Syntax of $\mathcal{A}$: Observation and Queries?

- Observations (O):
  - f **after** a1, . . ., an.
    - f was observed to be true after a1, . . ., an is executed in the initial situation.
    - We are told that f would be true if a1, . . ., an were to be executed in the initial situation.
  - **initially** f.
- Queries (Q): f **after** a1, . . ., an.
  - Will f be true after executing a1, . . ., an in the initial situation.
  - Would f be true if a1, . . ., an were executed in the initial situation.
- Main reasoning relation (D, O) $\models$ Q.

  Defining the above is our next goal. (Semantics!)

Semantics of $\mathcal{A}$: motivating examples.

- D1 = { load **causes** loaded; shoot **causes** ¬ alive **if** loaded }
- D2 = { load **causes** loaded; shoot **causes** ¬ alive;
  executable shoot **if** loaded }
- O1 = { **initially** ¬ loaded; **initially** alive }
- O2 = { **initially** alive }
- O3 = = { **initially** alive; ¬ alive **after** load, shoot }
- Plan verification (hypothetical reasoning): (D1, O1) $\models$ ¬ alive **after** load, shoot?
- Simple planning: (D1, O1) $\models$ ¬ alive **after** α?
- Conformant planning: (D1, O2) $\models$ ¬ alive **after** α?
- Observation assimilation: (D1, O3) $\models$ **initially** loaded?
- Exercise: Replace D1 by D2 in the above reasoning tasks.

Semantics of $\mathcal{A}_0$: lead-in to the definition of $\models$

- Domain descriptions in $\mathcal{A}_0$ do not have executability conditions.
- Our goal: To define (D, O) $\models$ Q.
- *Lets first consider O such that it is of the form* **initially** *f.*
- Q1: What kind of information does such an O encode.
- Q2: What kind of information D encodes.
- Hint for Q1: initial state.
- Hint for Q2: Each effect proposition: a **causes** f **if** p1, . . ., pn
  means a transition p1, ... pn $\xrightarrow{a}$ f
- Ans1: O tells us what the initial state of the world is. (If O is not complete – does not have information about all the fluents – it leads us to multiple initial states.)
  Given O, what is an initial state (σ) corresponding to O? Define? (What is a good representation of a state?)

- Ans2: D defines the transition between states due to actions.
  Given a D, define the transition function Φ_D implied by D?
- What is Φ_{D_1}? What are initial states corresponding to O1 and O2?
- The set of states: { { }, {alive}, {loaded}, {alive, loaded} }.
  - Φ_{D_1}(load,{ }) = {loaded}.
    Φ_{D_1}(load,{ alive }) = {alive,loaded}.
    Φ_{D_1}(load,{loaded }) = {loaded}.
    Φ_{D_1}(load,{alive, loaded }) = {alive,loaded}.
  - Φ_{D_1}(shoot,{ }) = {}.
    Φ_{D_1}(shoot,{ alive }) = {alive}.
    Φ_{D_1}(shoot,{loaded }) = {loaded}.
    Φ_{D_1}(shoot,{alive, loaded }) = {loaded}.
- Initial state corresponding to O1: {alive}
- Initial states corresponding to O2: {alive}; {alive, loaded}
- How to define (D1, O1) $\models$ Q.

Semantics of $\mathcal{A}_0$: defining $\models$

- Given: D, O and the language of D and O. The language consists of a set of actions Act and a set of fluents Fl. A *state* is a subset of Fl.
- A fluent f holds in a state σ if $f \in \sigma$. A fluent literal $\neg f$ holds in a state σ if $f \notin \sigma$.
- *Characterizing O when it consists of only observations about the initial state*:

  A set of fluents σ is an initial state corresponding to O, if for all observations of the form **initially** g (where g is a fluent literal), g holds in σ.
- Given a domain description D in $\mathcal{A}_0$ its transition function is defined as follows:
  - For all actions a, fluents f, and states σ:
    - If a **causes** f **if** $p1, \ldots, pn \in D$ and $p1, \ldots, pn$ hold in σ then f must be in $\Phi(a, \sigma)$.
    - If a **causes** $\neg f$ **if** $p1, \ldots, pn \in D$ and $p1, \ldots, pn$ hold in σ then f must not be in $\Phi(a, \sigma)$.
    - If D does not include such effect propositions (about a and f) then $f \in \Phi(a, \sigma)$ iff $f \in \sigma$.

   * For any domain description D, there is at most one such function and we refer to its as Φ_D. If Φ_D exists we say D is consistent.
 - If Φ_D exists then it is defined as follows:
   * $E_D^+(a,\sigma) = \{f : a$ **causes** f **if** $p1,\ldots,pn \in D$ and $p1...pn$ hold in $\sigma\}$
   * $E_D^-(a,\sigma) = \{f : a$ **causes** $\neg f$ **if** $p1,\ldots,pn \in D$ and $p1...pn$ hold in $\sigma\}$
   * $\Phi_D(a,\sigma) = \sigma \cup E_D^+(a,\sigma) \setminus E_D^-(a,\sigma)$

- **For general observations:** A set σ_0 is said to be an initial state corresponding to a consistent domain description D and a set of observations O if for all observations of the form f **after** $a_1,\ldots,a_m$, f holds in $\Phi_D(a_m, \Phi_D(a_{m-1},\ldots \Phi_D(a_1,\sigma_0)\ldots))$.
  $\Phi_D(a_m, \Phi_D(a_{m-1},\ldots \Phi_D(a_1,\sigma_0)\ldots))$ will be denoted by $[a_m,\ldots,a_1]\sigma_0$.
- **Models of (D,O):** (σ_0,Φ) is a model of (D,O) iff Φ is the transition function of D and σ_0 is an initial state corresponding to D and O.
- (D,O) is said to be consistent if it has a model.
- (D,O) is said to be complete if it has a unique model.
- $(D,O) \models f$ **after** $a_1,\ldots,a_m$ if for all models (σ_0,Φ) of (D,O), f holds in $[a_m,\ldots,a_1]\sigma_0$.

How hard is it to implement $\models$: summary of complexity results

- R1: Given D, deciding whether Φ_D exists or not is polynomial.
- R2: Given a consistent D and a set of fluents σ_0, deciding whether σ_0 is an initial state with respect to (D, O), is polynomial.
- R3: Deciding if (D, O) is consistent is NP-complete.
- R4: Deciding if $(D, O) \models f$ **after** $a_1, \ldots, a_n$ is coNP complete.
- R5: Given a D, a complete set of observations about the initial state O, and a fluent f, finding a polynomial sequence of actions $a_1, \ldots, a_m$ such that $(D, O) \models f$ **after** $a_1, \ldots, a_m$ is NP-complete. (i.e., Feasible planning in $\mathcal{A}_0$ with a complete initial state, and simple goal, is NP-complete.)