Hypothetical reasoning in initial state complete $\mathcal{A}_0$: tr_do_1 and tr_q_1

1. For a fluent f suppose the domain description D contains only the following effect axioms that either make f true or make it $false$.

   - a_1 **causes** f **if** $p_{11}, \ldots, p_{1r_1}$ $\ldots$ a_l **causes** f **if** $p_{l1}, \ldots, p_{lr_l}$
   - b_1 **causes** $\neg f$ **if** $q_{11}, \ldots, q_{1s_1}$ $\ldots$ b_m **causes** $\neg f$ **if** $q_{m1}, \ldots, q_{ms_m}$

   For an integer n, formulas obtained by instantiating N below with $1 \ldots n$ are in $tr\_do_1(D, O, n)$.

   $f[N+1] \iff (a_1[N] \wedge p_{11}[N] \wedge \ldots \wedge p_{1r_1}[N]) \vee \ldots \vee (a_l[N] \wedge p_{l1}[N] \wedge \ldots \wedge p_{lr_l}[N]) \vee (f[N] \wedge (\neg b_1[N] \vee \neg q_{11}[N] \vee \ldots \vee \neg q_{1s_1}[N]) \wedge \ldots \wedge (\neg b_m[N] \vee \neg q_{m1}[N] \vee \ldots \vee \neg q_{ms_m}[N]))$

2. for each **initially** f in O the proposition $f[1]$ is in $tr\_do_1(D, O, n)$.

3. A query Q given by f **after** $a_1, \ldots, a_t$ is translated to:

   $(a_1[1] \wedge \neg a_2[1] \ldots \wedge \neg a_r[1]) \wedge \ldots \wedge (\neg a_1[t] \wedge \ldots \neg a_{t-1}[t] \wedge a_t[t] \wedge \neg a_{t+1}[t] \ldots \wedge \neg a_r[t])$

   where $a_1, \ldots, a_r$ are all the actions in D. We will refer to the above as $tr\_q_1(Q)$.

4. Proposition 1: Given a consistent domain description D, a complete set of initial state observations O and a query Q of the form f **after** $a_1, \ldots, a_t$, $(D, O) \models Q$ iff $tr\_do_1(D, O, t) \cup tr\_q_1(Q) \models f[t+1]$.

5. Lemma 1: For a consistent domain description D and a complete set of initial state observations O, (D, O) has a unique model.

6. Lemma 2: Given a consistent domain description D, a complete set of initial state observations O and a query Q of the form f **after** $a_1, \ldots, a_t$, let $M = (\sigma_0, \Phi)$ be a unique model of (D, O).

   The propositional theory $tr\_do_1(D, O, t) \cup tr\_q_1(Q)$ has a unique model M given by $\{f[k+1] : f \in [a_1, \ldots, a_k]\sigma_0, 0 \leq k \leq t\} \cup \{a[i] : 1 \leq i \leq t\}$.

   Proof-sketch (by induction on t)

   Inductive step: $f \in [a_1, \ldots, a_k]\sigma_0$ if

   (i) $f \in E_D^+(a_k, [a_1, \ldots, a_{k-1}]\sigma_0)$ or

   (ii) $f \in [a_1, \ldots, a_{k-1}]\sigma_0$ and $f \notin E_D^-(a_k, [a_1, \ldots, a_{k-1}]\sigma_0)$.

   (i) Let $f \in E_D^+(a_k, [a_1, \ldots, a_{k-1}]\sigma_0)$. This means there must exist an effect propositions of the form a **causes** f **if** $p_1, \ldots, p_n$ such that $p_1, \ldots, p_n$ hold in $[a_1, \ldots, a_{k-1}]\sigma_0$. By the induction hypothesis if p_i is a positive fluent literal then $p_i[k]$ must be true in M and otherwise $p_i[k]$ must be false in M. Thus to satisfy the formula in $tr\_do_1(D, O, t)$, $f[k+1]$ must be true in M.

7. A linear regression algorithm: Start from $f[t+1]$, use $tr\_do_1(D, O, t)$ to obtain what should be true at t. Use $tr\_q_1(Q)$ to use the information about what action occurs at t and simplify the regressed formula. Pursue each branch of the simplified regressed formula as above until the initial state.

Hypothetical reasoning in initial state complete $\mathcal{A}_0$: tr_do_2 and tr_q_2

- Given D, O and n as mentioned in the previous slide, $tr\_do_2(D,O,n)$ consists of the formulas in $tr\_do_1(D,O,n)$ and the following:

  If $a_1, \ldots, a_r$ are all the actions in the domain then

  - For all $1 \leq i \leq t$, $a_1[i] \vee \ldots \vee a_r[i]$ is in $tr\_do_2(D,O,n)$.
  - For all $1 \leq i \leq t$ and $1 \leq j,k \leq r$ where $j \neq k$ $\neg(a_j[i] \wedge a_k[i])$ is in $tr\_do_2(D,O,n)$.

- A query Q given by f **after** $a_1, \ldots, a_t$ is translated to:

  $a_1[1] \wedge \ldots \wedge a_t[t]$

  which we will refer to as $tr\_q_2(Q)$.

- Proposition 2: Given a consistent domain description D, a complete set of initial state observations O and a query Q of the form f **after** $a_1, \ldots, a_t$, $(D,O) \models Q$ iff $tr\_do_2(D,O,t) \cup tr\_q_2(Q) \models f[t+1]$.

Planning in initial state complete $\mathcal{A}_0$

1. Proposition 3: Let D be a consistent domain description, O be a complete set of observations about the initial state, and g be the goal. A sequence of actions $a_1, \ldots, a_t$ is a plan iff there is a model of $tr\_do_2(D, O, t) \cup g[t+1]$ containing $\{a_1[1], \ldots, a_t[t]\}$ as the only action occurrences.
2. The above suggests that to find a plan, if we know that a plan of length t exists then we should give a propositional solver the propositional theory $tr\_do_2(D, O, t) \cup tr\_q_2(Q) \cup g[t+1]$ and ask it to find a model. Each model (if exists) will encode a plan.

   If we do not know the plan length but have an idea of an upper bound, then

   - we can have a dummy action which does not affect the world; or
   - we can replace $g[t+1]$ by $g[1] \vee \ldots \vee g[t+1]$.

Some good SAT solvers

1. M. Moskewicz, C. Madigan, Y. Zhao, L. Zhang, and S. Malik. Chaff: Engineering an efficient sat solver. In *Design Automation Conference*, 2001.
2. J. Marques-Silva and K. Sakallah. GRASP: a search algorithm for propositional satisfiability. *IEEE transactions on computers*, 48:506–521, 1999.
3. H. Zhang. SATO: An efficient propositional prover. In *Proc. of CADE'97*, pages 272–275, 1997.