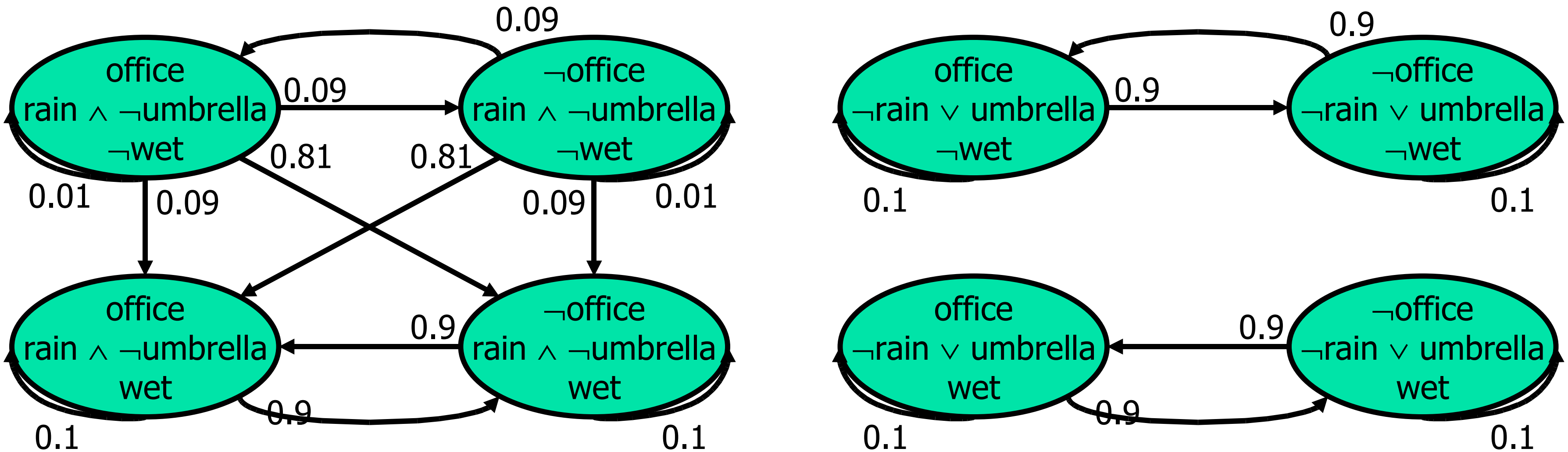
  (probabilistic p_1^i E_1^i … p_k^i E_k^i)

- Consequence:

  (when ϕ (probabilistic p_1^i E_1^i … p_k^i E_k^i))

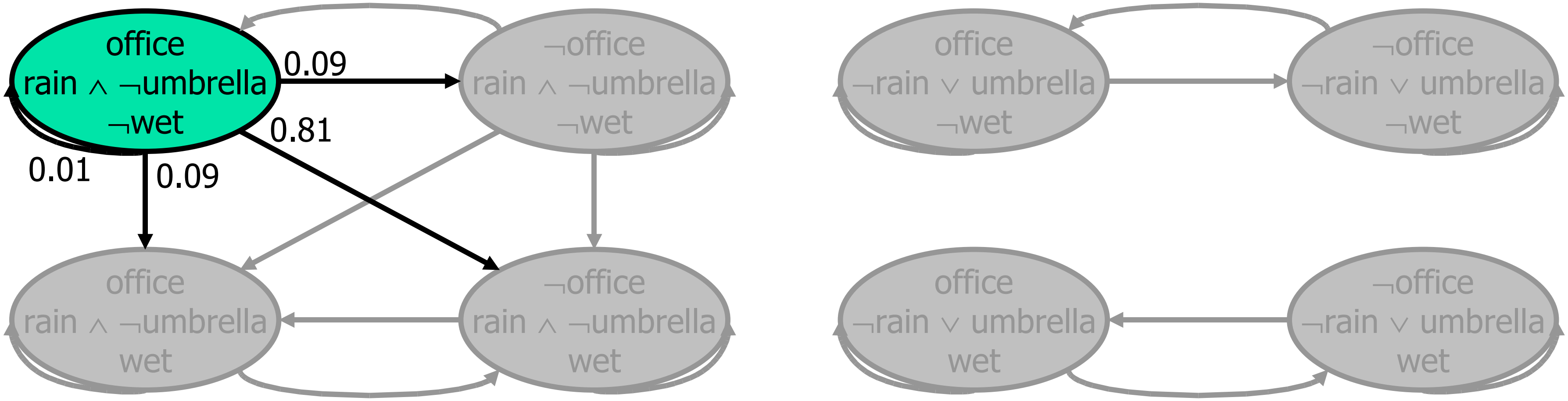
Stochastic Actions: Example

```
(:action move
          :parameters ()
          :effect (and (when (office)
                             (probabilistic 0.9 (not (office))))
                       (when (not (office))
                             (probabilistic 0.9 (office)))
                       (when (and (rain) (not (umbrella)))
                             (probabilistic 0.9 (wet)))))
```



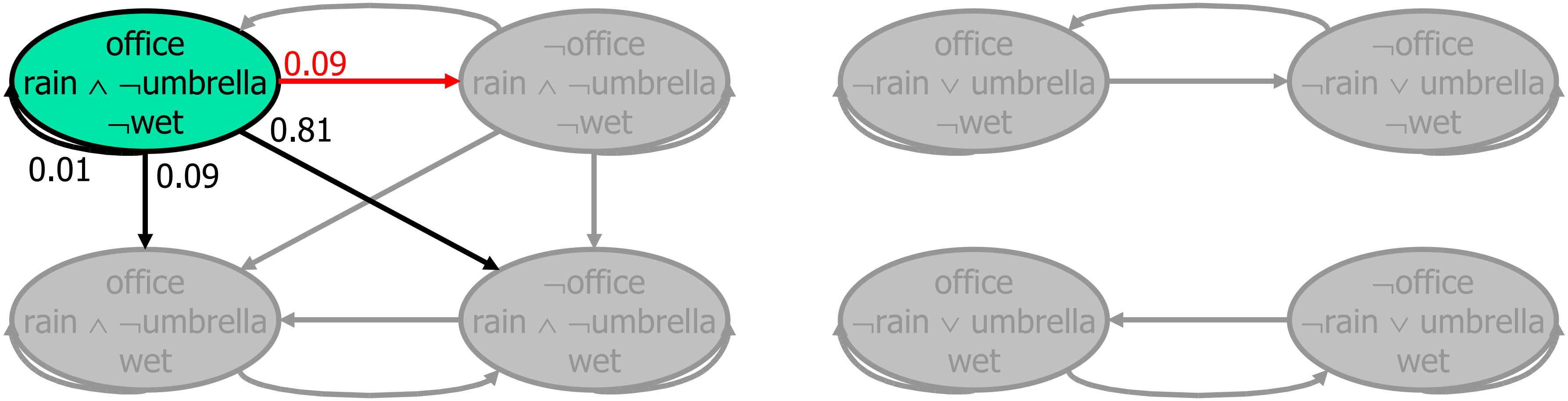
Stochastic Actions: Example

```
(:action move
         :parameters ()
         :effect (and (when (office)
                         (probabilistic 0.9 (not (office)) 0.1 (and)))
                     (when (not (office))
                         (probabilistic 0.9 (office) 0.1 (and)))
                     (when (and (rain) (not (umbrella)))
                         (probabilistic 0.9 (wet) 0.1 (and)))))
```



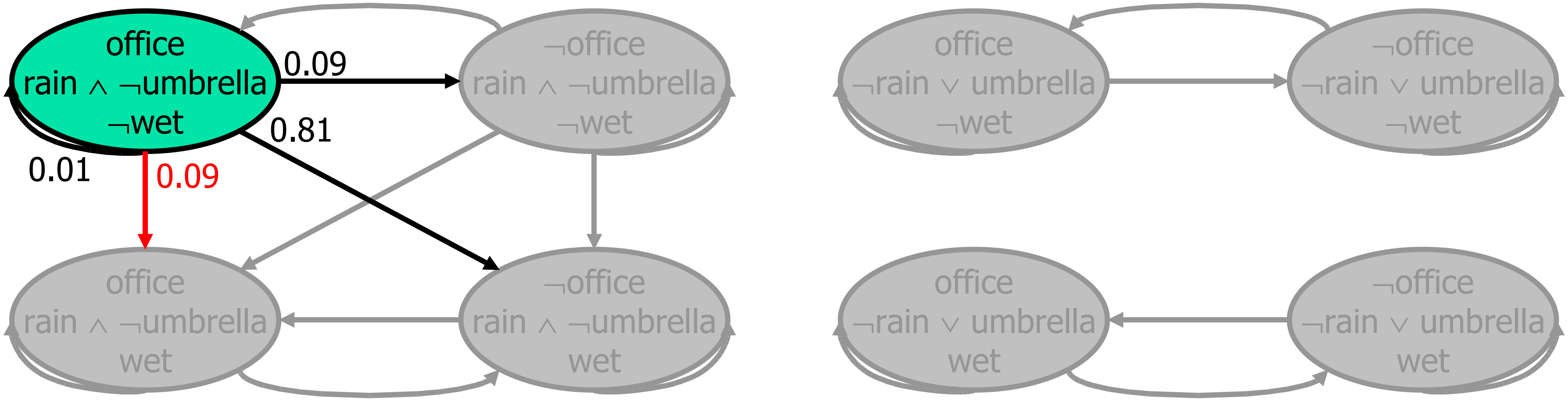
Stochastic Actions: Example

```
(:action move
        :parameters ()
        :effect (and (when (office)
                        (probabilistic 0.9 (not (office)) 0.1 (and)))
                    (when (not (office))
                        (probabilistic 0.9 (office) 0.1 (and)))
                    (when (and (rain) (not (umbrella)))
                        (probabilistic 0.9 (wet) 0.1 (and)))))
```



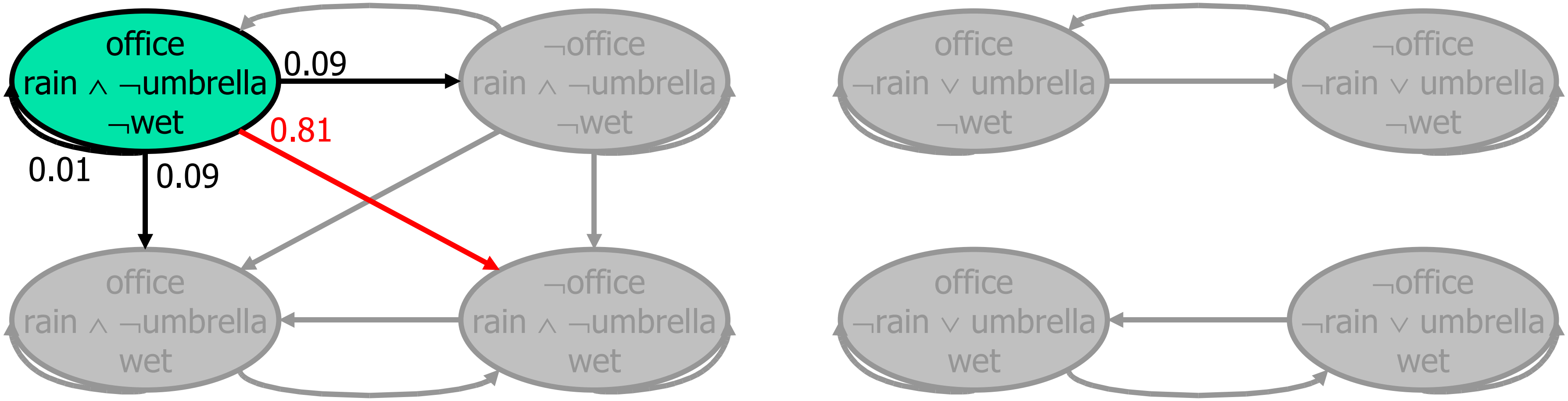
Stochastic Actions: Example

```
(:action move
     :parameters ()
     :effect (and (when (office)
                    (probabilistic 0.9 (not (office)) 0.1 (and)))
                  (when (not (office))
                    (probabilistic 0.9 (office) 0.1 (and)))
                  (when (and (rain) (not (umbrella)))
                    (probabilistic 0.9 (wet) 0.1 (and)))))
```



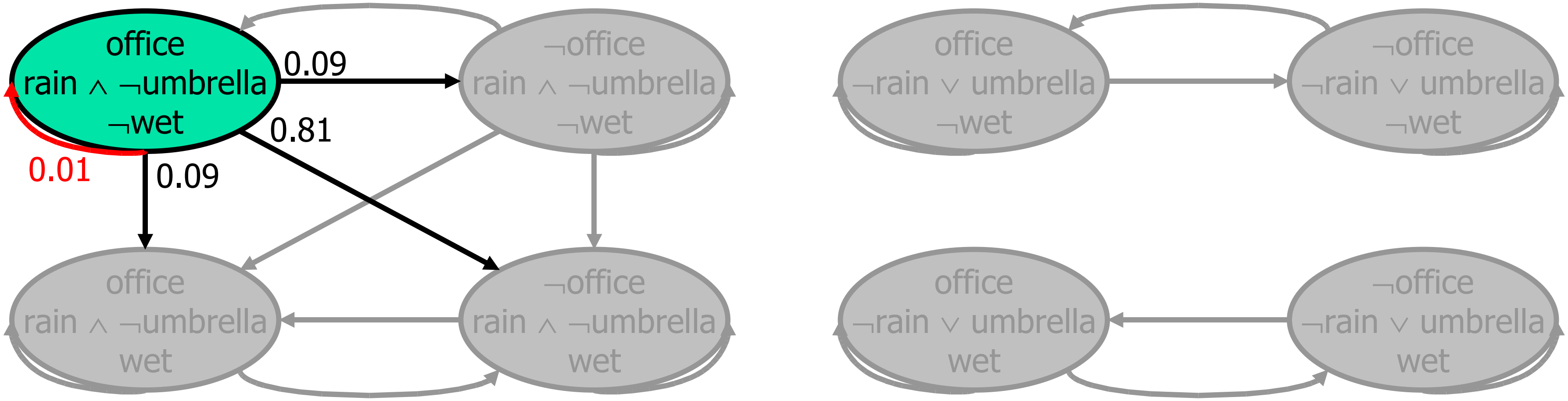
Stochastic Actions: Example

```
(:action move
          :parameters ()
          :effect (and (when (office)
                             (probabilistic 0.9 (not (office)) 0.1 (and)))
                       (when (not (office))
                             (probabilistic 0.9 (office) 0.1 (and)))
                       (when (and (rain) (not (umbrella)))
                             (probabilistic 0.9 (wet) 0.1 (and)))))
```



Stochastic Actions: Example

```
(:action move
          :parameters ()
          :effect (and (when (office)
                           (probabilistic 0.9 (not (office)) 0.1 (and)))
                       (when (not (office))
                           (probabilistic 0.9 (office) 0.1 (and)))
                       (when (and (rain) (not (umbrella)))
                           (probabilistic 0.9 (wet) 0.1 (and))))
```



Exogenous Events

- Like stochastic actions, but beyond the control of the decision maker
- Defined using :event keyword instead of :action keyword
- Common in control theory to say that everything is an event, and that some are controllable (what we call actions)

Exogenous Events: Example

```
(:action move
          :parameters ()
          :effect (and (when (office)
                              (probabilistic 0.9 (not (office))))
                          (when (not (office))
                              (probabilistic 0.9 (office)))))

(:event make-wet
          :parameters ()
          :precondition (and (rain) (not (umbrella)))
          :effect (probabilistic 0.9 (wet)))
```

Expressiveness

- Discrete-time MDPs
  - Exogenous events are so far only a modeling convenience and do not add to the expressiveness

Adding Time

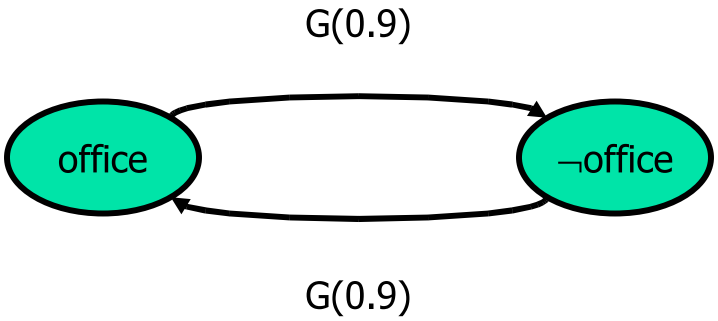
- States have stochastic duration
- Transitions are instantaneous

Actions and Events with Stochastic Delay

- Associate a delay distribution $F(t)$ with each action a
- $F(t)$ is the cumulative distribution function for the delay from when a is enabled until it triggers
- Analogous for exogenous events

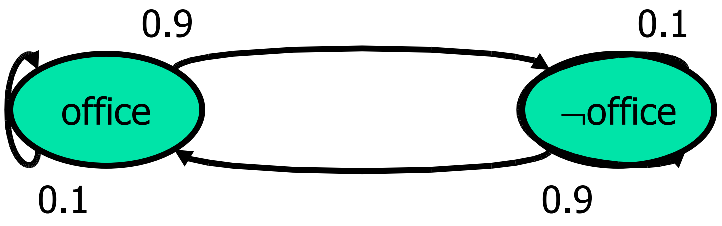
Delayed actions and events: Example

```
(:delayed-action move
                    :parameters ()
                    :delay (geometric 0.9)
                    :effect (and (when (office) (not (office)))
                                 (when (not (office)) (office))))
```



Delayed actions and events: Example

```
(:delayed-action move
                    :parameters ()
                    :delay (geometric 0.9)
                    :effect (and (when (office) (not (office)))
                                   (when (not (office)) (office))))
```



Expressiveness

- Geometric delay distributions
  - Discrete-time MDP
- Exponential delay distributions
  - Continuous-time MDP
- General delay distributions
  - At least semi-Markov decision process

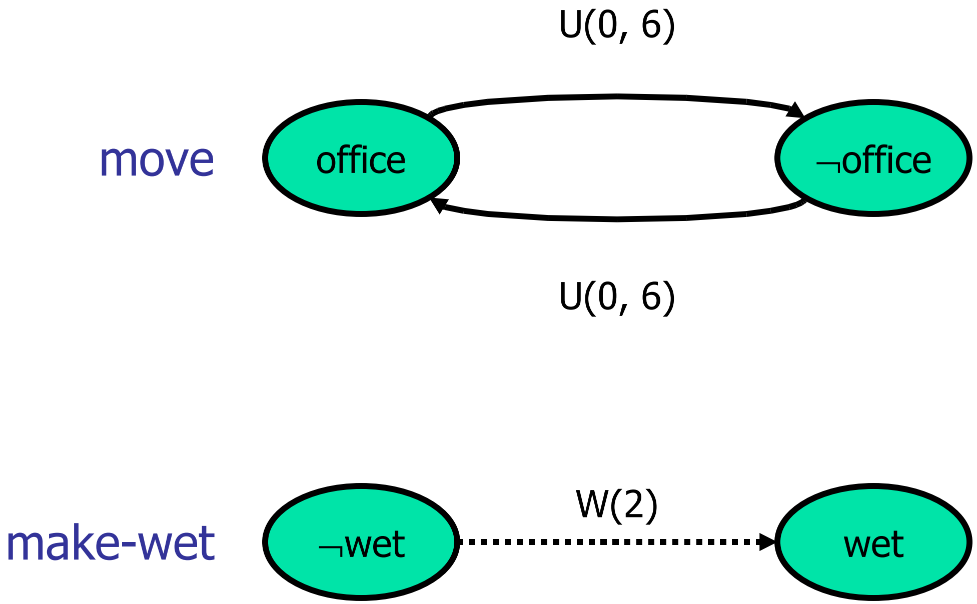
General Delay Distribution: Example

```
(:delayed-action move
                    :parameters ()
                    :delay (uniform 0 6)
                    :effect (and (when (office) (not (office)))
                                   (when (not (office)) (office))))

(:delayed-event make-wet
                    :parameters ()
                    :delay (weibull 2)
                    :precondition (and (rain) (not (umbrella)))
                    :effect (wet))
```

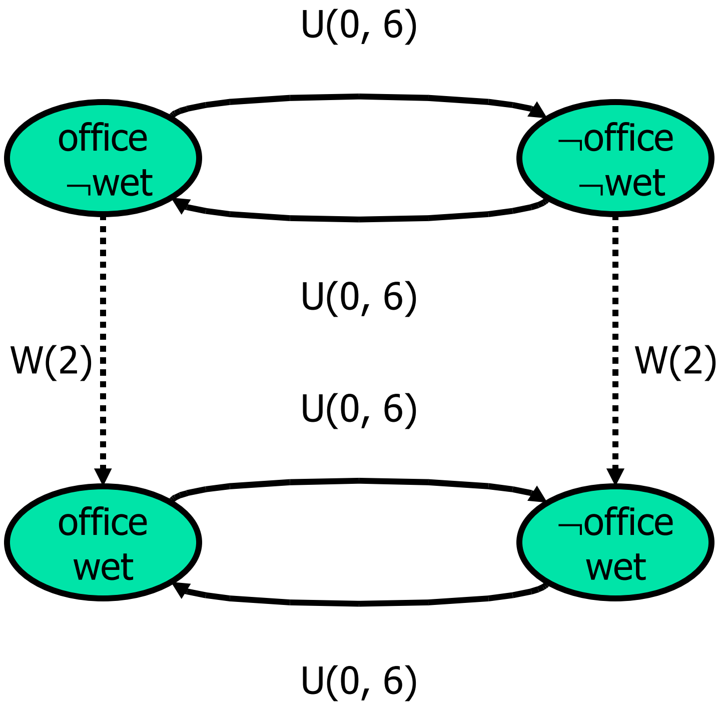
Concurrent Semi-Markov Processes

- Each action and event separately



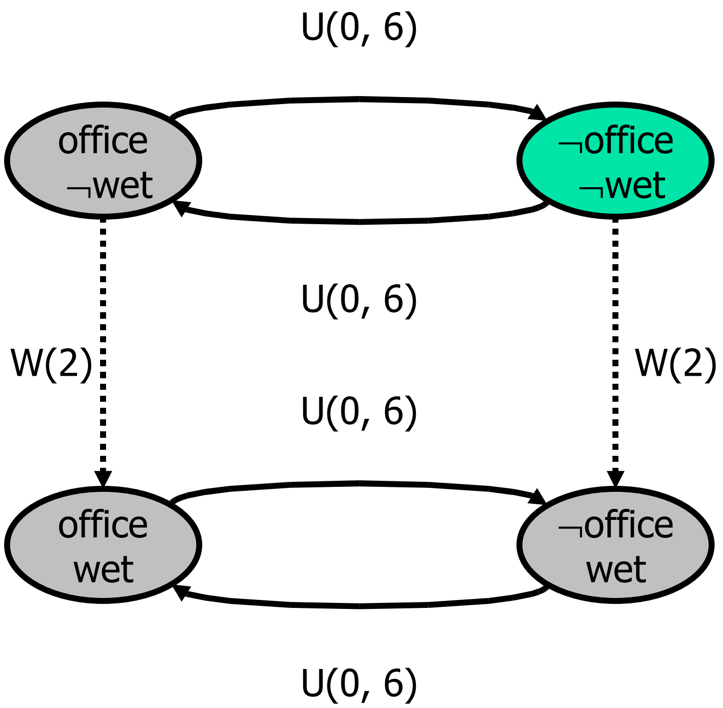
Generalized Semi-Markov Process

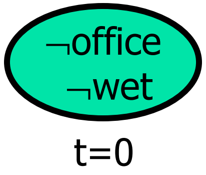
- Putting it together



Generalized Semi-Markov Process (cont.)

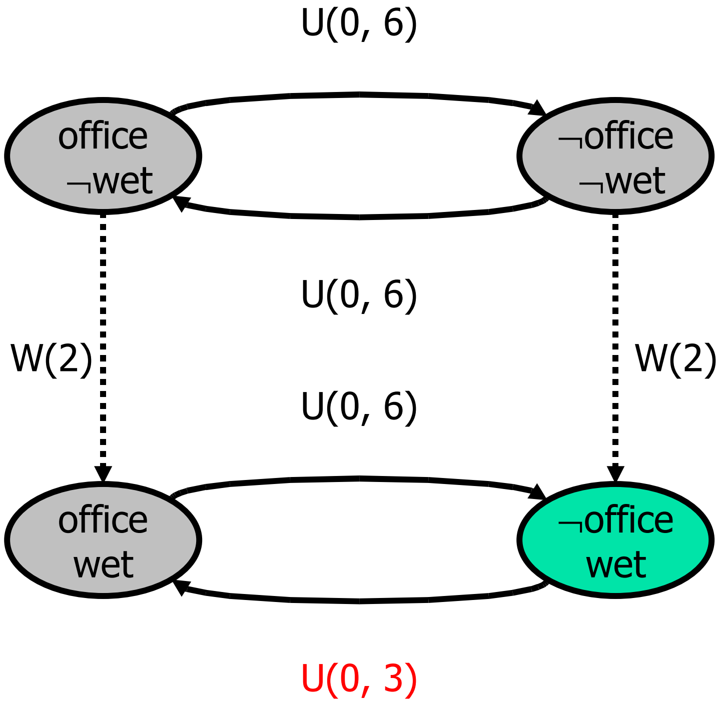
- Why generalized?

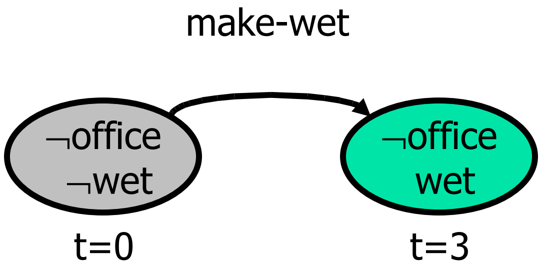




Generalized Semi-Markov Process (cont.)

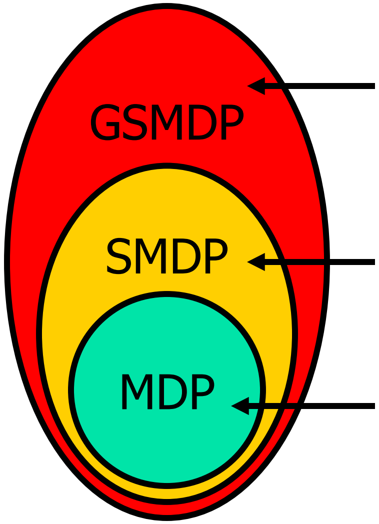
- Why generalized?





Expressiveness

- Hierarchy of stochastic decision processes



concurrency
general delay distributions
probabilistic effects

general delay distributions
probabilistic effects

memoryless delay distributions
probabilistic effects

Probabilistic Temporally Extended Goals

- Goal specified as a CSL (PCTL) formula
  - ϕ ::= true | a | $\phi \wedge \phi$ | $\neg\phi$ | $Pr_{\sim p}(\rho)$
  - ρ ::= $\phi\ U^{\leq t}\ \phi$ | $\Diamond^{\leq t}\ \phi$ | $\Box^{\leq t}\ \phi$

Goals: Examples

- Achievement goal
  - $Pr_{\geq 0.9}(\Diamond \text{ office})$
- Achievement goal with deadline
  - $Pr_{\geq 0.9}(\Diamond^{\leq 5} \text{ office})$
- Achievement goal with safety constraint
  - $Pr_{\geq 0.9}(\neg \text{wet } U \text{ office})$
- Maintenance goal
  - $Pr_{\geq 0.8}(\Box \neg \text{wet})$

Summary

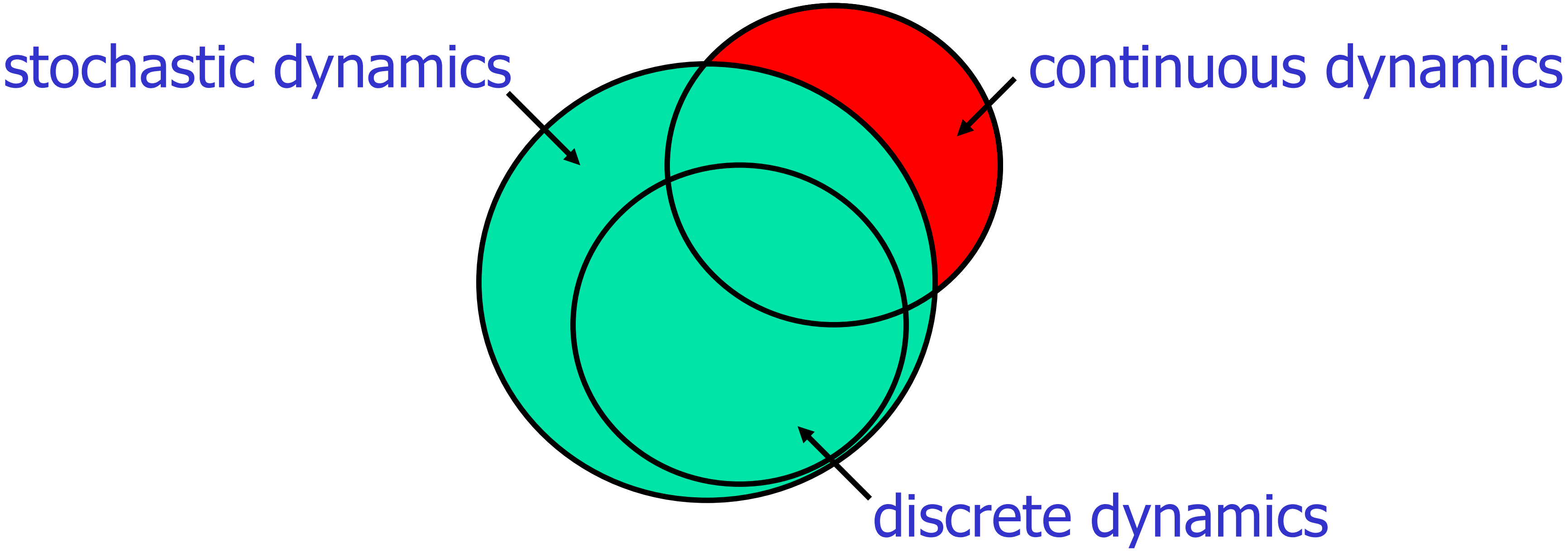
- PDDL Extensions
  - Probabilistic effects
  - Exogenous events
  - Delayed actions and events
- CSL/PCTL goals

Thursday: Planning!

Panel Statement

- Stochastic decision processes are useful
    - Robotic control, queuing systems, …
- Baseline PDDL should be designed with stochastic decision processes in mind
- Formalisms are needed to express complex constrains on valid plans
    - PCTL/CSL goals

Role of PDDL



stochastic dynamics

continuous dynamics

discrete dynamics